

**Universidade Federal do Pará
Instituto de Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica**

DISSERTAÇÃO DE MESTRADO

MARCUS GUERRRA DA ROCHA

METODOLOGIA PARA O DESPACHO DE POTÊNCIA REATIVA
VISANDO O CONTROLE DE TENSÃO BASEADO EM ALGORITMOS
GENÉTICOS

DM 06 / 2009

UFPA/ITEC/PPGEE
Campus Universitário do Guamá
Belém – Pará - Brasil

**Universidade Federal do Pará
Instituto de Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica**

MARCUS GUERRRA DA ROCHA

METODOLOGIA PARA O DESPACHO DE POTÊNCIA REATIVA
VISANDO O CONTROLE DE TENSÃO BASEADO EM ALGORITMOS
GENÉTICOS

**Dissertação de Mestrado
submetida à Banca Examinadora
do Programa de Pós – Graduação
em Engenharia Elétrica da UFPA
para obtenção do Grau de Mestre
em Engenharia Elétrica.**

**Orientação: Prof. Dr. Marcus
Vinicius Alves Nunes**

UFPA/ITEC/PPGEE
Campus Universitário do Guamá
Belém – Pará - Brasil

R713m Rocha, Marcus Guerra da

Metodologia para o despacho de potência reativa visando o controle de tensão baseado em algoritmos genéticos / Marcus Guerra da Rocha; orientador, Marcus Vinicius Alves Nunes.-Belém, 2009.

Dissertação (Mestrado) - Universidade Federal do Pará, Instituto de Tecnologia, Programa de Pós-graduação em Engenharia Elétrica, Belém, 2009.

1. Sistemas de energia elétrica – planejamento 2. Inteligência computacional. 3. Sistemas de energia elétrica – simulação. I. orientador. II. título.

CDD 22. ed. 621.3191

Universidade Federal do Pará
Instituto de Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica

“METODOLOGIA PARA O DESPACHO DE POTÊNCIA REATIVA VISANDO O
CONTROLE DE TENSÃO BASEADO EM ALGORITMOS GENÉTICOS”

AUTOR: MARCUS GUERRA DA ROCHA

DISSERTAÇÃO DE MESTRADO SUBMETIDA À AVALIAÇÃO DA BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS – GRADUAÇÃO EM ENGENHARIA ELÉTRICA DA UNIVERSIDADE FEDERAL DO PARÁ E JULGADA ADEQUADA PARA OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA COM ÊNFASE EM COMPUTAÇÃO APLICADA

APROVADA EM :

BANCA EXAMINADORA:

Prof. Dr. Marcus Vinicius Alves Nunes (UFPA)
ORIENTADOR

Prof. Dr. Ubiratan Holanda Bezerra (UFPA)
MEMBRO DA BANCA EXAMINADORA

Prof. Dr. Roberto Célio Limão (UFPA)
MEMBRO DA BANCA EXAMINADORA

Prof. Dra. Maria Emilia Lima de Tostes (UFPA)
MEMBRO DA BANCA EXAMINADORA

Prof. Dr. Raimundo Nonato das M. Machado (UFPA)
MEMBRO DA BANCA EXAMINADORA

VISTO:

Prof. Dr. Marcus Vinicius Alves Nunes (UFPA)
COORDENADOR DO PROGRAMA DE PÓS –
GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DEDICATÓRIA

A minha família.

AGRADECIMENTOS

A todas as pessoas que me ajudaram no desenvolvimento deste trabalho, com destaque ao integrantes do GSEI, Antônio, Conceição, Luiz Miguel, Andrey Ramos, Andrey Lopes, Diego Rosal, João Paulo e Edgar, que nunca hesitaram em contribuir com seu conhecimentos.

Agradeço a todos os professores do PPGEE, em especial ao meu orientador Prof. Marcus Vinicius, e aos professores Ubiratan e Jurandy, que sempre estiveram dispostos a repassar grande sabedoria e experiência que possuem.

A ELETRONORTE que financiou a pesquisa, que teve como um dos resultados esta dissertação. A FAPESPA, pela bolsa que me foi concedida, assim como pelo excelente trabalho quem vem desenvolvendo na busca pela maturidade científica de nossa região.

A minha família, que mesmo atravessando pelos momentos mais difíceis, me deram o suporte necessário na tomada de minhas decisões.

RESUMO

Este trabalho tem como objetivo apresentar um aplicativo para auxiliar no planejamento de sistemas elétricos, através de uma metodologia para controle de tensão e minimização das perdas, através da otimização da injeção de reativos, mantendo a tensão nos barramentos dentro de limites pré estabelecidos. A metodologia desenvolvida é baseada em um sistema híbrido, que utiliza inteligência computacional baseada em um algoritmo genético acoplado a um programa de fluxo de carga (ANAREDE), que interagem para produzir uma solução ótima. Os resultados obtidos mostram que a técnica baseada no algoritmo genético é bem adequada ao tipo de problema ora tratado referente a minimização de perdas reativas e a melhoria do perfil da tensão em redes elétricas, sendo este atualmente um problema crítico em parte do Sistema Interligado Nacional (SIN).

ABSTRACT

This work aims to present a software to support the planning of power systems, through a methodology of voltage control and losses minimization, by the optimization of reactive injection, keeping the voltage in the bus between the established boundaries. The developed methodology is based in a hybrid system, that use artificial intelligence based in a genetic algorithm linked to a load flow software(ANAREDE), that interact to produce an optimal solution. The results showed that the technique based in a genetic algorithm is adequate to the problem of minimization the reactive losses.

LISTA DE TABELAS

Tabela 4.1 – Mapeamento de <i>shunts</i>	47
Tabela 6.1 – Parâmetros do AG	67
Tabela 6.2 – Configuração do Microcomputador	73
Tabela 6.3 – Valores de Reativo Propostos	74
Tabela A.1 – Descrição dos Campos	80
Tabela B.1 – Tensão e Limites nas Barras	81
Tabela B.2 – Valores de Tensão Antes e Após a Otimização	83
Tabela C.1 – Formato dos Dados de Área	86
Tabela C.2 – Formato dos Dados de Barra	87

LISTA DE FIGURAS

Figura 2.1 – Algoritmo Genético Simplificado	21
Figura 2.2 – Fluxograma de um Algoritmo Genético Simples	21
Figura 2.3 – AG binário	28
Figura 2.4 – Operação de <i>crossover</i> em um AG binário	30
Figura 2.5 – AG Contínuo.	31
Figura 3.1 – Estrutura Básica de um Sistema Elétrico de Potência	36
Figura 3.2 – Modelo de Diagrama Unifilar	37
Figura 4.1 – Fluxograma Algoritmo	46
Figura 4.2 - Representação do Cromossomo	48
Figura 4.3 – Interação com o Fluxo de Carga	49
Figura 5.1 – Módulos do Aplicativo	53
Figura 5.2 – Diagrama de Componentes	55
Figura 5.3 – Diagrama de Classe	56
Figura 5.4 – Diagrama Caso de Uso	57
Figura 5.5 – Diagrama de Atividade	58
Figura 5.6 – Arquivo de Entrada	59
Figura 5.7 – Diagrama Entidade – Relacionamento	60
Figura 5.8 – Código de Execução “DARE”	61
Figura 5.9 – Código de Execução “DBAR”	62
Figura 5.10 – Identificação do <i>Shunt</i>	63
Figura 6.1 – Sistema Interligado Nacional	65
Figura 6.2 – Sistema ELN-MG	66
Figura 6.3 – Melhores Indivíduos de Todos os Experimentos	67
Figura 6.4 – Melhores Indivíduo	68
Figura 6.5 – Interface Inicial	69
Figura 6.6 – Definir Arquivo de Entrada de Dados	69
Figura 6.7 – Definição das Áreas de Interesse	70
Figura 6.8 – Definição das Barras à serem Monitoradas	71

Figura 6.9 – Definição dos Limites de Tensão	71
Figura 6.10 – Definição dos Shunts Monitorados	72
Figura 6.11 – Definição dos Limites de Reativos dos Shunts	73
Figura 6.12 – Comparação entre os Valores de Reativos Antes e Depois da Otimização	74
Figura 6.13 – Valores de Tensão nas Barras	75
Figura A.1 – Arquivo de Configuração	80
Figura C.1 – Código de Execução DARE	85
Figura C.2 – Código de Execução DBAR	86
Figura D.1 – Melhor Indivíduo no Experimento 1	88
Figura D.2 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 1	88
Figura D.3 – Melhor Indivíduo no Experimento 2	89
Figura D.4 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 2	89
Figura D.5 – Melhor Indivíduo no Experimento 3	90
Figura D.6 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 3	90
Figura D.7 – Melhor Indivíduo no Experimento 4	91
Figura D.8 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 4	91
Figura D.9 – Melhor Indivíduo no Experimento 5	92
Figura D.10 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 5	92
Figura D.11 – Melhor Indivíduo no Experimento 6	93
Figura D.12 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 6	93
Figura D.13 – Melhor Indivíduo no Experimento 7	94
Figura D.14 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 7	94
Figura D.15 – Melhor Indivíduo no Experimento 8	95
Figura D.16 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 8	95

Figura D.17 – Melhor Indivíduo no Experimento 9	96
Figura D.18 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 9	96
Figura D.19 – Melhor Indivíduo no Experimento 10.....	97
Figura D.20 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 10	97

SUMÁRIO

Capítulo 1 – Introdução	16
1.1 – Introdução	16
1.2 – Revisão Bibliográfica	17
1.3 – Contribuições	18
1.4 – Estrutura do Trabalho	18
Capítulo 2 – Algoritmos Genéticos	20
2.1 – Introdução	20
2.2 – Considerações Iniciais	20
2.3 – Definição de Cromossomo	22
2.4 – Definição de Função Objetivo	23
2.5 – População de Indivíduos	24
2.6 – Seleção	24
2.7 – <i>Crossover</i>	26
2.8 – Mutação	26
2.9 – Representação de Algoritmos Genéticos	27
2.9.1 – Algoritmos Genéticos Binários	27
2.9.1.1 – Codificação e Decodificação de Cromossomos	28
2.9.1.2 – <i>Crossover</i>	30
2.9.1.3 – Mutação	30
2.9.2 – Algoritmos Genéticos Contínuo	30
2.9.2.1 – <i>Crossover</i>	31
2.9.2.2 – Mutação	33
2.10 – Considerações Finais	33
Capítulo 3 – Conceitos Básicos Sobre Sistemas de Potência	34
3.1 – Introdução	34

3.2 – Definições Gerais	34
3.2.1 – Componentes	35
3.2.2 – Representação Unifilar	36
3.2.3 – Normalização – Grandezas por Unidade (p.u)	37
3.3 – Fluxo de Potência	38
3.3.1 – Matriz de Admitância	38
3.3.2 – Classificação das Barras	39
3.3.2.1 – Barra de Referência	39
3.3.2.2 – Barra de Carga (PQ)	39
3.3.2.2 – Barra de Tensão Controlada (PV)	40
3.3.3 – Fluxo de Potência Generalizado	40
3.3.4 – Método de Solução	40
3.3.4.1 – Método de Newton - Raphson	41
3.3.4.1 – Método Desacoplado Rápido	44
3.4 – Considerações Finais	44
Capítulo 4 – Metodologia de Otimização	45
4.1 – Introdução	45
4.2 – Visão Geral da Metodologia de Otimização	45
4.3 – Representação do Cromossomo	47
4.4 – Interação com o Fluxo de Carga	48
4.5 – Detalhes do Algoritmo Genético Utilizado	50
4.6 – Definição da Função Objetivo	50
4.7 – Considerações Finais	51
Capítulo 5 – Detalhamento do Software	52
5.1 – Introdução	52
5.2 – Ferramentas e Bibliotecas Utilizadas	52
5.3 – Histórico de Desenvolvimento	53
5.4 – Arquitetura do Software em Termos Estruturais	54

5.4.1 – Estrutura	54
5.4.2 – Interação com o Usuário	56
5.5 – Dados de Entrada	58
5.6 – Banco de Dados	60
5.6.1 – Tabela “AREA”	60
5.6.2 – Tabela “BARRA”	61
5.6.3 – Tabela “SHUNT”	62
5.7 – Considerações Finais	63
Capítulo 6 – Estudo de Caso – Sistema Mato Grosso	64
6.1 – Introdução	64
6.2 – Sistema Base	64
6.3 – Arquivo de Entrada de Dados	66
6.4 – Parâmetros do Algoritmo Genético	66
6.5 – Execução do Estudo de Caso	68
6.6 – Conclusões	75
Capítulo 7 – Conclusão e Sugestões para Trabalhos Futuros	76
Bibliografia	77
Anexo A – Arquivo de Configuração do Algoritmo Genético	80
Anexo B – Dados do Sistema e dos Experimentos	81
Anexo C – Códigos de Execução do Arquivos de Entrada de Dados do ANAREDE Utilizados no Aplicativo	85
Anexo D – Experimentos Realizados	88

CAPITULO 1 – Introdução

1.1 Introdução

A preocupação com o desenvolvimento de ferramentas computacionais que auxiliem a operação e o planejamento de sistemas elétricos de potência é uma constante entre as empresas do setor elétrico, visto que a complexidade dos problemas deste setor geralmente acabam por inviabilizar a solução e análise destes via métodos que não sejam automatizados.

A proposta desta dissertação é apresentar uma metodologia, baseada em software, para controle de tensão dos barramentos e a minimização das perdas em um sistema elétrico, através de um método baseado em um algoritmo genético acoplado ao um programa de fluxo de carga.

O programa de fluxo de carga utilizado neste aplicativo é o ANAREDE, desenvolvido pelo Centro de Pesquisas de Energia Elétrica (CEPEL), e o padrão de entrada de dados segue o mesmo adotado por este.

O principal ponto inovador desta metodologia, é que esta utiliza em sua composição padrões e softwares já consagrados pelo setor elétrico, possibilitando assim uma maior usabilidade da ferramenta, de forma que sistemas elétricos modelados para serem utilizados com o ANAREDE, que facilmente são encontrados em empresas ou órgãos do setor, podem ser utilizados sem que haja necessidade de qualquer forma de ajuste ou conversão para um novo formato de dados.

Um outro ponto a destacar, é o fato de que a utilização de um programa de fluxo de carga consolidado pelo setor elétrico, faz com que o aplicativo esteja sempre atualizado, e possibilita que sejam utilizados sistemas elétricos reais e complexo, como o Sistema Interligado Nacional (SIN), que possui diversas especificidades e complexidades, que programas de fluxo de carga simplificados não seriam capazes de gerar.

1.2 Revisão Bibliográfica

A utilização de algoritmos genéticos na solução de problemas que envolvem a otimização da configuração operativa de sistemas elétricos, tem sido uma opção amplamente considerada por muitos pesquisadores em diversas instituições acadêmicas ao redor do mundo, como pode ser observado a seguir.

WONG et al. (1995) utiliza um algoritmo genético para solucionar o problema de fluxo de carga.

PINHEIRO et al. (1998) utiliza uma técnica de otimização baseada em algoritmo genético, acoplada a um programa de fluxo de carga, com a intenção de obter um controle de tensão e a minimização das perdas em um sistema elétrico de potência.

Zhu et al. (2002) propõe um método para reconfiguração de redes de distribuição baseada em um algoritmo genético refinado.

MENDEL et al. (2006) apresenta um sistema híbrido de otimização, utilizado na restauração de sistemas elétricos de potência. Este sistema é baseado em um algoritmo genético, que utiliza um programa de fluxo de carga para avaliar as opções sugeridas pelo algoritmo.

MALACHI et al. (2006) apresenta um algoritmo para seleção da ação de controle corretiva, para a tensão nas barras e para a potencia reativa nos geradores em sistemas de potencia, utilizando algoritmos genéticos.

MIASAKI et al. (2007) apresenta uma estratégia para representar o problema de planejamento da expansão da transmissão a longo prazo. Um algoritmo genético especializado foi implementado com intenção de fornecer como solução, a quantidade e o local, onde devem ser instaladas as novas linhas e transmissão e os dispositivos de compensação em série.

Beê (2007) em sua dissertação de mestrado, implementa um algoritmo genético na busca ótima dos tipos de bancos de capacitores(fixo ou automático), tamanhos dos bancos (em kvar) e localização dos mesmos em alimentadores de distribuição.

TING et al. (2008) apresenta uma metodologia de otimização híbrida, utilizando os Algoritmos Genéticos e Enxame de Partículas para a avaliação de fluxo de carga em sistemas elétricos de potência.

ESTEVAM (2008), em sua tese de doutorado propõe um algoritmo *branch-and-bound* não linear para resolver o problema de planejamento e despacho ótimo de fontes de potência reativa em sistemas de energia elétrica.

1.3 Contribuições

O desenvolvimento deste trabalho apresenta ganhos, tanto para o meio acadêmico quanto para as empresas do setor elétrico, sendo os principais:

- Demonstra a eficácia do uso de técnicas de inteligência artificial para soluções de problemas complexos envolvendo o setor elétrico;
- Apresenta uma metodologia, onde se utiliza um software externo para auxiliar no cálculo do *fitness* de indivíduos, em um algoritmo genético;
- Apresenta uma solução de software para o auxílio no planejamento da operação de sistemas elétricos, onde utiliza-se padrões de entrada de dados já consolidados pelo setor elétrico.

1.4 Estrutura do Trabalho

A dissertação foi desenvolvida em seis capítulos, conforme a descrição apresentada a seguir:

- **Capítulo 2 – Algoritmos Genéticos:** Neste capítulo é apresentada uma introdução teórica sobre algoritmos genéticos, abordando os principais conceitos e técnicas, com o intuito de fornecer um subsídio mínimo necessário para o leitor desta dissertação.
- **Capítulo 3 – Conceitos Básicos Sobre Sistemas de Potência:** Conceitos Básicos e revisão de alguns aspectos teóricos sobre sistemas de potência e sobre fluxo de

carga são abordados neste capítulo, visando, assim como no Capítulo 2, fornecer um subsídio mínimo necessário para o leitor desta dissertação.

- **Capítulo 4 – Metodologia de Otimização:** Neste capítulo são discutidos assuntos relativos à metodologia de otimização utilizada, detalhando as características do algoritmo genético, assim como a interação com o programa de fluxo de carga.
- **Capítulo 5 – Detalhamento do Software:** Neste capítulo é apresentada uma visão geral sobre o aplicativo desenvolvido, abordando a sua estrutura de software, assim como suas funcionalidades.
- **Capítulo 6 – Estudo de Caso – Sistema Mato Grosso:** Neste capítulo são demonstradas as funcionalidades do sistema apresentado, através de um estudo de caso, utilizando como base um sistema real.
- **Capítulo 7 – Conclusões e Sugestões para Trabalhos Futuros**

CAPITULO 2 - Algoritmos Genéticos

2.1 - Introdução

Os algoritmos genéticos (AG's) têm sido amplamente utilizados em diversas áreas do conhecimento para a otimização de problemas reais, devido principalmente a sua aplicabilidade em sistemas que envolvem muitos parâmetros. Dentre as diversas áreas pode-se citar: a engenharia como um todo, a computação científica, a biologia, a economia, entre outras.

Neste capítulo o algoritmo genético é abordado de forma geral, sendo apresentados seus principais conceitos, dando-se ênfase nas duas principais derivações de AG, que são os AG's binários e os AG's contínuos, que têm sido amplamente usados pela comunidade científica na otimização de grandes sistemas (HAUPT, 2004).

2.2 - Considerações Iniciais

Muitos problemas reais possuem um alto grau de complexidade, geralmente envolvendo um grande número de parâmetros que devem ser considerados, o que se reflete em um elevado número de variáveis a serem combinadas para que se obtenha uma solução ótima. A consequência desses fatores é que a otimização destes via técnicas de otimizações clássicas, como por exemplo, o método simplex, torna-se lenta e em certos casos inviável.

Com isso, surgiu a necessidade do desenvolvimento de técnicas que solucionassem esse tipo de problema com maior rapidez e eficácia. Dentre as várias técnicas estão os Algoritmos Genéticos (Holland, 1975), Exame de Partículas (Kennedy, 1995), Redes Neurais (Haykin, 1994).

Os Algoritmos Genéticos foram propostos primeiramente por John Holland (Holland, 1975), e estes se constituem de uma técnica de otimização baseada nos princípios da genética e da seleção natural (HAUPT, 2004).

A idéia básica dos AG's é a de manter uma população de indivíduos (ou **chromossomos**), representando soluções candidatas para o problema, que evoluem ao longo do tempo (ou **gerações**) através de um processo de competição, em que os mais aptos (melhores *fitness*) têm maiores chances de sobreviver e de se reproduzir. A reprodução se baseia em um processo de **seleção** de indivíduos e de modificação das soluções candidatas que eles representam, através de operadores como cruzamento (ou *crossover*) e **mutação**. Um esquema simplificado de um AG é apresentado na forma de um algoritmo na figura 2.1, e através de uma fluxograma na figura 2.2.

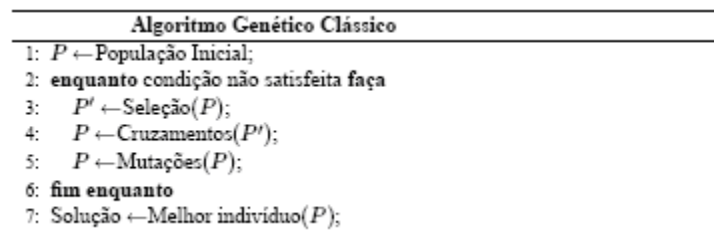


Figura 2.1 – Algoritmo Genético simplificado

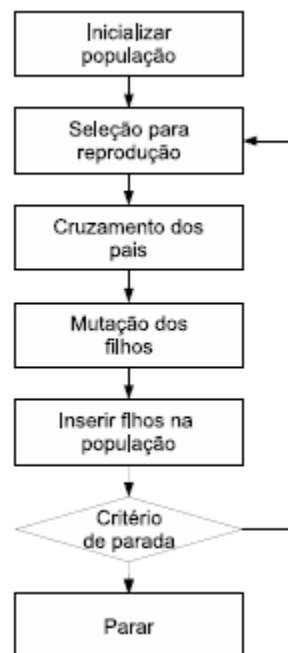


Figura 2.2 – Fluxograma de um Algoritmo Genético Simples

Dentre as principais vantagens do uso de AG's estão (HAUPT, 2004):

- Trabalha com variáveis contínuas e discretas;
- Não requer informações de derivadas;
- Realiza buscas simultâneas;
- Trabalha com um grande número de variáveis;
- Trabalha bem com computação paralela;
- Otimiza variáveis em funções extremamente complexas;
- Prover uma lista de variáveis otimizadas, não apenas uma como em outros métodos;
- Pode trabalhar com variáveis codificadas;
- Trabalha com diversos tipos de dados dentre estes: dados gerados numericamente, dados experimentais, funções analíticas etc.

2.3 – Definição de *Cromossomo*

O conjunto das variáveis que serão otimizadas é denominado de **cromossomo**. Considerando N_{var} , o número de variáveis, e $p_1, p_2, p_3, \dots, p_{N_{var}}$, a representação de cada variável a ser otimizada, pode-se então representar um cromossomo da seguinte forma:

$$cromossomo = [p_1, p_2, p_3, \dots, p_{N_{var}}] \quad (2.1)$$

Tomando como exemplo o caso de um *cromossomo* com representação binária e que possui um N_{var} igual a 3, tem-se que este poderia ser configurado da seguinte forma:

$$Cromossomo = [0010, 0101, 0101]$$

Assim como o caso de um *cromossomo* com representação contínua e que possui o mesmo valor para N_{var} , ter-se-ia então a seguinte configuração:

$$Cromossomo = [2.364, 8, 10.9632]$$

O número de variáveis a serem otimizadas, deve ser mínimo, de forma que apenas sejam utilizadas as variáveis de interesse do problema, pois a necessidade de processamento computacional cresce proporcionalmente com o aumento do número de variáveis.

2.4 - Definição da Função Objetivo

A função objetivo do sistema geralmente é uma função matemática, um experimento ou um procedimento, que têm como entrada um conjunto de variáveis, ou seja, um *cromossomo*, e como saída a aptidão (*fitness*) de um determinado *cromossomo*, como pode ser observado na equação 2.2.

$$fitness = f(cromossomo) \quad (2.2)$$

O *fitness* de um *cromossomo* determina o quanto a solução representada pelo mesmo está apta para solucionar o problema em questão. Este parâmetro é de suma importância no processo de otimização em um AG, visto que é a partir deste que são determinadas quais soluções têm maior probabilidade de serem mantidas para uma próxima geração no processo iterativo do Algoritmo Genético.

2.5 - População de Indivíduos

A população de indivíduos em um AG é definida como sendo um conjunto de *cromossomos*, onde seu tamanho, ou seja, a quantidade de *cromossomos* que contém esta população é determinada de acordo com a necessidade do problema, sendo na maioria dos casos um número fixo, e em outros variável, como pode ser observado em (Shi, 2003) e (Koumouisis, 2006) .

$$população = \begin{bmatrix} \text{cromosomo 1} \\ \text{cromosomo 2} \\ \text{cromosomo 3} \\ \vdots \\ \text{cromosomo } \eta \end{bmatrix} \quad (2.3)$$

Para que se inicie o processo iterativo e evolucionário do AG, é necessário a formação de uma população inicial, que pode ser gerada de forma aleatória, respeitando as restrições impostas pelo problema em questão, ou de forma pré-determinada, com o intuito de melhorar o desempenho do AG. O uso desta última solução deve ser visto com cautela, pois a indevida utilização desta pode induzir o algoritmo a convergir para uma solução local, ou seja, uma solução não otimizada.

2.6 – Seleção

Este operador tem como objetivo, simular o mecanismo de seleção natural que atua sobre as espécies biológicas, em que os pais mais capazes geram mais filhos, ao mesmo tempo em que pais menos aptos também podem gerar descendentes, privilegiando os indivíduos com função de avaliação alta, sem desprezar completamente aqueles indivíduos com função de avaliação extremamente baixa (Linden, 2006) .

Os *cromossomos* com maior *fitness* são geralmente selecionados de acordo com uma taxa de seleção X_{rate} , como apresentado na equação 2.4.

$$N_{keep} = X_{rate} \cdot N_{pop} \quad (2.4)$$

Onde: N_{keep} = é a quantidade de *cromossomos* que devem ser selecionados;

N_{pop} = é a quantidade de *cromossomos* que constituem população.

O valor de X_{rate} é escolhido arbitrariamente, de acordo com as características do problema, sendo que a utilização de taxas muito altas podem ter como consequência uma população pouco diversificada. Os *cromossomos* selecionados serão mantidos para compor a próxima geração de *cromossomos*, enquanto que os não selecionados serão descartados.

Os N_{keep} *cromossomos* selecionados por possuírem os maiores *fitness*, serão apenas uma parte dos N_{pop} que irão compor a *geração* seguinte. Sendo que os $N_{pop} - N_{keep}$ indivíduos, serão formados a partir de operações de *crossover* entre pares de indivíduos, selecionados a partir dos N_{keep} *cromossomos*.

A seleção dos pares para a realização da operação de *crossover* pode ser realizada através de diversos métodos, dentre eles podemos destacar:

- **Método do Torneio:** Este método consiste em selecionar uma série de indivíduos aleatoriamente de uma população e fazer com que eles entrem em competição direta pelo direito de se reproduzir, utilizando os seus *fitness* como fator determinante nesta, ou seja, o indivíduo que tiver maior *fitness* vencerá a competição.

O número de indivíduos que participam do torneio é definido pelo **tamanho do torneio(k)**, sendo que o valor de 'k' tem que ser maior ou igual a 2, visto que para valores menores que este não haverá competição.

- **Método da Amostragem Estocástica Uniforme:** Neste, todos os indivíduos são mapeados para seguimentos contíguos de uma linha, sendo que o tamanho de cada segmento é proporcional ao valor da avaliação do indivíduo que está sendo mapeado. Para selecionar – se os

n indivíduos que serão pais na respectiva geração, é sorteado um número i entre 0 e $1/n$, que serve como base para o sorteio. Depois são atribuídos n ponteiros que passam a apontar para segmentos de reta, nas posições $i, i + 1/n, i + 2/n, \dots, i + n - 1/n$. Os indivíduos que foram apontados serão selecionados para a aplicação dos operadores genéticos.

- **Pareamento do topo ao fim:** Esse método consiste em parear os *cromossomos* na ordem decrescente de aptidão entre os N_{keep} cromossomo de maior aptidão.

- **Pareamento Aleatório:** Nesse método o par de cromossomos é escolhido aleatoriamente dentro dos N_{keep} cromossomos de maior aptidão.

- **Pareamento através do “Método da Roleta”:** Nesse método cada cromossomo é associado a uma probabilidade, de acordo com seu *fitness*, de forma que quanto maior for este, maior será a probabilidade do correspondente cromossomo ser selecionado. Sendo assim, é como se cada cromossomo ocupasse uma “fatia” de uma roleta, sendo que quanto maior a probabilidade de determinado *cromossomo* maior será a “fatia” deste. A seleção é realizada através de um número aleatório entre 0 e 1 , sendo selecionado o *cromossomo* que possuir o número encontrado.

2.7 – Crossover

Esse operador tem como objetivo, a criação de um ou dois *cromossomos*, a partir de um par de *cromossomos* selecionados. Sendo que, esta operação deve ser feita de tal forma que os *cromossomos* gerados devem possuir informação do par gerador destes.

A forma como esta operação é realizada varia geralmente com o tipo de representação utilizada no AG, ou com a estratégia de otimização utilizada.

2.8 – Mutação

A mutação é um operador que tem como principal função prover diversidade na população do sistema, com intuito de diminuir a probabilidade de o AG encontrar um mínimo ou

um máximo local. A forma como é realizada esta operação, assim como o *crossover*, depende do tipo de representação utilizada, e a quantidade de indivíduos que serão modificados pela mutação é determinado pela *taxa de mutação*, sendo esta geralmente pequena, o que significa que apenas uma pequena porcentagem da população sofre mutação.

2.9 - Representações de Algoritmos Genéticos

Os AG podem ser classificados de acordo com a representação das variáveis utilizada, sendo que as formas mais comuns são:

- Algoritmo genético binário;
- Algoritmo genético contínuo.

2.9.1 - Algoritmo Genético Binário

Os algoritmos genéticos binários são AG's que representam as variáveis com uma *string* binária codificada. Um diagrama de blocos de um AG binário pode ser observado na figura 2.3(HAUPT, 2004).

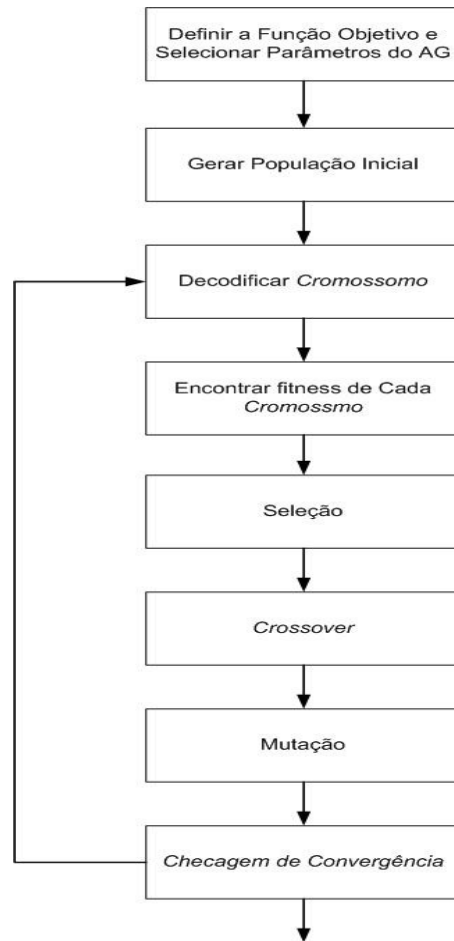


Figura 2.3 – AG binário

2.9.1.1 - Codificação e Decodificação de Cromossomos

Em virtude de o Algoritmo Genético binário trabalhar apenas com *bits*, faz – se necessário a codificação das variáveis do sistema para a forma binária, salvo os casos em que as variáveis do sistema já são variáveis binárias (ou seja, apenas podem assumir os valores ‘0’ e ‘1’).

Geralmente a codificação das variáveis é realizada através de uma quantização, sendo que quanto maior for o número de bits, menor será o erro. Uma formulação matemática proposta para a codificação binária de acordo com (HAUPT, 2004) é:

$$p_{norm} = \frac{p_n - p_{lo}}{p_{hi} - p_{lo}} \quad (2.5)$$

$$gene[m] = round \left\{ p_{norm} \cdot 2^m - \sum_{p=1}^{m-1} gene[p] \cdot 2^{-p} \right\} \quad (2.6)$$

Onde:

p_{norm} = variável normalizada, $0 \leq p_{norm} \leq 1$.

p_{lo} = Limite inferior da variável.

p_{hi} = Limite superior da variável.

$round \{ \}$ = Arredonda para o inteiro mais próximo

$gene [m]$ = Versão binária de p_n

E para decodificação segundo também (HAUPT, 2004), pode ser:

$$p_{quant} = \sum_{m=1}^{N_{gene}} gene[m] \cdot 2^{-m} + 2^{-(m+1)} \quad (2.7)$$

$$q_n = p_{quant} \cdot (p_{hi} - p_{lo}) + p_{lo} \quad (2.8)$$

Onde:

p_{quant} = versão quantizada de p_{norm}

q_n = versão quantizada de p_n

2.9.1.2 – Crossover

Em AG's binários, a operação de *crossover*, considerando a utilização de apenas um ponto de *crossover*, é dada da seguinte forma:

- 1 – Seleciona-se, arbitrariamente, um ponto dentro do *cromossomo* que será denominado de ponto de *crossover*, que será o mesmo para ambos os *cromossomos-pais*.
- 2 – Realiza-se uma permutação entre as partes após o ponto de *crossover*, gerando assim dois novos *cromossomos* como poder ser observado na figura 2.4.

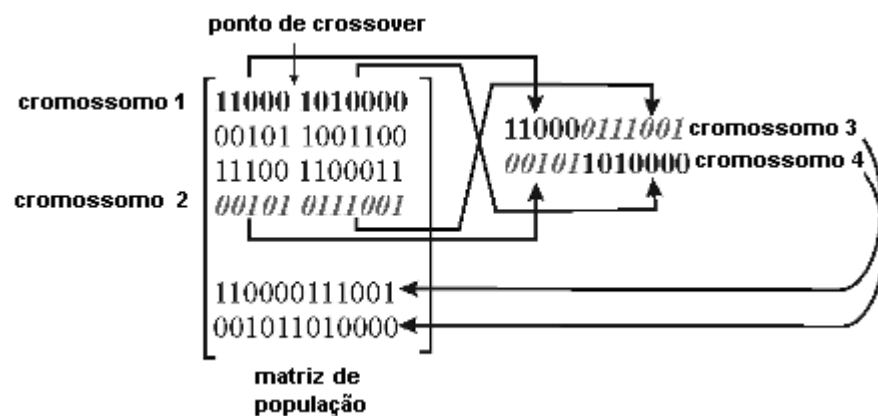


Figura 2.4 – Operação de *crossover* em um AG binário

2.9.1.3 - Mutação

Em AG's binários a mutação, resume-se geralmente em mudar aleatoriamente um *gene* de '1' para '0', sendo que a quantidade de mutações vai ser definida através da *taxa de mutação*.

2.9.2 - Algoritmo Genético Contínuo

Quando o problema a ser solucionado necessita de uma alta precisão, os AG's binários tornam-se inviáveis para esse tipo de problemas, pois seriam necessários muitos bits para a

representação de uma única variável. Nestes problemas o uso de AG's contínuos torna-se mais adequado.

Neste tipo de AG as variáveis são representadas de forma real, ou seja, as variáveis não necessitam ser codificadas, sendo a precisão limitada apenas pelo erro roundoff do computador. Um diagrama de bloco deste tipo de AG pode ser observado na figura 2.5(HAUPT, 2004).

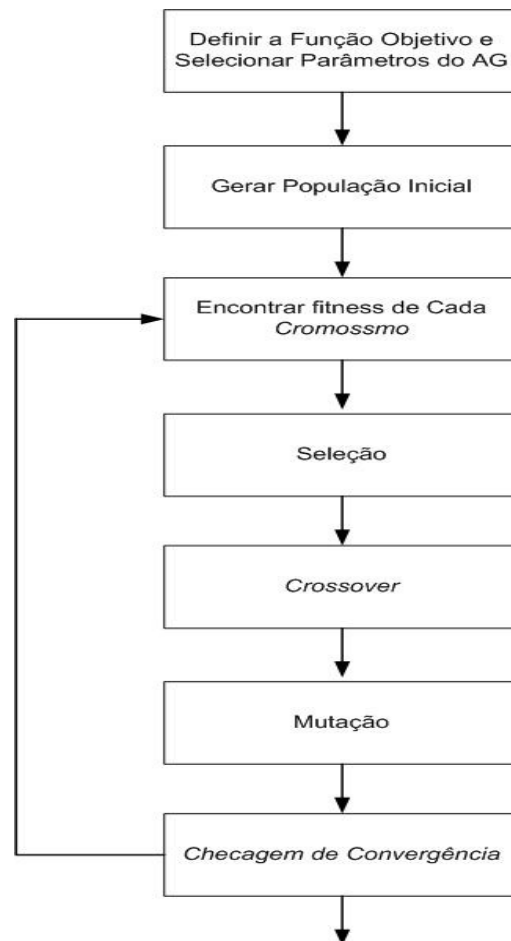


Figura 2.5 – AG Contínuo.

2.9.2.1 – *Crossover*

A operação de *crossover* em AG's contínuos pode ser realizada de diversas formas, sendo importante que os *cromossomos* gerados possuam informações de ambos os *cromossomos-pais*.

Um método muito usado para realizar esta operação consiste em combinar os *genes* dos *cromossomos* selecionados de acordo com as equações 2.9 e 2.10.

$$p_{new1} = \beta \cdot p_{1n} + (1 - \beta) \cdot p_{2n} \quad (2.9)$$

$$p_{new2} = (\beta - 1) \cdot p_{1n} + \beta \cdot p_{2n} \quad (2.10)$$

Onde: β é um número aleatório entre 0 e 1;

p_{1n} = enésima variável do *cromossomo-pai 1*;

p_{2n} = enésima variável o *cromossomo-pai 2*;

p_{new1} = novo *gene* do *cromossomo-filho 1*;

p_{new2} = novo *gene* do *cromossomo-filho 2*.

Sendo assim, escolhendo-se um ponto dentro do cromossomo para ser o *ponto de crossover*, os indivíduos gerados podem assumir a forma apresentada nas equações 2.11 e 2.12, caso sejam utilizados para a combinação os *genes* a direita do *ponto de crossover*, ou então podem assumir a forma apresentada nas equações 2.13 e 2.14, a seguir, caso sejam utilizados para a combinação os *genes* a esquerda do *ponto de crossover* para a combinação.

$$cromossomo-filho_1 = [p_{11}, p_{12}, \dots, p_{new11}, p_{new12}, \dots] \quad (2.11)$$

$$cromossomo-filho_2 = [p_{21}, p_{22}, \dots, p_{new21}, p_{new22}, \dots] \quad (2.12)$$

$$cromossomo-filho_1 = [p_{new11}, p_{new12}, \dots, p_{1j}, p_{1j+1}, \dots] \quad (2.13)$$

$$cromossomo-filho_2 = [p_{new21}, p_{new22}, \dots, p_{2j}, p_{2j+1}, \dots] \quad (2.14)$$

2.9.2.2 - Mutação

A mutação em AG's contínuos é realizada de forma semelhante à forma como é realizada em AG's binários, sendo que nesta, em vez de alterarmos um bit de '0' para '1', alteramos uma variável por um número aleatório dentro de uma determinada faixa de acordo com os limites do problema a ser solucionado.

2.10 – Considerações Finais

Neste capítulo foi realizada uma breve explanação sobre Algoritmos Genéticos, aonde foi priorizada a apresentação das principais características sobre assunto, dando ao leitor subsidio suficiente para entender os capítulos posteriores.

Esta técnica vem sendo amplamente utilizada ao longo dos anos, na busca de soluções otimizadas para problemas do setor elétrico como pode ser visto em (), () e ().

CAPITULO 3 - Conceitos sobre Sistemas de Potência

3.1 – Introdução

Este capítulo tem como objetivo introduzir alguns conceitos básicos sobre sistemas de potência e sobre o problema do fluxo de carga, com intuito de prover um subsídio mínimo necessário para o leitor leigo no assunto compreender as discussões e desenvolvimentos realizados neste trabalho.

3.2 – Definições Gerais

O conjunto de equipamentos utilizados para o fornecimento de energia elétrica aos consumidores é denominado de Sistemas Elétricos de Potência. Este, por sua vez, é composto por equipamentos geradores, acumuladores e transformadores, linhas de transmissão, sistemas de distribuição, e por dispositivos e instalações de controle. Sua estrutura permite que, em um primeiro estágio, a energia obtida da natureza, sob diversas formas, seja transformada em energia elétrica e convenientemente convertida para o transporte, em um segundo momento, através de um sistema de linhas de transmissão e, o estágio seguinte, distribuição, até seus consumidores. Assim, as linhas de transmissão, em uma etapa intermediária, conectam os equipamentos de geração aos sistemas de distribuição, e estes levam a energia até as diversas cargas de uma determinada área (Mendel, 2006).

O projeto de sistemas de potência é desenvolvido de forma que um número adequado de estações geradoras sejam interligadas por linhas de transmissão, de tal modo que a energia produzida por este conjunto possa ser utilizada na extensão de área coberta pela sistema.

Os problemas encontrados mais facilmente em estudos sobre análise de sistema de potência são o fluxo de carga, curto – circuito e a estabilidade. Cada uma destas análises engloba uma classe de problemas encontrados em sistemas elétricos de potência. Esta divisão pode ser realizada da seguinte forma, adotando como critério o tempo (Prof. Bira):

1. $\Delta t = 10^{-3} s$: transitórios eletromagnéticos.
2. $\Delta t = 10^{-1} s$: transitórios eletromecânicos.
3. $\Delta t = 1 s$: atuação do regulação de velocidade.
4. $\Delta t = 10^4 a 10^2 s$: controle carga – frequência.
5. $\Delta t = 1 a 4$ semanas: planejamento da operação do sistemas
6. $\Delta t = 5 a 20$ anos: planejamento da expansão do sistema.

Dos problemas apresentados acima, todos utilizam soluções baseadas em fluxo de carga seja diretamente, como é caso dos itens 4, 5 e 6, ou indiretamente, através do calculo de condições iniciais etc.

3.2.1 – Componentes

Um sistema elétrico é composto por diversos tipos de componentes, ver figura 3.1, divididos com relação à natureza da atividade realizada, sendo os três principais:

- Sistemas de Geração;
- Sistemas de Transmissão;
- Sistemas de Distribuição.

Estrutura Básica de um Sistema Elétrico

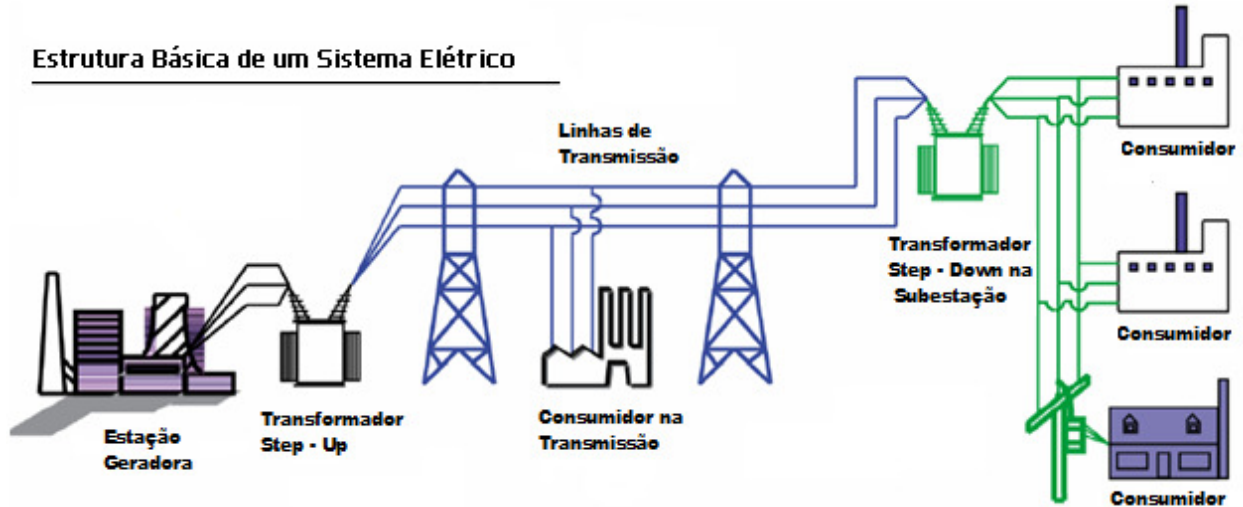


Figura 3.3 – Estrutura Básica de um Sistema Elétrico de Potência

3.2.2 – Representação Unifilar

Considerando uma situação de equilíbrio entre as fases, os sistemas elétricos de potência podem ser tratados matematicamente com apenas uma das fases e o neutro, o que normalmente acontece com os sistemas de transmissão. Simplificando ainda mais, retirando o neutro, e utilizando símbolos padronizados para representar os componentes de um sistema, pode-se obter uma representação concisa e objetiva para estes sistemas, ver figura 3.2, que é denominada de *diagrama unifilar*.

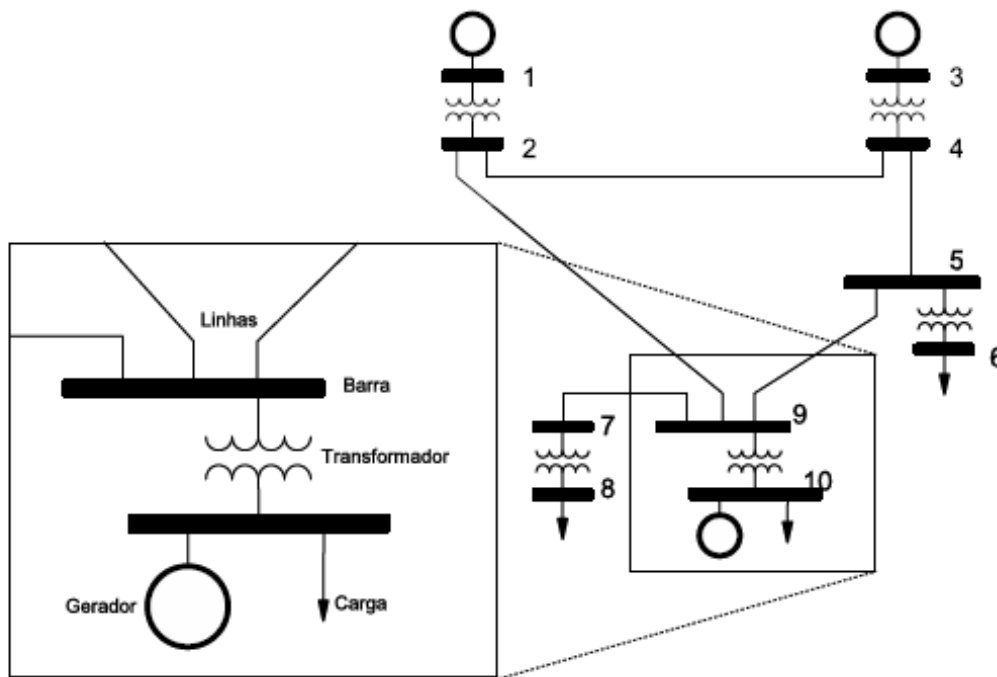


Figura 3.2 – Modelo de Diagrama Unifilar

3.2.3 – Normalização – Grandezas por Unidade (p.u.)

Comumente as grandezas de um sistema elétrico de potência são representadas de uma forma normalizada, com intuito de facilitar as operações matemática realizadas na análise de circuitos.

Esta representação é calculada em relação ao valor de referência ou valor base, podendo esta ser uma porcentagem ou em valor por unidade [p.u], como pode ser observado na equação 3.1.

$$valor[p.u] = \frac{valor\ real}{V_{ref}} \quad (3.1)$$

onde V_{ref} é o valor de referência ou base.

3.3 – Fluxo de Potência

O cálculo do fluxo de carga (ou fluxo de potência) de uma rede de energia elétrica consiste essencialmente na determinação do estado elétrico da rede, ou seja, valores de tensão nos nós e, a partir daí, a distribuição dos fluxos de potência nos ramos, dadas certas restrições de geração, condições de carga e configuração topológica da rede.

Neste tipo de problema, a modelagem do sistema é estática, ou seja, a rede é representada por um conjunto de equações e inequações algébricas. Esse tipo de representação é utilizada em situações nas quais as variações com o tempo são suficientemente lentas para que se possa ignorar os efeitos transitórios. Considera-se, também, que o sistema é equilibrado, ou seja, que os valores de grandezas elétricas são idênticos para todas as fases (exceto, é claro, pelo defasamento entre as tensões de cada fase). Isto permite representar o sistema em seu modelo unifilar. Além disso, assume-se que os elementos passivos do sistema são representados por parâmetros concentrados.

A análise propiciada pelo fluxo de potência é essencial para o estudo de sistemas de potência e forma a base para outras avaliações, como o estudo de curto circuito. O fluxo de potência desempenha um papel central no planejamento de expansão de sistemas elétricos de potência. Além disso, ela é empregada em diversos estudos realizados durante a operação do sistema de potência.

3.3.1 – Matriz de Admitância

Para desenvolver o modelo matemático do fluxo de potência da rede, deve-se obter a *Matriz de Admitância* das barras do sistema, que compõe uma matriz $n \times n$, onde n é o número de barras no sistema, construída a partir das admitâncias dos elementos do circuito equivalente aos segmentos que compõem a rede elétrica. Muitos dos segmentos são representados por uma combinação de elementos *shunt* (conectados entre um barramento e o nodo de referência) e elementos em série (conectados entre dois barramentos do sistema). A formulação da matriz segue duas regras simples (Halwary, 2000):

- A admitância dos elementos *shunt* conectados entre o nó k e a referência é adicionada à posição (k,k) da matriz;

- A admitância de um elemento conectado entre os nodos j e k adicionada às posições (j,j) e (k,k) da matriz. O negativo desta admitância é adicionado às posições (j,k) e (k,j) da matriz.

3.3.2 – Classificação das Barras

Cada barra possui um conjunto de quatro quantidades de interesse, a potência real (P), a potência reativa (Q), a tensão (V) e o ângulo desta tensão (A). Em cada barra do sistema, duas destas variáveis serão especificadas e duas permanecerão como incógnitas. Sendo estas classificadas de acordo com essas especificações como:

- Barra de Referência;
- Barra de Carga (PQ);
- Barra de Tensão Controlada (PV).

3.3.2.1 – Barra de Referência

Na barra de referência são especificadas a tensão (V) e o ângulo (A), entretanto as potências ativas e reativas são desconhecidas. Para a escolha desta barra deve ser considerado um que possua fontes de potência ativa e reativa, visto que esta deve ser capaz ter capacidade de suprir os valores de potência ativa e reativa determinadas pelo fluxo de potência. Geralmente para este tipo de barra seleciona-se uma unidade geradora de grande capacidade (Halwary, 2000).

3.3.2.2 – Barra de Carga (PQ)

Neste tipo de barra são especificados os valores de potência ativa e reativa. Utilizada geralmente na representação de cargas.

3.3.2.3 – Barra de Tensão Controlada (PV)

Nesse tipo de barra os valores de tensão e de potência ativa são especificados, e a potência reativa será a variável calculada pelo fluxo de potência.

3.3.3 – Fluxo de Potência Generalizado

Para o caso genérico de n barras, pode-se empregar as equações da matriz de admitâncias e a equação 3.2. Esta equação define a equação básica do fluxo de potência, sendo empregada diretamente para todas as barras de carga, e após as devidas manipulações, para as barras de tensão controlada.

$$P_i - jQ_i = (V_i^*) \sum_{j=1}^n (Y_{ij} V_j) \quad (3.2)$$

De posse de um modelo matemático, pode-se seguir determinando as tensões do sistema, resolvendo-se a equação 3.2, para $n-1$ barras do sistema (excluindo-se a barra de referência), após as manipulações necessárias (de acordo com o tipo de cada barra). Uma vez que as tensões nas barras sejam conhecidas, procede-se com a barra de referência.

Após estas primeiras soluções, pode-se proceder com a solução das potências e dos ângulos desconhecidos. Com estas novas soluções, resta apenas determinar o fluxo de potência entre as barras e as perdas de todo o sistema.

3.3.4 – Métodos de Solução

A solução das equações não-lineares simultâneas de um fluxo de potência requer o uso de técnicas iterativas, mesmo para os sistemas mais simples. Apesar de existirem diversas técnicas de solução de equações não-lineares, serão apresentados apenas os métodos de Newton-Raphson e Método Desacoplado Rápido.

3.3.4.1 – Método de Newton-Raphson

Este método tem sido amplamente utilizado na solução de equações não-lineares em diversas áreas. E será descrito nesta seção com um conjunto de equações genéricas, por questões de simplicidade, podendo ser estendido para um número arbitrário de equações.

Considere as equações não lineares 3.3 e 3.4(Halwary, 2000).

$$f_1(x_1, x_2) = k_1 \quad (3.3)$$

$$f_2(x_1, x_2) = k_2 \quad (3.4)$$

Considerando $x_1^{(0)}$ e $x_2^{(0)}$ como soluções inexatas para as equações 3.3 e 3.4 e $\Delta x_1^{(0)}$ e $\Delta x_2^{(0)}$ como correções necessárias para $x_1^{(0)}$ e $x_2^{(0)}$ tornem-se soluções exatas. Assim, as equações $x_1^{(0)}$ e $x_2^{(0)}$ podem ser reescritas:

$$f_1(x_1^{(0)} + \Delta x_1^{(0)}, x_2^{(0)} + \Delta x_2^{(0)}) = k_1 \quad (3.5)$$

$$f_2(x_1^{(0)} + \Delta x_1^{(0)}, x_2^{(0)} + \Delta x_2^{(0)}) = k_2 \quad (3.6)$$

Expandindo-se estas equações em série de Taylor em torno das estimativas:

$$f_1(x_1^{(0)}, x_2^{(0)}) + \left. \frac{\partial f_1}{\partial x_1} \right|^{(0)} \Delta x_1^{(0)} + \left. \frac{\partial f_1}{\partial x_2} \right|^{(0)} \Delta x_2^{(0)} + T.O.S. = k_1 \quad (3.7)$$

$$f_2(x_1^{(0)}, x_2^{(0)}) + \left. \frac{\partial f_2}{\partial x_1} \right|^{(0)} \Delta x_1^{(0)} + \left. \frac{\partial f_2}{\partial x_2} \right|^{(0)} \Delta x_2^{(0)} + T.O.S. = k_2 \quad (3.8)$$

onde o índice ⁽⁰⁾ nas derivadas parciais indica a avaliação desta com os valores das estimativas iniciais $x_1^{(0)}$ e $x_2^{(0)}$, e T.O.S. indica os termos de ordem superior, não expandidos.

Se as estimativas iniciais forem tais que $\Delta x_1^{(0)}$ e $\Delta x_2^{(0)}$ sejam pequenos, podemos ignorar os termos de ordem superior e re-escrever as equação 3.7 e 3.8 em forma matricial como:

$$\begin{bmatrix} \left. \frac{\partial f_1}{\partial x_1} \right|^{(0)} & \left. \frac{\partial f_1}{\partial x_2} \right|^{(0)} \\ \left. \frac{\partial f_2}{\partial x_1} \right|^{(0)} & \left. \frac{\partial f_2}{\partial x_2} \right|^{(0)} \end{bmatrix} \begin{bmatrix} \Delta x_1^{(0)} \\ \Delta x_2^{(0)} \end{bmatrix} = \begin{bmatrix} k_1 - f_1(x_1^{(0)}, x_2^{(0)}) \\ k_2 - f_2(x_1^{(0)}, x_2^{(0)}) \end{bmatrix} \quad (3.9)$$

A matriz de derivadas parciais da equação 3.9 é conhecida como *matriz jacobiana* e deve ser avaliada com os valores das estimativas iniciais. Multiplicando-se ambos os lados desta equação pela inversa da *matriz jacobiana* obtém-se uma aproximação das correções $\Delta x_1^{(0)}$ e $\Delta x_2^{(0)}$. Como os termos de ordem superiores foram desprezados, a simples adição destas correções às estimativas iniciais ainda não produzirá soluções exatas, mas estimativas bem melhores. Este procedimento pode ser repetido até obter-se estimativas que atendam às tolerâncias desejadas (Mendel, 2006).

Sendo assim, termos corrigidos para l -ésima iteração são dados pela equação 3.10 e a estimativa de solução inicial é atualizada com os resultados da equação 3.11.

$$\begin{bmatrix} \Delta x_1^{(l)} \\ \Delta x_2^{(l)} \end{bmatrix} = \begin{bmatrix} \left. \frac{\partial f_1}{\partial x_1} \right|^{(l)} & \left. \frac{\partial f_1}{\partial x_2} \right|^{(l)} \\ \left. \frac{\partial f_2}{\partial x_1} \right|^{(l)} & \left. \frac{\partial f_2}{\partial x_2} \right|^{(l)} \end{bmatrix}^{-1} \begin{bmatrix} k_1 - f_1(x_1^{(l)}, x_2^{(l)}) \\ k_2 - f_2(x_1^{(l)}, x_2^{(l)}) \end{bmatrix} \quad (3.10)$$

$$x^{(l+1)} = x^{(l)} + \Delta x^{(l)} \quad (3.11)$$

A solução do conjunto original de equações não-lineares foi substituída pela solução repetida de um sistema de equações lineares. Esta solução requer a avaliação da *matriz jacobiana*, com os valores das estimativas atuais, a cada iteração.

As equações do fluxo de potência podem ser inseridos no método de Newton-Raphson pela separação destas equações em suas partes real e imaginária, e considerando os valores de tensão(V) e ângulo(A) como variáveis desconhecidas. Re-escrevendo a equação 3.10 especificamente para fluxo de potência, tem-se a equação 3.12:

$$\begin{bmatrix} \Delta \underline{A}^{(l)} \\ \Delta \underline{V}^{(l)} \end{bmatrix} = \begin{bmatrix} \frac{\partial \underline{P}}{\partial \underline{A}}^{(l)} & \frac{\partial \underline{P}}{\partial \underline{V}}^{(l)} \\ \frac{\partial \underline{Q}}{\partial \underline{A}}^{(l)} & \frac{\partial \underline{Q}}{\partial \underline{V}}^{(l)} \end{bmatrix}^{-1} \begin{bmatrix} \underline{P}_{prev} - \underline{P}^{(l)} \\ \underline{Q}_{prev} - \underline{Q}^{(l)} \end{bmatrix} \quad (3.12)$$

onde as barras nos valores das variáveis da equação 3.12 indicam vetores, estendendo o caso de duas equações visto como exemplo para o caso geral da equação de potência. Os valores indicados com índice *prev* representam as previsões de potência ativa e reativa, injetadas no sistema. Os valores de $\underline{P}^{(l)}$ e $\underline{Q}^{(l)}$ representam os valores de injeções de potência ativa e reativa, calculadas com base no modelo do sistema e nas *l-ésimas* estimativas de tensões e ângulos do método.

A cada iteração do método, as estimativas de tensão e ângulos são atualizadas, a *matriz jacobiana* é re-avaliada, e as variações entre os valores previstos e os calculados novamente medidas. As iterações do método são executadas até que as estimativas de solução atinjam uma tolerância aceitável ou sejam executados um dado número de máximo de iterações.

Uma vez que se tenha obtido uma solução adequada, pode-se prosseguir com o calculo das injeções de potências reativas nas barras P-V e da injeção de potência complexa na barra de referência.

3.3.4.2 – Método Desacoplado Rápido

O método desacoplado rápido simplifica o de Newton-Raphson, explorando as fortes relações existente entre a potência real(P) e o ângulo (A) e entre a potência reativa(Q) e a tensão (V), relações comumente verificadas em sistemas de alta tensão reais (referenciar), Nestes sistemas, o fluxo de potência ativa é muito mais sensível as mudança no ângulos de fase do que as mudanças na magnitude das tensões. Da mesma forma, o fluxo de potência reativa é muito mais sensível às mudanças nas tensões do que nos ângulos. A *matriz jacobiana* simplificada, pois as derivadas parciais de P em relação à V e de Q em relação à A se aproximam de zero, e se re-escreve a equação 3.12 como:

$$\Delta \underline{A}^{(l)} = [B']^{-1} [\underline{P}_{prev} - \underline{P}^{(l)}] \quad (3.13)$$

$$\Delta \underline{V}^{(l)} = [B'']^{-1} [\underline{Q}_{prev} - \underline{Q}^{(l)}] \quad (3.14)$$

onde B' é uma aproximação da matriz de derivadas parciais de P em relação a A e B'' , uma aproximação da matriz de derivadas parciais de Q em relação a V . B' e B'' são, normalmente, mantida constantes durante as iterações, eliminando a necessidade de se atualizar a *matriz jacobiana* a cada iteração (necessária no método de Newton-Raphson).

Mesmo com as diversas aproximações realizadas nesse método, ele apresenta boas propriedades de convergência. O método *desacoplado rápido* passou largamente difundido, por ser computacionalmente menos custoso do que o de Newton-Raphson.

3.4 – Considerações Finais

Neste capítulo foram apresentados alguns conceitos básicos sobre análise de sistemas de potência e ainda uma breve introdução sobre o problema do fluxo de carga, os quais são essenciais para o correto entendimento das funcionalidades do aplicativo descrito nesta dissertação.

CAPITULO 4 – Metodologia de Otimização

4. 1 - Introdução

Neste capítulo serão considerados aspectos referentes à estrutura do algoritmo genético utilizado na aplicação, considerando aspectos como a representação cromossômica, método de seleção, operadores genéticos, entre outros. Detalhes sobre a interação do algoritmo genético e o programa de fluxo de carga também são o foco deste capítulo, onde uma seção inteira será dedicada para tal discussão. A função objetivo e as restrições do problema também fazem parte do escopo deste, e serão detalhadamente explicados no decorrer deste capítulo.

4.2 – Visão Geral da Metodologia de Otimização

Com intuito de atender os objetivos do aplicativo, foi desenvolvido um sistema híbrido de otimização, que se compõe de um algoritmo genético e de um programa de cálculo de fluxo de carga. Essas duas ferramentas integradas trabalham de forma cíclica até que o critério de parada do algoritmo seja atendido, como pode ser observado no fluxograma da figura 4.1

O processo iterativo inicia com a definição de uma população inicial para o algoritmo, gerada de forma aleatória. O tamanho da população é fixo durante todo o processo iterativo, podendo ser alterado antes do início das iterações, o qual deve ser realizado através de um arquivo de configuração, ver Anexo A.

A população inicial, assim como as populações que serão geradas no decorrer do processo evolutivo do algoritmo genético, correspondem ao conjunto de possíveis soluções para problema, representando os valores de reativos a serem injetados pelos dispositivos shunts no sistema.

Cada população gera um conjunto de configurações (indivíduos), que devem ser analisadas individualmente pelo o programa de fluxo de carga. A partir dos relatórios gerados

pelo programa de fluxo de carga, são extraídos um conjunto de dados, que servem como base para o cálculo do *fitness* de cada indivíduo.

O critério de parada para o processo iterativo é definido pelo número de gerações, que é previamente definido através do arquivo de configuração, ver Anexo A.

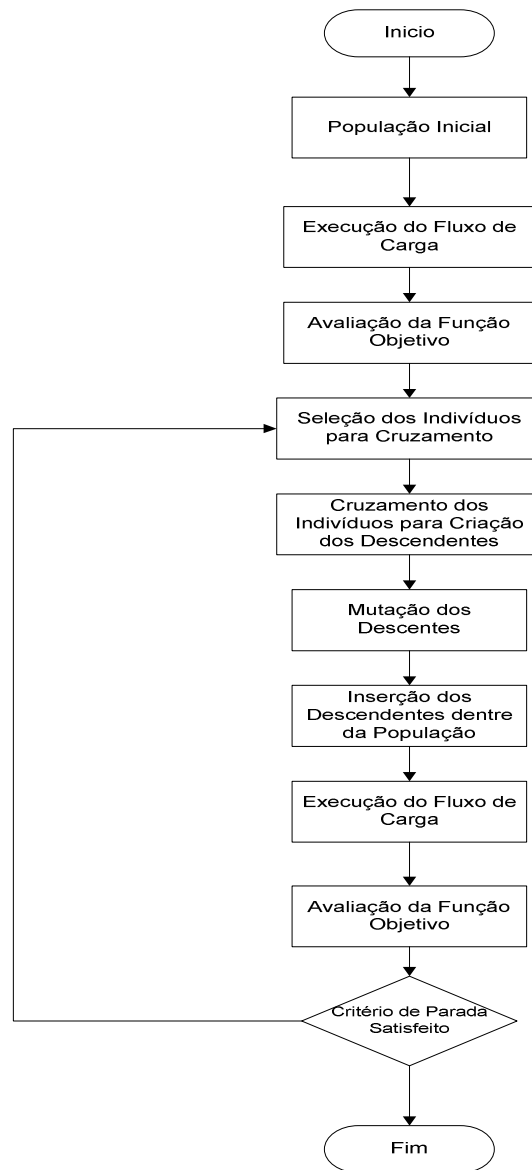


Figura 4.1 – Fluxograma Algoritmo

4.3 – Representação do *Cromossomo*

Para se obter o sucesso na busca da solução otimizada pelo processo evolutivo do AG, deve-se representar o objeto do estudo através de uma maneira viável de ser tratada pela máquina, com intuito de aumentar a acurácia das soluções obtidas pelo método. O uso de uma representação não adequada ao problema pode vir a gerar soluções errôneas ou não otimizados.

Considerando-se estes fatores, o cromossomo desenvolvido para este algoritmo foi projetado de acordo com os seguintes requisitos:

- Representar todos os dispositivos *shunts* selecionados para monitoramento,
- Considerar possíveis variações na quantidade de dispositivos *shunts* monitorados;
- Representar os valores de reativos dos *shunts* selecionados, dentro dos limites estabelecidos pelo modelo ou pelo usuário;

Tendo em vista os requisitos citados acima, a seguinte metodologia foi desenvolvida:

- Mapeia-se os N *shunts* selecionados para o monitoramento, com seus respectivos reativos, máximo e mínimos, como pode ser observado na tabela 4.1.

Tabela 4.1 – Mapeamento de *shunts*

Barras	N da Barras	Susceptancia	Máximo	Mínimo
1	Barra ₁	$Sh. 1$	Max_1	Min_1
2	Barra ₂	$Sh. 2$	Max_2	Min_2
3	Barra ₃	$Sh. 3$	Max_3	Min_3
...
N	Barra _{N}	$Sh. N$	Max_N	Min_N

- Após o mapeamento, cria-se um vetor, ver equação 4.1, onde cada posição ' i ' deste vetor corresponde a um *shunt* monitorado, e o conteúdo representa o valor da susceptancia do respectivo *shunt*. Esta metodologia equivale a construir um *cromossomo* com formato apresentado na figura 4.2, a seguir.

$$Sh = Sh[i], \text{ onde } \begin{cases} i = 1, 2, 3, \dots, N \\ Min_i \leq Sh[i] \leq Max_i \end{cases} \quad (4.1)$$

Sh ₁	Sh ₂	Sh ₃	Sh ₄	Sh ₅	Sh ₆	...	Sh _n
-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----	-----------------

Figura 4.2 - Representação do Cromossomo

Onde:

Sh_i : Representa o valor de reativo no *shunt*, onde $i = 0, 1, 2, 3, \dots, n$

N : Número total de Dispositivos *Shunt*.

4.4 – Interação com o Fluxo de Carga

Cada indivíduo gerado pelo processo evolutivo do algoritmo genético deve ser avaliado de forma individual, o que reflete na necessidade de executar o programa de fluxo de carga diversas vezes em apenas uma única geração ou iteração do algoritmo.

Tendo este requisito como necessário, uma atenção especial foi dada para este processo, e assim uma rotina para gerenciamento deste foi desenvolvida, ver figura 4.3. Esta tem como funções:

- Criar cópias dos arquivos de entrada de dados originais, com o intuito de manter a integridade do sistema original;
- Modificar cópias de arquivos de dados, de acordo com as necessidades do algoritmo;
- Executar programa de fluxo de carga, utilizando as versões modificadas do arquivo de entrada de dados original;
- Extrair dados de interesse dos relatórios de saída do programa de fluxo de carga;
- Entregar para a função objetivo as entradas necessárias para a avaliação dos indivíduos;

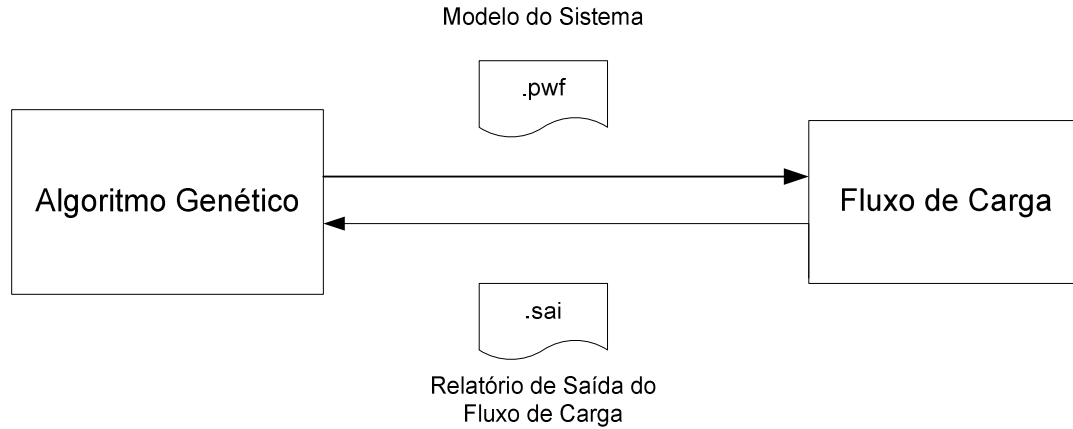


Figura 4.3 – Interação com o Fluxo de Carga

A quantidade de execuções do programa de fluxo de potência em uma única geração está diretamente relacionada com o tamanho da população adotada. Considerando-se estes aspectos pode-se equacionar a quantidade total de execuções da seguinte forma:

$$N_{exec} = N_{pop} \times N_{ger} \quad (4.2)$$

Onde:

- N_{exec} é o número total de execuções do fluxo de carga, ou fluxo de potência;
- N_{pop} é o número de indivíduos na população, lembrar que este valor não varia durante o processo evolutivo;
- N_{ger} é o numero de gerações.

Analisando a equação 4.1, observa-se que para pequenos valores de tamanho de população e número de gerações obtêm-se um grande número de execuções do programa de fluxo de carga, o que a principio pode representar um problema em virtude do tempo de processamento, mas que, no entanto devido a capacidade de processamento dos processadores atuais, esse tempo se mostrou aceitável durante os testes realizados, sendo este diretamente relacionado com o tamanho do sistema utilizado, e dos valores das variáveis N_{pop} e N_{ger} .

4.5 – Detalhes do Algoritmo Genético Utilizado

O algoritmo genético utilizado nesta aplicação é o *steady-state*, que tem como característica manter indivíduos de populações anteriores nas próximas populações, substituindo os piores indivíduos pelos novos gerados através do cruzamento.

Nas operações genéticas de *crossover* e mutação foram utilizadas as técnicas convencionais discutidas no capítulo 2, visto que não houve necessidade de inserção de nenhuma uma técnica especial para processamento dos indivíduos, e as técnicas convencionais geraram resultados suficientemente satisfatórios para o problema em questão.

As taxas de *crossover* e mutação podem ser definidas pelo usuário através de um arquivo de configuração, detalhado no Anexo A.

4.6 - Definição da Função Objetivo

A função objetivo idealizada para este aplicativo avalia o desvio de tensão, entre as tensões geradas nas barras monitoradas após os cálculos do programa de fluxo de carga, e as tensões especificadas para estas barras no modelo do sistema elétrico.

Os dados necessários para o cálculo desta função são providos basicamente por duas fontes, a primeira delas advém dos relatórios gerados pelo fluxo de carga, como já citado anteriormente, e a segunda pelo arquivo de entrada de dados original fornecido pelo usuário.

Dos relatórios gerados pelo fluxo de carga são extraídos os valores de tensão (V_i) das barras selecionadas pelo usuário para serem monitoradas. E dos arquivos de entrada de dados, são retirados as tensões especificadas (V_i^{spec}) e os valores de máximo (V_i^{max}) e mínimo (V_i^{min}) de tensão.

A função custo dos desvios de tensão foi formulada como descrita na equação 4.3, a seguir:

$$Custo\ Desvio\ de\ Tensão = \sum_{k=1}^n \left(\frac{V_i - V_i^{spec}}{\Delta V_i^{max}} \right)^2 \quad (4.3)$$

Onde:

- V_i : é a magnitude de tensão na barra i
- V_i^{spec} : é a magnitude da tensão especificada na barra i
- ΔV_i^{max} : é o máximo desvio de tensão na barra i

$$\Delta V_i^{max} = V_i^{max} - V_i^{min} \quad (4.4)$$

4.7 – Considerações Finais

Neste capítulo foi apresentado a metodologia criada para a solução do problema, aonde foram destacados aspectos como a interação das rotinas implementadas com o programa de fluxo de carga e os arquivos contendo os modelos do sistema elétrico a ser estudado. A função objetivo projetada para a solução do problema também foi detalhada, assim como a forma como os valores necessários para alimentação das variáveis de entrada desta função são extraídos.

CAPITULO 5 - Detalhamento do Software

5.1 – Introdução

Este capítulo tem como objetivo apresentar uma visão geral sobre o aplicativo desenvolvido, abordando a estrutura de software utilizada, assim como suas funcionalidades. Serão apresentadas ainda, de forma sucinta, as ferramentas e bibliotecas utilizadas, assim como um pequeno histórico de desenvolvimento do aplicativo.

5.2 – Ferramentas e Bibliotecas Utilizadas

O sistema foi implementado como um todo utilizando a linguagem C++, que foi adotada devido sua robustez e grande aplicação em problemas da mesma natureza, assim como o seu excelente desempenho, quando utilizada em aplicações que demandam alto processamento.

Para a implementação do algoritmo genético foi utilizada a biblioteca GALib : “A C++ Library of Genetic Algorithm Components” versão 2.4, produzida por Matthew Wall do *Massachusetts Institute of Technology* (Wall, 1996).

A GALib oferece uma estrutura de classes que implementa grande parte das atividades necessárias para o desenvolvimento de um algoritmo genético, desde a criação de genomas básicos ao controle da evolução de uma população (Wall, 1996).

Para construção da interface gráfica e para compilação foi utilizada a IDE (*Integrated Development Enviroment*) Code Gear RAD Studio 2009., que é uma tecnologia de compilação e construção de interfaces gráficas desenvolvida pela Borland para a linguagem C/C++ e Delphi.

Como programa de fluxo de carga foi utilizado o software ANAREDE, produzido pelo CEPEL (Centro de Pesquisas de Emergia Elétrica), que tem como objetivo o cálculo do estado operativo da rede elétrica para definidas condições de carga, geração, topologia e determinadas restrições operativas, permitindo a representação de diversos elementos do sistema, como

compensadores estáticos, compensadores série, geradores, modelos de cargas e elos de corrente contínua.

Na figura 5.1 são demonstrados de uma forma geral como os principais módulos do aplicativo funcionam, podendo-se observar que há uma interatividade entre estes, visto que o correto funcionamento de cada um depende das entradas e saídas geradas pelos outros, o que ficará mais evidente nas sessões seguintes.

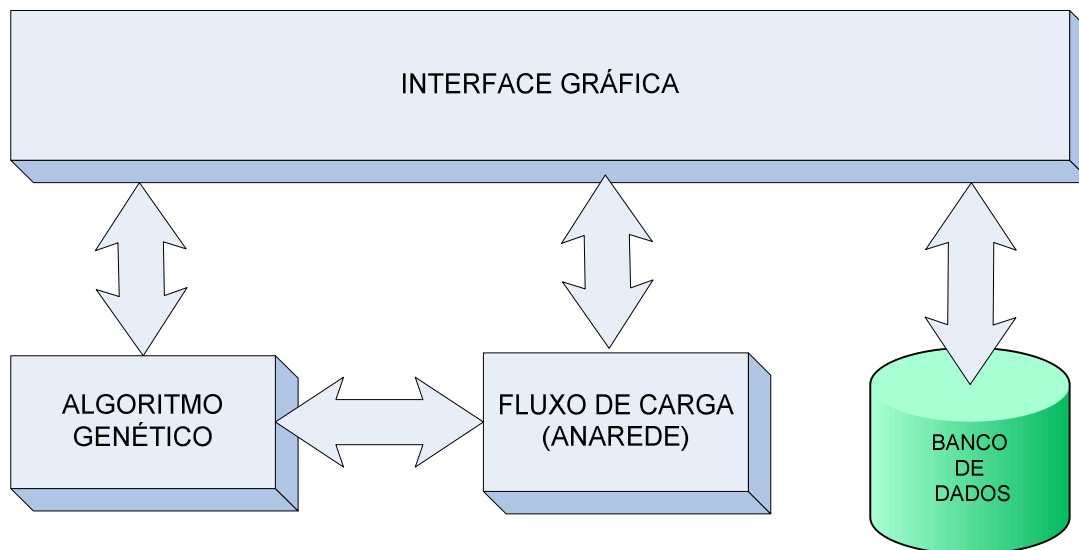


Figura 5.1 – Módulos do Aplicativo

5.3 – Histórico do Desenvolvimento

Como já citado anteriormente este software foi baseado em um projeto de P&D, desenvolvido através de uma parceria entre as Centrais Elétricas do Norte do Brasil SA (ELETRONORTE) e a Universidade Federal do Pará, sendo realizado especificamente pelo Grupo de Engenharia em Sistemas Elétricos e Instrumentação, que tinha como objetivo desenvolver uma ferramenta computacional que auxiliasse no planejamento da operação do sistema de transmissão da ELETRONORTE no Estado do Mato Grosso, com foco no planejamento ótimo do uso de fontes reativas do sistema (bancos de capacitores e reatores), de modo a obter condições de operação otimizadas.

O software resultante deste projeto corresponde em parte ao que está sendo apresentado neste trabalho, sendo que além da análise em regime permanente, são realizadas análises de transitórios eletromagnéticos, como pode ser observado em (Bezerra, 2008).

A primeira versão deste aplicativo foi desenvolvida utilizando a IDE (*Integrated Development Environment*) Borland C++ Builder 5.0, que foi suficiente para implementar as funcionalidades requeridas, mas no entanto esta possui um grande número de bibliotecas desatualizadas e que poderiam vir a gerar problemas de incompatibilidades com os sistemas operacionais mais novos, e conseqüentemente inutilizando o aplicativo. Com o objetivo, de superar este problema o aplicativo foi recompilado, utilizando uma versão mais atual do compilador C++, e as bibliotecas, assim como as rotinas, foram atualizadas, utilizando a IDE Code Gear RAD Studio 2009, que possui a versão mais recente do C++ Builder, até o presente momento.

5.4 – Arquitetura do Software em Termos Estruturais

Nesta seção é apresentada a arquitetura de software, ou a estrutura do sistema, de acordo com as representações dos elementos do sistema e suas relações.

5.4.1 – Estrutura

Organizando a estrutura do aplicativo em termos dos componentes do sistema e de suas relações, pode-se desenvolver um diagrama de componentes (Fowler, 2005), ver figura 5.2, apresentando o relacionamento entre os quatro principais componentes deste aplicativo, que são:

- “Banco de Dados”: Ver seção 5.5;
- “OCORE”: Este componente representa o arquivo executável do aplicativo, conglomerando todo o conjunto de interfaces gráficas para a interação com o usuário, assim como os métodos e rotinas necessárias para a aquisição e tratamento dos dados provindos do arquivo de entrada de dados, ver seção 5.4.

- “executaAG.dll”: Nesta, DLL (*Dynamic-link library*, ou Biblioteca de ligação dinâmica), estão reunidos os métodos, e as rotinas que implementam o algoritmo genético, utilizando como biblioteca base, a GALib, assim como a codificação da função objetivo.
- “GALib”: Pacote representando o conjunto de classes que envolvem a biblioteca GALib.

Analisando os relacionamentos entre os componentes do diagrama da figura 5.2., observa-se que há uma relativa independência entre os componentes, a qual foi desenvolvida de forma proposital, com o intuito de facilitar a manutenção e atualização do sistema. Com isso, alterações ou incrementações na função objetivo ou na estrutura do algoritmo genético poderiam ser realizadas, sem necessidade ou com um mínimo de adequação dos outros componentes.

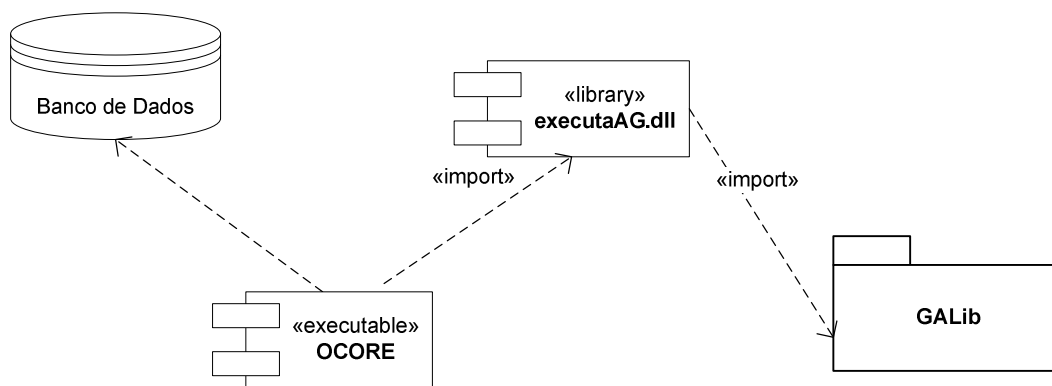


Figura 5.2 – Diagrama de Componentes

Dividindo a estrutura do sistema em termos de unidades de código ou em conjunto de códigos implementados, e organizando estes em módulos pode-se construir um diagrama de classes (Fowler, 2005), ver figura 5.3.

Neste diagrama de classes, o sistema foi dividido em quatro módulos, sendo dois pacotes e duas classes, que são detalhados a seguir:

- Interface Gráfica: Este pacote agrupa o conjunto de classes e formulários que compõe a Interface Gráfica do sistema.

- `executaAG`: Esta classe é responsável pela a execução ao algoritmo genético, assim como a execução do fluxo de carga.
- `alteraEntAnarede`: Esta classe é responsável por realizar as alterações na cópia do arquivo de entrada de dados utilizada no processo iterativo.
- `GAlib`: Este pacote representa a biblioteca `GAlib`, citada anteriormente, junto com todas as suas classes e relacionamentos.

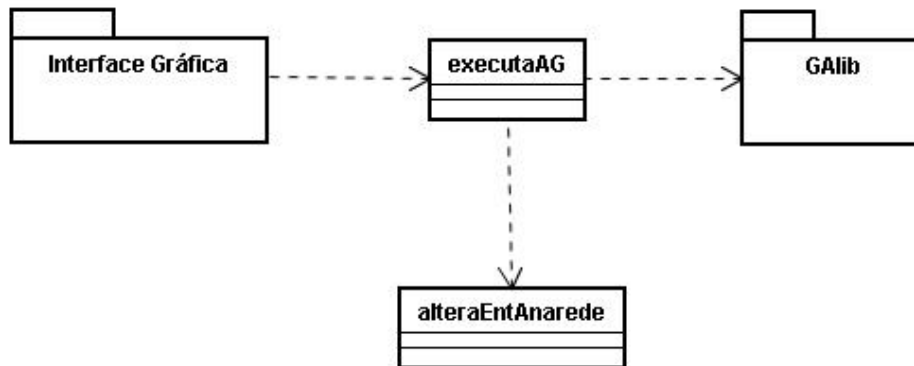


Figura 5.3 – Diagrama de Classe

5.4.2 – Interação com o usuário

A interação do aplicativo com o usuário é realizada através da interface gráfica, a qual foi desenvolvida de acordo com os seguintes requisitos funcionais:

- Carregar arquivo de entrada de dados;
- Definir parâmetros necessários para a otimização;
- Iniciar processo iterativo de otimização (algoritmo genético);
- Exibir resultados da otimização.

O diagrama de caso de uso apresentado na figura 5.4, apresenta um cenário mostrando as funcionalidades do aplicativo, onde observa-se que todos os casos de uso que estão diretamente ligados ao ator “usuário”, estão relacionados aos requisitos da interface gráfica do sistema. Nota-se também que o caso de uso “otimizar sistema” inclui dois dos casos de uso, que são o “Executar Algoritmo Genético” e o “Executar Fluxo de Carga”, que correspondem a funcionalidades do aplicativo que não são executadas diretamente pelo usuário, e sim pelo

aplicativo através processo iterativo de otimização, representado neste pelo caso de uso “Otimizar Sistema”.

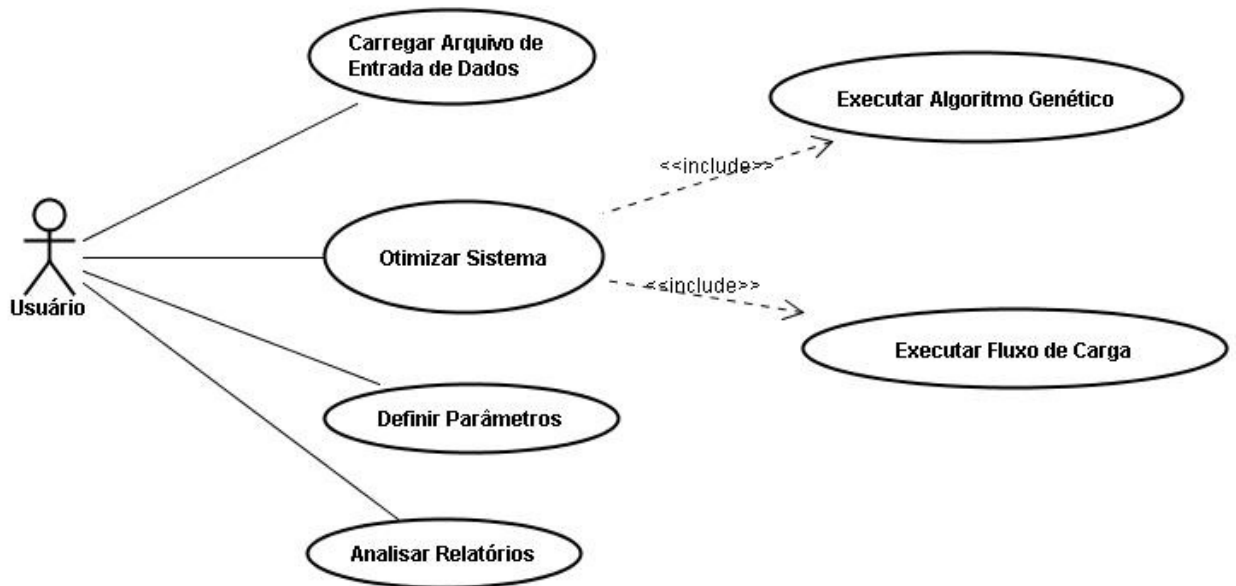


Figura 5.4 – Diagrama Caso de Uso

Na figura 5.5, é apresentado um diagrama de atividades, com o objetivo de demonstrar o fluxo de atividade a qual envolve a interação do usuário com o aplicativo. Observa-se que este diagrama é dividido em três regiões ou *swinlanes*, com objetivo de destacar as três principais etapas de funcionamento deste.

A primeira etapa tem basicamente as funções de receber do usuário o arquivo de entrada de dados com os dados referentes a modelagem do sistema elétrico, e definir os parâmetros necessários para iniciar a otimização. A partir desta, o sistema terá as informações de quais áreas do mesmo devem ser consideradas no processo iterativo, e qual barra ou barras destas áreas devem ser monitoradas, assim como os seus limites de tensão.

Outra definição realizada nesta etapa se refere a quais dos dispositivos *shunts*, pertencentes a área ou áreas definidas anteriormente deverão ser otimizados, e ainda os limites de reativos que devem ser considerados.

A segunda etapa corresponde à otimização do sistema, aonde nesta, a única interação com o usuário que existe é a inicialização do processo iterativo, visto que todas as informações necessárias para o início deste processo, foram definidas na *swinlane* anterior.

A terceira e ultima *swinlane*, corresponde a análise dos resultados, sendo estes apresentados na forma gráfica, através de gráficos comparativos entre as tensões nas barras monitoradas antes da otimização e depois desta, e ainda através do relatório de saída do programa de fluxo de carga.

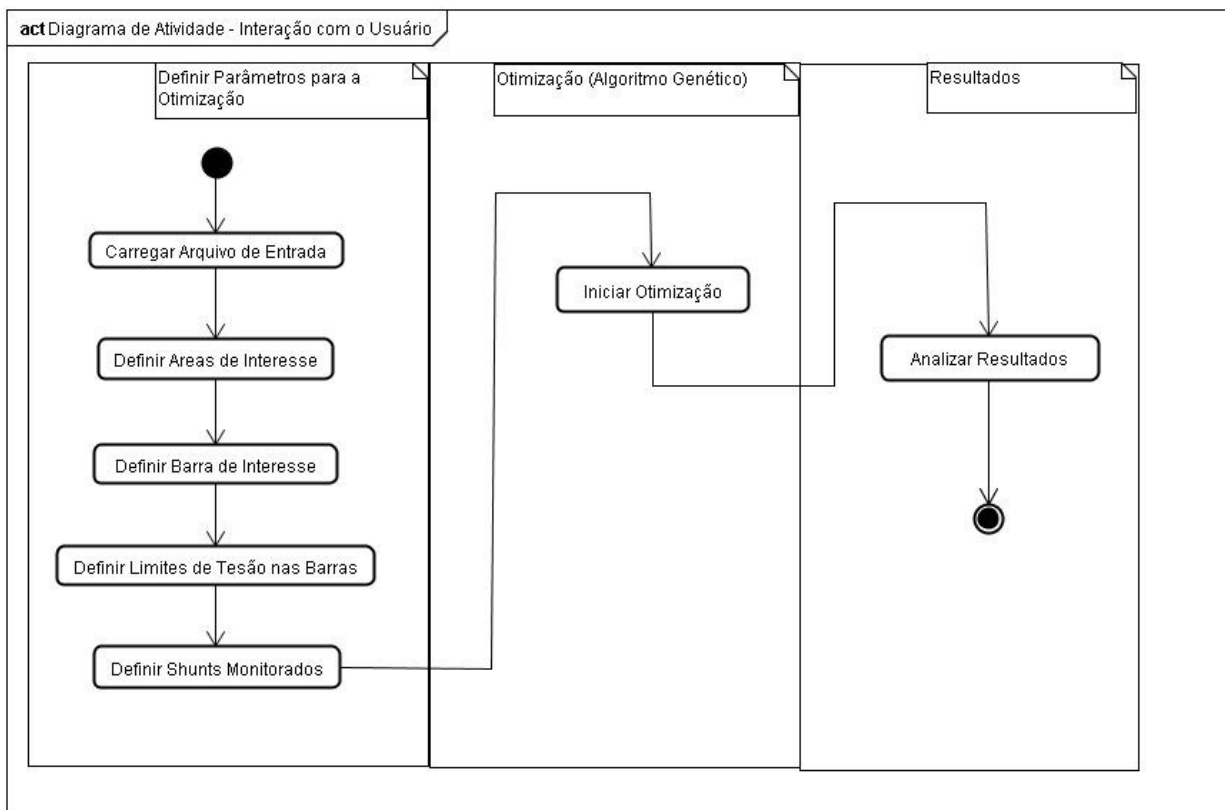


Figura 5.5 – Diagrama de Atividade

5.5 - Dados de Entrada

Com o intuito de tornar o aplicativo viável para a aplicação desejada, utilizou-se como padrão os modelos de arquivos de entrada do ANAREDE fornecidos como casos de referência pelo Operador Nacional do Sistema Elétrico (ONS), que servem como base para os estudos de fluxo potência realizados no âmbito do planejamento da operação elétrica do Sistema Interligado

Nacional (SIN), estando estes disponíveis para livre download no site do mesmo (www.ons.org.br).

Estes arquivos de texto possuem a modelagem de todo o sistema elétrico estando incluída a topologia, o carregamento de rede e variáveis de controle associados a solução de problemas de fluxo de carga. Na figura 5.6 é apresentado um exemplo de um sistema modelado neste formato.

```

ITITU
ONS - MENSAL - DEZEMBRO 2006 - PESADA
DOPC IMPR
(Op) E (Op) E (Op) E (Op) E (Op) E (Op) E (Op) E (Op) E
QLIM L VLIM L CREM L CTAP L STEP L NEWT L RCVG L RMON L FILE L CELO L
9999
ULOG
4
C:\\CORE\\arquivos\\relatorio.sai
DCTE
(Mn) ( Val) (Mn) ( Val) (Mn) ( Val) (Mn) ( Val) (Mn) ( Val) (Mn) ( Val)
BASE 100. DASE 100. TEPA 1. EXST 4. TETP 5. TBPA 5.
TLPP 1. TEPR 1. QLST 4. TLPR 1. TLPQ 2. TSBZ .01
TSBA 5. ASTP .05 VSTP 5. TLVC .1 TLTC .01 TSPR .1E-7
ZMAX 500. TLPV .5 VDM 200. VDMN 40. TUDC .001 TADC .01
PGER 30. TPST 2. VFLD 70. ZMIN .001 HIST 470 LFIT 10
ACIT 90 LFCV 1 DCIT 10 VSIT 10 LPIT 50 LFLP 10
PDIT 10 LCRT 24 LPRT 60 CSTP 500.
ICIT 30 DMAX 5 FDIV 2. ICMN .05 VART 5. TSTP 32
ICMV 1. APAS 90. CPAR 70. VAVT 2. VAVF 5. VMVF 15.
VPVT 2. VPFV 5. VPMF 10. VSVF 20. VINP 1. VSUP 1.
9999
DBAR
(No) O TB( nome )G( v)( A)( Pg)( Qg)( Qn)( Qm)( Bc)( P1)( q1)( Sh)(A(vf)
10 1WANGRA-1--1GR8 980-41. 520. 6.31-211. 414. 28.5 16.9 441000
11 1WANGRA-2--1GR8 980-39.1350.55.04-600. 600. 67.9 47.1 441000
12 1WLCBARRET-6GR81040-4.91100.296.7-540. 420. 11000
14 1WFUNIL-1--1GR81000-53. 29.-4.87 -50. 50. 11000
15 1WFUNIL-2--2GR81030-67. 59.69.11-100. 100. 11030
16 1WFURNAS--8GR81050-14.1200.359.2-720. 480. 11000
17 1 FURNAS--0008 950-20. 0. 0. 0. 0. 11000
18 2WITUMBIAR-6GR8103025.22000.216.1-9999 9999 451000
20 1WMARIMBON-5GR810508.631200.83.76-400. 400. 11000
21 1WMANSO---2GR8101036.6 85.3.844 -40. 42. 451000
22 1WM.MOR.A--5GR81020 -5. 242.17.91-100. 105. 11000
24 1WM.MOR.B--4GR8 970-11. 161.40.72 -24. 62. 11000
28 1WP.COLOMB-4GR81030-10. 328.53.67-100. 100. 11000
30 1WSCRUZ-19-00081000-66. 0. 0. 0. 0. 11000
31 1WSCRUZ-13-0008 950-66. 0. 0. 0. 0. 11000
35 1WCORUMBA--3GR8100028.7 375. -9.9-180. 180. 11000
36 1WS.MESA---3GR8100037.71050.-64.4-570. 570. 11000

```

Figura 5.6 – Arquivo de Entrada

Com a introdução destes dados no sistema, o aplicativo através de uma rotina interna, extrai as informações necessárias para a correta interação com usuário, que são:

- Dados de Barra;
- Dados de Linha;
- Limites de Tensão nas Barras;
- Limites de Tensão nos Shunts;
- Grupos de Limites de Tensão;

- Dados de Áreas.

As informações extraídas, por sua vez, são armazenadas em um banco de dados relacional, que é detalhadamente definido na próxima sessão, e utilizadas posteriormente para definir, junto com as informações fornecias pelo usuário através da interface gráfica, quais os pontos do sistema elétrico o algoritmo de otimização deve considerar, assim como, fornece as informações necessárias para a definição das restrições da função objetivo do algoritmo.

5.6 – Banco de Dados

Com o intuito de auxiliar este aplicativo a alcançar seus objetivos, foi desenvolvido um pequeno banco de dados relacional, que consiste em três tabelas relacionadas, que podem ser analisadas no diagrama entidade-relacionamento que se encontra na figura 5.7, a descrição detalhada das tabelas, campos relacionamentos é feita nas sub-sessões seguintes.

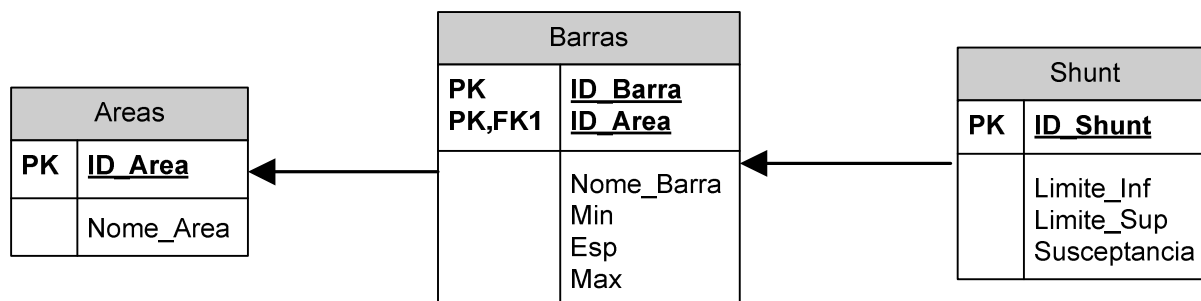


Figura 5.7 – Diagrama Entidade - Relacionamento

5.6.1 – Tabela “AREA”

Esta tabela tem como objetivo armazenar os dados referentes às áreas nas quais o sistema em estudo está dividido, estas por sua vez são mapeadas pelo aplicativo através do código de execução “DARE”, como pode ser observado na figura 5.8, e armazenadas nos referentes campos, que são definidos abaixo:

- ID_Area: Neste campo são armazenados os identificadores numéricos das áreas, que por serem únicos, foram escolhidos para representar a chave primária desta tabela;

- **Nome_Area:** Os nomes que identificam as áreas são armazenados neste campo, com o principal objetivo de facilitar a identificação da área por parte do usuário.

```

V .95 1.048
X .95 1.1
Z .95 1.046
99
DARE
(ar) (xchg) ( Identificacao da area ) (xmin) (xmax)
1 0. * FURNAS *
2 0. * CEMIG G/T *
3 0. * CEMIG D *
4 0. * CTEEP - GRANDE SAO PAULO *
5 0. * C T E E P *
6 0. * C T E E P - SECUNDARIA *
7 0. * CPFL *
8 0. * C E S P *
9 0. * LIGHT *
10 0. * CERJ *
11 0. * ESCELSA *
12 0. * CELG *
13 0. * CEB *
14 0. * AES TIETE *
15 0. * DUKE ENERGY - GP *
16 0. * CELTINS *
17 0. * ITAIPU *
18 0. * CATAGUAZES *
19 0. * TRACTEBEL ENERGIA - SUL *
20 0. * ELETROSUL *
21 0. * COPEL *
22 0. * COPEL - SECUNDARIA *

```

Figura 5.8 – Código de Execução “DARE”

5.6.2 – Tabela “BARRA”

Como tabela “AREA”, esta tem como objetivo armazenar as informações referentes as barras do sistemas, sendo estas mapeadas através do código de execução “DBAR” no arquivo de entrada, ver figura 5.9, e armazenadas nos campos referentes como descrito abaixo:

- **ID_Barra:** Neste campo são armazenadas os identificadores das barras, que assim como os identificadores das áreas, são únicos e então foram escolhidos para serem a chave primária da tabela;
- **ID_Area:** Este campo consta na tabela apenas para realizar o relacionamento entre esta tabela e a tabela “AREAS”, sendo assim este consiste de uma chave-estrangeira;
- **Nome_Barra:** O nome da barra é armazenado neste campo, e assim como na tabela “AREAS”, tem com objetivo facilitar a identificação da barra por parte do usuário.
- **Min:** Tensão mínima da barra, que é iniciada como padrão em 0.95
- **Esp:** Tensão especificada, valor que é fornecido pelo arquivo de entrada

- **Max:** Tensão máxima da barra, que é iniciada como padrão em 1.05, ou, no caso de a tensão especificada for superior a esta, o próprio valor da tensão especificada.

```

B2 - Notepad
File Edit Format View Help
PGER 30. TPST 2. VFLD 70. ZMIN .001 HIST 470 LFIT 10
ACIT 90 LFCV 1 DCIT 10 VSIT 10 LPIT 50 LFLP 10
PDIT 10 LCRT 24 LPRT 60 CSTP 500.
ICIT 30 DMAX 5 FDIV 2. ICMN .05 VART 5. TSTP 32
ICMV 1. APAS 90. CPAR 70. VAVT 2. VAVF 5. VMVF 15.
VPVT 2. VPMF 5. VPMF 10. VSVF 20. VINP 1. VSUP 1.
9999
DBAR
(NO) TB( nome )G( V)( A)( Pg)( Qg)( Qn)( Qm)( Bc)( P1)( Q1)( Sh)(A)(Vf)
10 1WANGRA-1--1GR8 980-41. 520. 6.31-211. 414. 28.5 16.9 441000
11 1WANGRA-2--1GR8 980-39.1350.55.04-600. 600. 67.9 47.1 441000
12 1WLCBARRET-6GR81040-4.91100.296.7-540. 420. 11000
14 1WFUNIL-1--1GR81000-53. 29.-4.87 -50. 50. 11000
15 1WFUNIL-2--2GR81030-67. 59.69.11-100. 100. 11030
16 1WFURNAS--8GR81050-14.1200.359.2-720. 480. 11000
17 1 FURNAS--0008 950-20. 0. 0. 0. 0. 11000
18 2WITUMBIAR-6GR8103025.22000.216.1-9999 9999 451000
20 1WMARIMBON-5GR810508.631200.83.76-400. 400. 11000
21 1WMANSO---2GR8101036.6 85.3.844 -40. 42. 451000
22 1WM.MOR.A--5GR81020 -5. 242.17.91-100. 105. 11000
24 1WM.MOR.B--4GR8 970-11. 161.40.72 -24. 62. 11000
28 1WP.COLOMB-4GR81030-10. 328.53.67-100. 100. 11000
30 1WSCRUZ-19-00081000-66. 0. 0. 0. 0. 11000
31 1WSCRUZ-13-0008 950-66. 0. 0. 0. 0. 11000
35 1WCORUMBA--3GR8100028.7 375. -9.9-180. 180. 11000
36 1WS.MESA---3GR8100037.71050.-64.4-570. 570. 11000
38 1WGRAJAU-2-1C581015-61. 0.13.16-120. 200. 178 1.3 0. 11000
39 1WB.GERAL2-0008101717.3 0. 0. 0. 0. 207 11000

```

Figura 5.9 – Código de Execução “DBAR”

5.6.3 – Tabela “SHUNT”

Esta tabela tem como objetivo armazenar as barras que possuem dispositivos shunts, as quais são identificadas pelo código de execução “DBAR” no arquivo de entrada, ver figura 5.10., além de armazenar informações necessárias para o processamento do algoritmo. Os campos referentes a esta tabela são descritos abaixo:

- **ID_Shunt:** Este campo armazena a identificação da barra em que o shunt está conectado, sendo este escolhido para ser chave-primária desta tabela.
- **Limite_Inf:** Neste campo são armazenadas as informações referentes ao limite inferior de tensão que deve ser considerado para shunts, e é iniciada com o valor ‘0’;
- **Susceptancia:** A susceptancia definida no modelo introduzido pelo arquivo de entrada é armazenada neste campo.
- **Limite_Sup:** Este campo é utilizado para armazenar o limite superior de tensão que deve ser considerado, e assim como o campo “Limite_Inf”, é de fundamental importância para

a definição das restrições do algoritmo. Este campo é iniciado com o mesmo valor do campo “Susceptancia”.

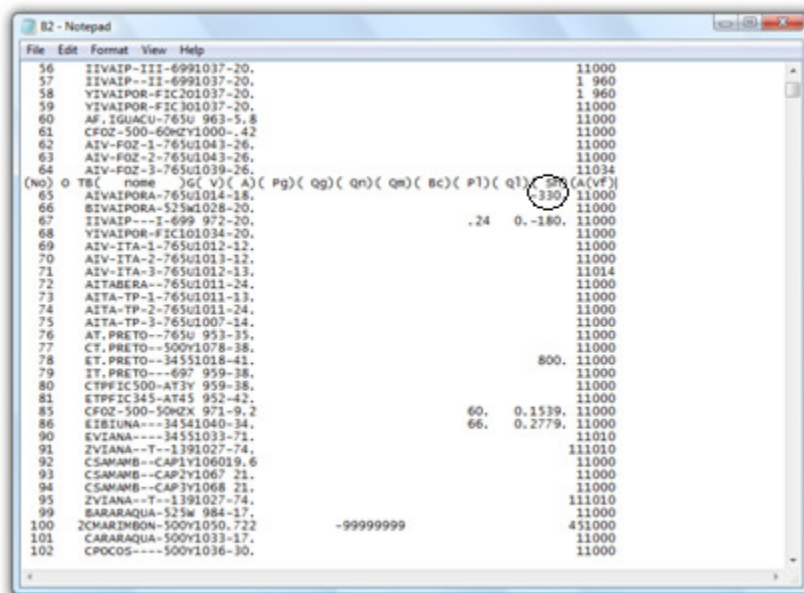


Figura 5.10 – Identificação do *Shunt*

5.7 – Considerações Finais

Neste capítulo foi apresentada e detalhada a estrutura de funcionamento do aplicativo, aonde pode ser observado as principais formas de relacionamento entre os componentes do sistema. Foi ainda detalhada a estrutura do banco de dados utilizado, assim como o relacionamento e a função de cada tabela que o compõe.

O formato do arquivo de entrada de dados necessário para o funcionamento correto do aplicativo também foi detalhado, sendo este um dos principais pontos a ser observado na utilização do sistema, visto que o fornecimento de dados incorretos ou foras dos padrões aceitáveis, podem vir a gerar resultados incorretos ou até mesmo o não funcionamento, visto que o algoritmo não será capaz de extrair os dados necessários para a realização do processo iterativo.

CAPITULO 6 – Estudo de Caso – Sistema Mato Grosso

6. 1 – Introdução

Este capítulo tem como objetivo demonstrar as funcionalidades do sistema desenvolvido, assim como comprovar a eficácia do algoritmo genético na solução de problemas de otimização de reativos em sistemas de transmissão de energia elétrica. Como base na otimização, foi utilizado um sistema real, brevemente descrito na próxima seção, com o intuito de comprovar que a ferramenta produzida pode ser amplamente utilizada pelo setor energético, e que algoritmo genético projetado obtém respostas satisfatórias mesmo exposto a situações não ideais. Com o intuito de demonstrar toda a dinâmica de interação com o usuário, será realizada uma análise passo – a – passo do software, que se inicia na definição do arquivo de entrada de dados a ser utilizado, até a análise gráfica dos resultados.

6. 2 – Sistema Base

O Sistema utilizado neste estudo é o SIN (Sistema Interligado Nacional), que é um sistema de coordenação e controle, formado pelas empresas das regiões Sul, Sudeste, Centro-Oeste, Nordeste e parte da região Norte, que congrega o sistema de produção e transmissão de energia elétrica do Brasil, o qual é um sistema hidrotérmico de grande porte, com predominância de usinas hidrelétricas e proprietários múltiplos, estatais e privados.

O SIN engloba praticamente toda a produção de eletricidade do país ficando de fora apenas 3,4%, de acordo com o ONS, que corresponde a pequenos sistemas isolados localizados na região amazônica, ver figura 6.1. Sendo assim, este sistema caracteriza uma excelente fonte de estudos, visto a sua complexidade e abrangência no território nacional.

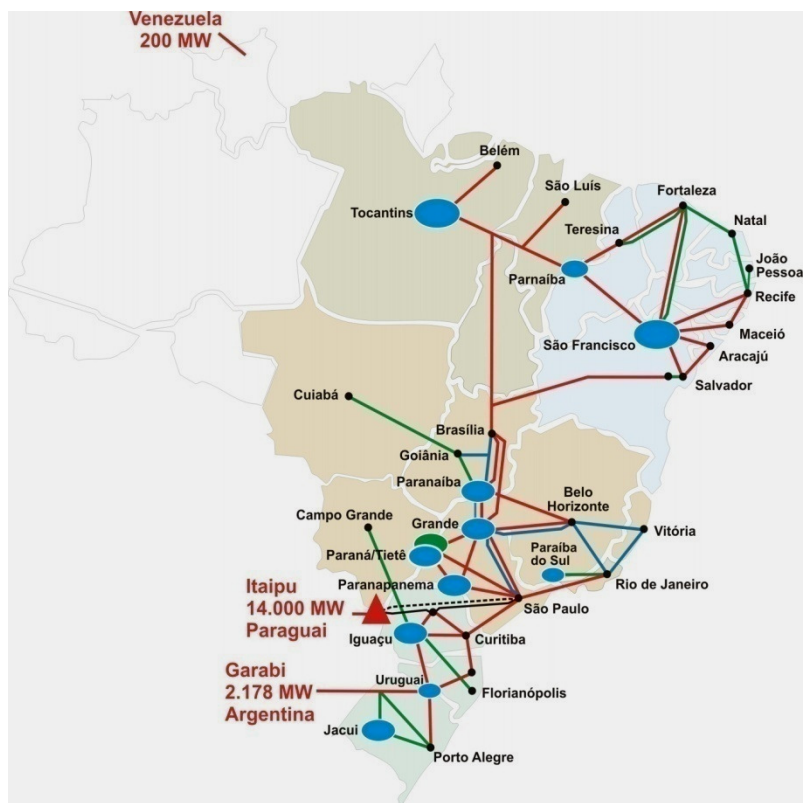


Figura 6.1 – Sistema Interligado Nacional

Para viabilizar o estudo, visto que o SIN possui milhares de barras, a área de estudo será delimitada às áreas que envolvem o sistema da Regional de Transmissão do Mato Grosso (CMT) das Centrais Elétricas do Norte do Brasil S/A (ELETRONORTE), que foi uma das financiadoras deste trabalho.

O sistema elétrico do estado de Mato Grosso, mostrado na figura 6.2, é constituído basicamente por um tronco radial principal em 230 kV que parte da subestação de Rio Verde, no estado de Goiás, chegando às subestações de Barra do Peixe, Rondonópolis, Cuiabá, Coxipó e Jauru. A área norte do estado é atendida por uma derivação em 230 kV, também radial, que parte da subestação Coxipó, atendendo as subestações de Nobres, Nova Mutum, Lucas do Rio Verde, Sorriso e Sinop. O circuito em 500 kV interliga as subestações de Cuiabá, Ribeirãozinho, Rio Verde Norte a Itumbiara, ponto de conexão com o SIN.

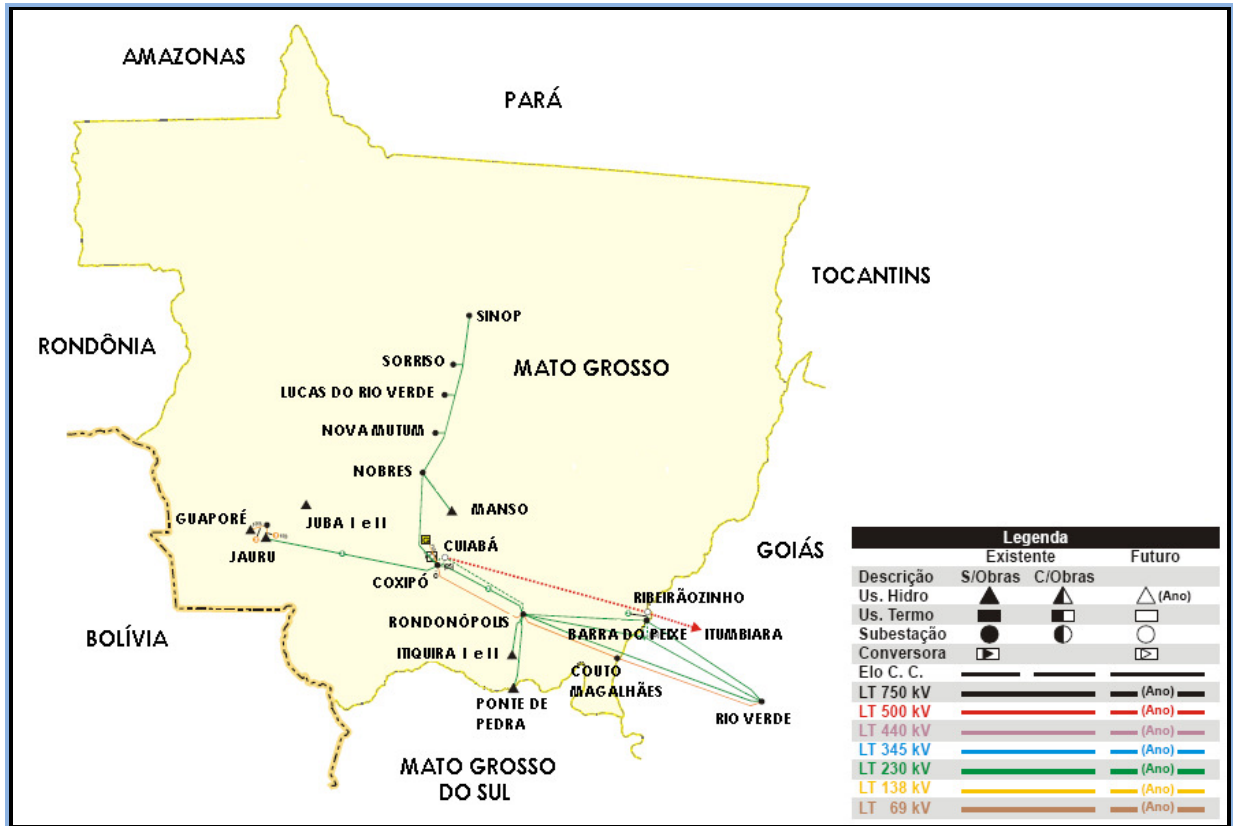


Figura 6.2 – Sistema ELN-MT

6.3 – Arquivo de Entrada de Dados

O arquivo de entrada de dados utilizado neste estudo de caso foi obtido diretamente do banco de dados do ONS (www.ons.org.br), e está no formato do programa ANAREDE.

Este arquivo foi desenvolvido pelo ONS em Dezembro de 2006, e foi modelado para o sistema operando em carga pesada.

A versão utilizada do ANAREDE foi a V08 (Agosto/2004).

6.4 – Parâmetros do Algoritmo Genético

Foram realizadas diversas execuções do algoritmo, e concluiu-se que utilizando os parâmetros apresentados na tabela 6.1, o aplicativo tem a capacidade de gerar resultados satisfatórios, atendendo os requisitos propostos para a elaboração deste.

TABELA 6.1 – PARÂMETROS DO AG

N_{Ger}	N_{pop}	Taxa de Mutação	Taxa de <i>Crossover</i>
200	15	0.02	0.8

Para este estudo de caso foram realizados 10 experimentos utilizando as mesmas entradas e os mesmos parâmetros. Os resultados obtidos foram bastante satisfatórios, observando-se que em todos os experimentos o algoritmo convergiu para soluções próximas.

Na figura 6.3 é apresentado o gráfico acumulado da evolução do melhor indivíduo dos 10 experimentos, e no Anexo D, são apresentados os gráficos individuais dos melhores indivíduos de todos os experimentos. Observa-se que todos estes convergem para uma faixa pequena de valores, que fica em torno de 12.763 e 13.0235, representando assim uma diferença de apenas 0.2605 entre o pior e o melhor resultado. Esta diferença prova a eficácia do algoritmo em relação a convergência para uma solução ótima, visto que não há uma significativa distância numérica entre as soluções otimizadas propostas pelo algoritmo.

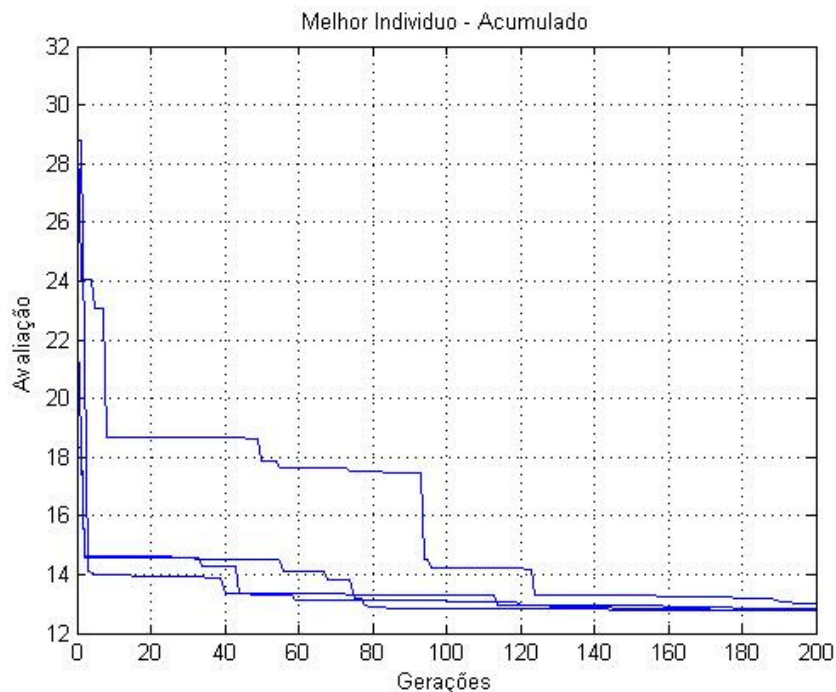


Figura 6.3 – Melhores Indivíduos de Todos os Experimentos

Dentre os experimentos realizados, foi selecionado o que possui a melhor avaliação para ser explorado neste capítulo, que neste caso foi o experimento '2'. Na figura 6.4, é exibida a evolução do melhor indivíduo deste experimento, e na próxima sessão será exibida a solução proposta por este.

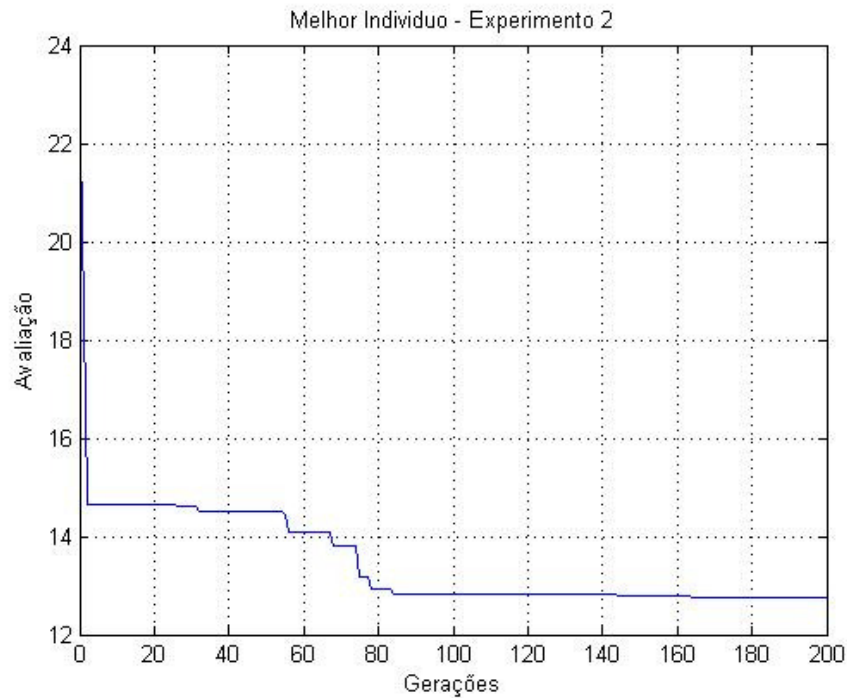


Figura 6.4 – Melhores Indivíduo

6.5 – Execução do Estudo de Caso

A análise do estudo de caso será realizada de acordo com a sequência de execução do programa, que por sua vez, está de acordo com o diagrama de atividades apresentado na figura 5.5, no capítulo 5.

A tela inicial do aplicativo é apresentada na figura 6.5 onde é disponibilizado para o usuário uma barra de ferramentas com duas opções, a primeira corresponde a um sub-menu que contém as seguintes opções:

- Execução de Fluxo de Carga: Essa opção disponibiliza para o usuário a possibilidade de realizar um estudo de fluxo de carga, utilizando o ANAREDE diretamente da interface;
- Execução da Otimização de Reativos: Selecionando esta opção o usuário irá iniciar o processo iterativo para a realização da otimização.

A segunda opção da barra de ferramentas disponibiliza acesso a algumas informações sobre o programa, dentre estas informações estão a autoria e manual de utilização.



Figura 6.5 – Interface Inicial

Selecionando-se a primeira opção da barra de ferramentas citada anteriormente, e a opção “Execução da Otimização de Reativos” do sub-menu correspondente, obtêm - se a interface de entrada para o arquivo texto contendo a modelagem do sistema (arquivo de entrada de dados).

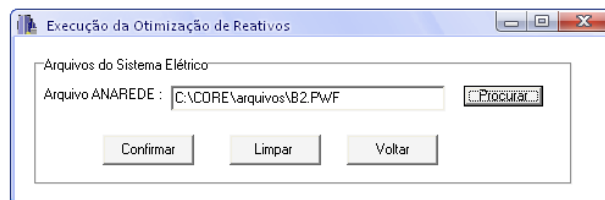


Figura 6.6 – Definir Arquivo de Entrada de Dados

O próximo passo da execução consiste em definir as áreas de interesse para a otimização. A interface da figura 6.7 é responsável por essa ação. Observa-se que esta exibe todas as áreas contidas no modelo, que neste caso são todas áreas contidas no SIN.

Para o estudo em questão, selecionamos apenas a área que envolve a regional de transmissão do Mato Grosso, que neste caso corresponde à área ‘45’.

Definidas a área de interesse, o próximo passo é definir quais das barras contidas nestas deveram ser monitoradas. Na figura 6.8 apresenta-se a interface responsável por essa atividade, observa-se que ela se assemelha bastante a interface utilizada para a definição das áreas, e obedecendo assim ao mesmo princípio de funcionamento.

Na tabela ao lado esquerda desta interface são exibidas todas as barras contidas nas áreas selecionadas, e na tabela ao lado direito são exibidas apenas as barras selecionadas para a monitoração pelo o algoritmo genético. Para o estudo em questão, foram selecionadas todas as barras pertencentes à área ‘45’.

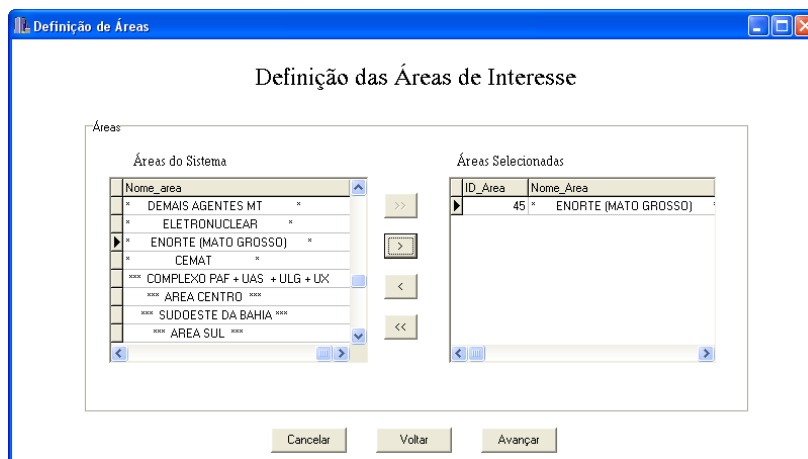


Figura 6.7 – Definição das Áreas de Interesse

Após a definição das barras a serem monitoradas, deve-se então definir os limites de tensão para estas barras, estes por definição são ajustados como ‘0.95’, para mínimo de tensão, e ‘1.05’ ou o próprio valor de tensão especificado, caso este seja superior a ‘1.05’, para o máximo

de tensão. Estes, por sua vez, podem ser alterados pelo usuário, assim como a tensão específica da barra, como pode ser observado na figura 6.9.

Para o estudo em questão, utilizou a configuração default, que pode ser observada no Anexo B, na figura B.1.

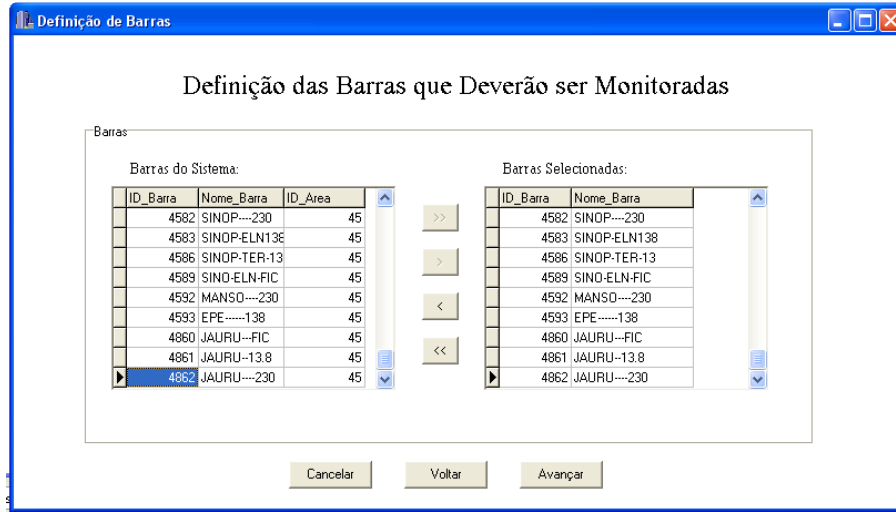


Figura 6.8 – Definição de Barras à serem Monitoradas

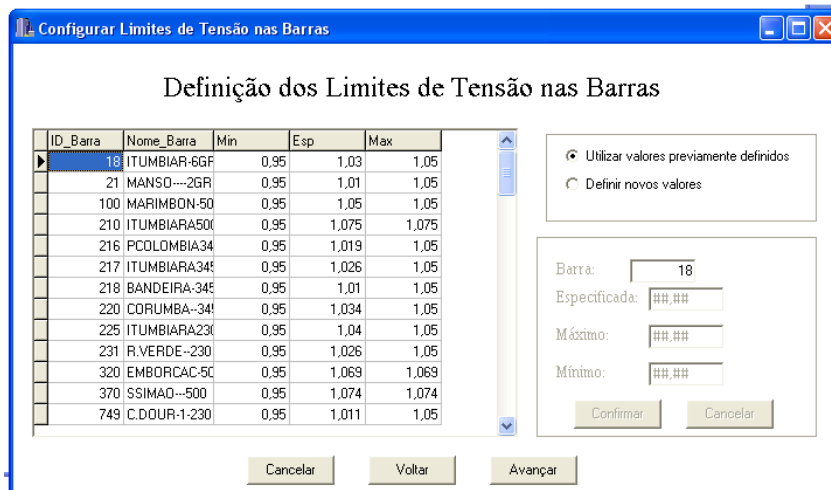


Figura 6.9 – Definição dos Limites de Tensão

A definição dos dispositivos shunts contidos nas áreas selecionadas é a próxima ação na execução do aplicativo. Esta é realizada através da interface da figura 6.10, e assim como nas interfaces para definição das áreas e das barras, a tabela ao lado esquerdo apresenta todos os dispositivos shunts contidos nas áreas selecionadas, e a tabela ao lado direito apresenta apenas os dispositivos shunts selecionados para serem monitorados.

No estudo em questão foram selecionados todos os dispositivos shunts a serem monitorados, com o intuito de fornecer um valor otimizado de reativos para todos dos dispositivos contidos no sistema Eletronorte – Mato Grosso.

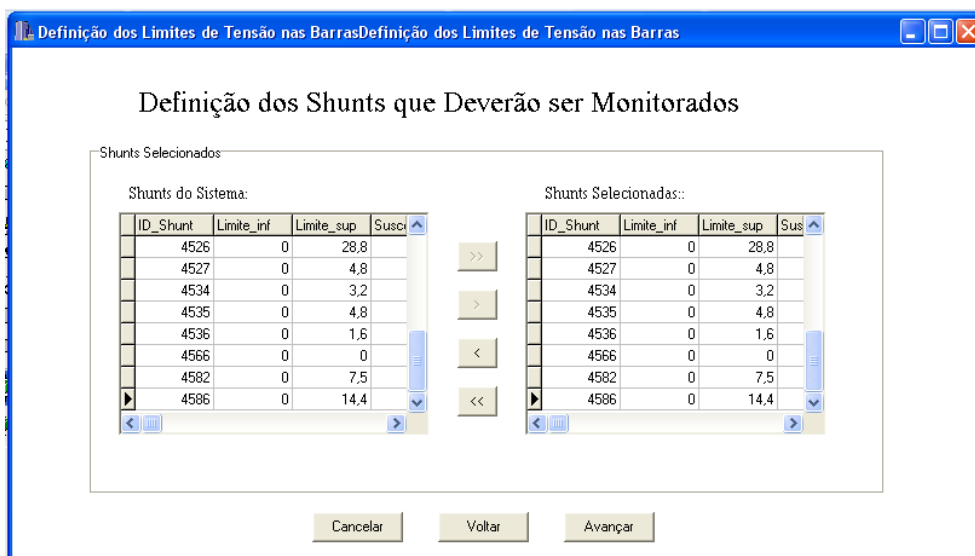


Figura 6.10 - Definição dos Shunts Monitorados

Assim como para os limites de tensão nas barras monitoradas, deve-se também configurar os limites de reativos para os dispositivos shunts. Esta ação é realizada através da interface apresentada na figura 6.11, a seguir que como pode-se observar, ela possui o mesmo princípio de funcionamento da interface de configuração dos limites de tensão, definida na figura 6.9.

Os valores dos limites, por definição neste programa, são configurados como '0', para mínimo, e o próprio valor da susceptancia, para máximo de reativo. Assim como no caso dos

limites de tensão, esses valores podem ser alterados pelo usuário para os valores que forem mais convenientes para o seu estudo.

Para o estudo em questão definiu-se que os valores de limites de reativos seriam mantidos com os valores de definição

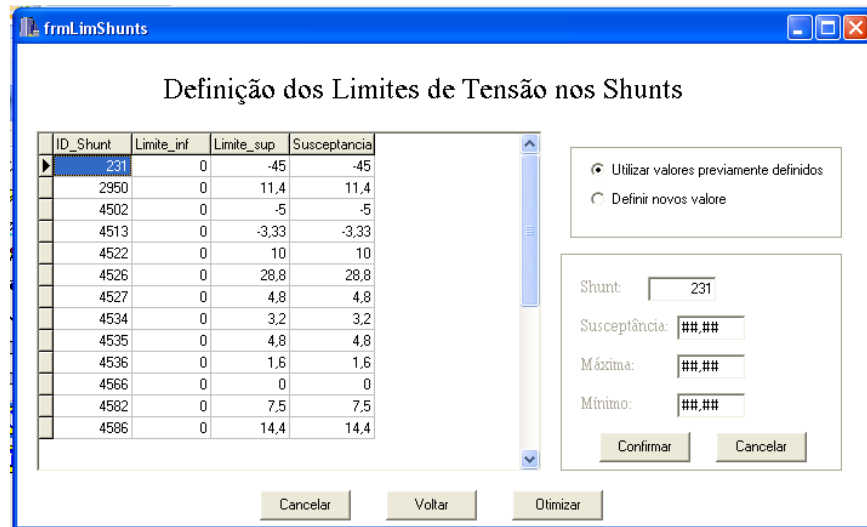


Figura 6.11 - Definição dos Limites de Reativos dos Shunts

Após esta ultima etapa, é iniciado o processo iterativo de otimização, que para este experimento foi realizado em um tempo de duas horas, sendo processado em um microcomputador comum com as configurações definidas na tabela 6.2.

TABELA 6.2 – CONFIGURAÇÃO DO MICROCOMPUTADOR

Processador	Intel Pentium Dual Core E2140 1.6 GHz
Memória RAM	1 GB
Sistema Operacional	Windows Xp

Os valores de reativos propostos pela solução otimizada do aplicativo são apresentados na tabela 6.3, e na figura 6.12. É apresentado um gráfico comparativo entre os valores de reativos propostos antes da solução otimizada, e os valores propostos após esta.

Analisando-se o gráfico da figura 6.12, pode-se constatar significativas diferenças entre os valores de antes a após a solução otimizada, sendo que nesta ultima valores menores de

reativos são injetados na rede, diminuindo assim as perdas relacionadas a esse procedimento, e ao mesmo tempo sendo mantidos os valores de tensão dentro dos limites pré estabelecidos, como pode ser observado na figura 6.13, e no ANEXO B, na tabela B.2.

TABELA 6.3 – Valores de Reativo Propostos

	ID Shunt	Reativo
1	231	-31.41
2	2950	6.035
3	4502	-3.882
4	4513	-2.338
5	4522	7.49
6	4526	24.85
7	4527	2.786
8	4534	2.497
9	4535	4.066
10	4536	1.431
11	4566	0
12	4582	7.471
13	4586	14.4

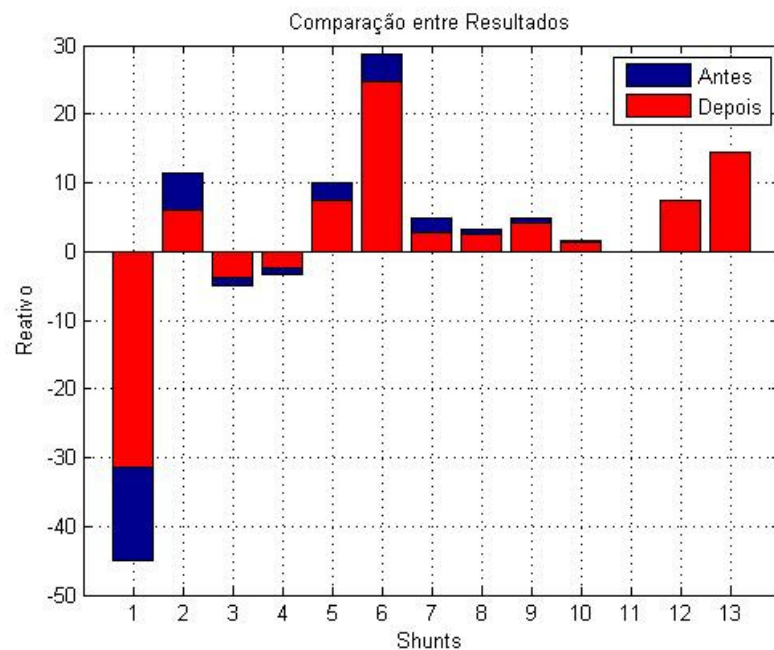


Figura 6.12 – Comparação entre os Valores de Reativos Antes e Depois da Otimização

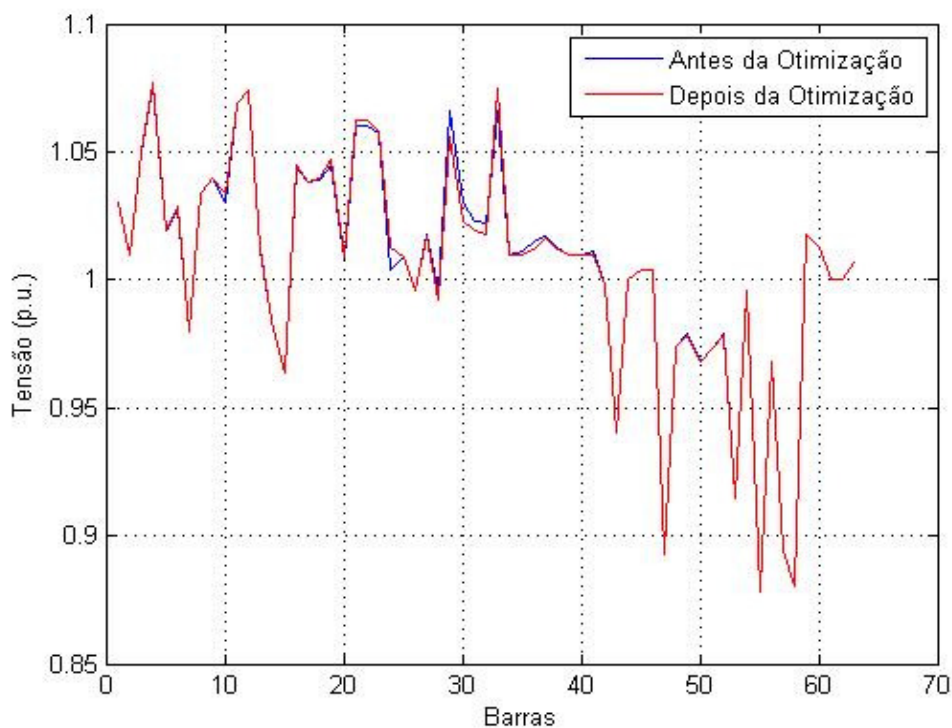


Figura 6.13 – Valores de Tensão nas Barras

6. 6 – Conclusões

Com a realização e análise deste estudo de caso, foi possível comprovar, utilizando um sistema real, que o aplicativo desenvolvido consegue atender seus requisitos, e assim atender a sua proposta e objetivo.

Observou-se que o algoritmo foi capaz de propor soluções otimizadas para a injeção de reativos no sistema elétrico, e ao mesmo tempo, manter os valores de tensão dentro dos limites especificados.

CAPITULO 7 - CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

O principal objetivo desta dissertação foi o de desenvolver um aplicativo baseado em inteligência computacional que auxiliasse no planejamento de sistemas elétricos, através de uma metodologia para controle de tensão e minimização das perdas, através da otimização da injeção de reativos, mantendo a tensão nos barramentos dentro de limites pré estabelecidos.

O uso de algoritmos genéticos para a solução do problema mostrou-se adequado, e o tempo de processamento aceitável, visto que este tipo aplicação não demanda de respostas em tempo real.

A utilização de um programa de fluxo de carga de uso consolidado no setor elétrico, gerou uma maior eficiência e conseqüentemente uma maior usabilidade do software desenvolvido, podendo este ser facilmente utilizado, sem qualquer adaptação ou conversão dos arquivos já modelados no formato do ANAREDE.

Para demonstrar as funcionalidades do aplicativo, foi apresentado um estudo de caso real envolvendo a área '45' do SIN, que corresponde a regional do Mato Grosso da ELETRONORTE, utilizando os dados obtidos através do ONS. Neste estudo o software apresentou resultados satisfatório, propondo uma solução com uma menor injeção de reativos, comparada a configuração proposta pelo arquivo do ONS, mantendo os valores de tensão nas barras, próximas aos valores antes da otimização.

Dentre as melhorias que ainda podem ser realizadas estão:

- Utilizar técnicas para aumentar a performance do algoritmo, tornando a solução mais rápida;
- Realizar melhorias na interface gráfica, principalmente na forma como os resultados gráficos são exibidos;
- Comparar a metodologia utilizadas para a otimização com outras técnicas como Enxame de Particulas ou outras variações de AG.

BIBLIOGRAFIA

Bezerra, U. H., Junior, G. C., Nunes, M. V. A., Filho, E. M. A., Carvalho, A. J. B., Junior, J. M. C., Rocha, M. G., Neto, J. R. S. S., 2008, “*Controle Ótimo por Meio da Minimização de Transitórios em Chaveamento de Compensadores de Reativos Estáticos, Série e Shunt, com Abordagem via Algoritmo Genético*”, Relatório de P&D, UFPA, Belém - PA, Brasil.

Beê, R. T., “*Alocação de bancos de capacitores em sistemas de distribuição de energia elétrica utilizando algoritmos genéticos*”. Dissertação de Mestrado, UFPR.

Belpiede, D., 2006, “*Aplicação de Algoritmos Evolutivos para a otimização do fluxo de potência em Sistemas de Subtransmissão de Energia*”. Dissertação de Mestrado, USP.

Carvalho, A. J. B., Rocha, M. G., Batista, D. A. R., Neto, J. R. S. S., Filho, E. M. A., Junior, J. M. C., Nunes, M. V. A., Bezerra, U. H., Junior, G. C., 2007, “*Ferramenta Computacional em C++ para Auxiliar a Análise das Sobretensões Provocadas por Manobras no Sistema de Transmissão de Mato Grosso Utilizando Simulações Desenvolvidos no ATP*”, XIX SNPTEE, Rio de Janeiro –RJ, Brasil.

El-Hawary, M. E., 2000, “*Electrical Energy Systems*”, Boca Raton, FL: CRC Press.

Estevam, C. R. N., 2008, “*Planejamento de reativos em sistemas de energia elétrica através de um algoritmo de branch-and-bound não linear*”. Teses de Doutorado, UNESP

Fowler, M., 2005, “UML Essencial”, Bookman

Goldberg, D., E., 1989, “*Genetic Algorithm in Search Optimization and Machine Learning*”, Addison Wesley

Haykin, S., 1994, “*Neural Networks A Comprehensive Foundation*”, Prentice Hall

Hinrichs, R. A., Kleinbach, M., 2003, “*Energia e Meio Ambiente*”, Thompson.

Kennedy, J. Eberhart, R. C., 1995, “*Particle Swarm Optimization*”, Proceedings IEEE, IV Conference on Neural Networks, 1995, pp. 1942-1948

Koumoussis, V. K., Katsaras, C. P., 2006, “A Saw – Tooth Genetic Algorithm Combining the Effects of Variable Population Size and Reinitialization to Enhance Performance.”, IEEE Transactions on Evolutionary Computation, Vol 10, No 1,

Linden, R., 2006, “Algoritmos Genéticos: Uma Importante Ferramenta da Inteligência Computacional”, Rio de Janeiro, Brasport.

Malachi, Y., Singer, S., 2006, “ A genetic algorithm for corrective control of voltage and reactive power”, Power System, IEEE Transactions on.

Mendel, C. A., 2006, “Utilização de Algoritmos Genéticos em Recomposição de Sistemas de Potência”, Dissertação de Mestrado, UNISINOS.

Miasaki, C. T., Romero, R., 2007, “Um algoritmo genético especializado aplicado ao planejamento da expansão do sistema de transmissão com alocação de dispositivos de compensação série”, Revista Controle & Automação, Vol. 18, no 2.

Pinheiro, F. A., 1998, “Aplicação de um Algoritmo Genético no Estudo das Perdas e do Controle de Tensão em Sistemas Elétricos de Potência”. Dissertação de Mestrado, UFMG.

Stevenson, W. D. Jr., 1986, *Elementos de Análise de Sistmas de Potência*, McGraw-Hill.

Shi, X. H., Wan, L. M., Lee, H. P., Yang, X. W., Wang, L. M., Liang, Y. C., 2003, “An Improved Genetic Algorithm with Variable Population-Size and a PSO_GA Based Hybrid Evolutionary Algorithm”, Proceedings of the School Internacional Conference on Machine Learning and Cybernetics, Xi'an.

Ting, T. O., Wong, K. P., Chung, C.Y., 2008, “Hybrid constrained genetic /particle swarm optimization load flowalgorithm”, IET generation, transmission & distribution.

Zhu, J. Z., 2002, “Optimal Reconfiguration of Electrical Distribution Network Using The Refined Genetic Algorithm”, Electric Power System Research.

Wall, M., 1996, “GAlib: A C++ Library of Genetic Algorithm Components”, Library Documentations, Massachusetts Institute of Tecnology.

Wong, K. P., Li, A., 1995, "*Solving The Load-Flow Problem Using Genetic Algorithm*", IEEE Xplore.

ANEXO A - Arquivo de Configuração do Algoritmo Genético

O arquivo de configuração do algoritmo genético, ver figura A.1, proporciona ao usuário do aplicativo alterar os parâmetros do AG, que são descritos na tabela A.1.

Tabela A.1 – Descrição dos Campos

Campo	Descrição
1	Número de Gerações (N_{ger}) $N_{ger} = 1, 2, 3, \dots$
2	Tamanho da População (N_{pop}) $N_{pop} = 1, 2, 3, \dots$
3	Probabilidade de Mutação $0 \leq \mu \leq 1$
4	Probabilidade de Crossover $0 \leq \gamma \leq 1$

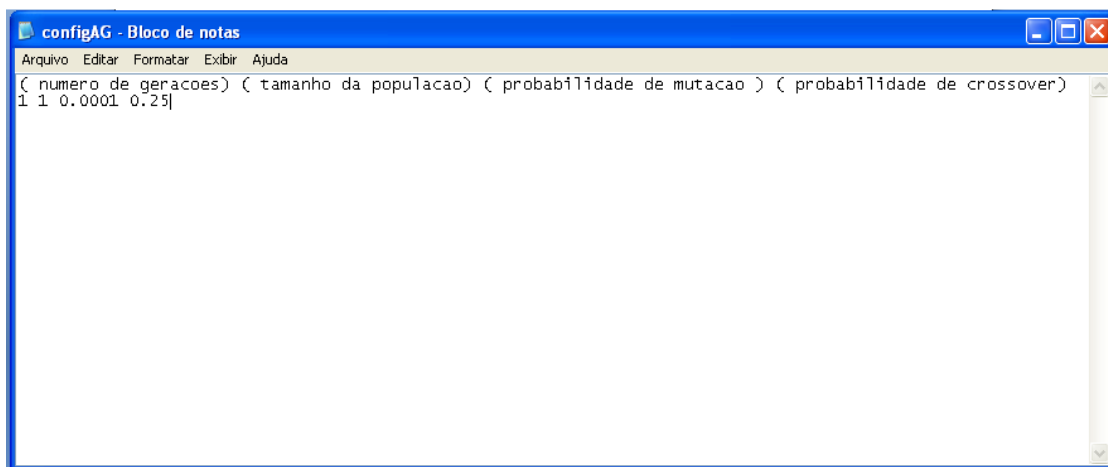


Figura A.1 – Arquivo de Configuração

ANEXO B - Dados do Sistema e dos Experimentos

Tabela B.1 – Tensão e Limites nas Barras

Barra	Esp	Min	Max
18	1.03	0.95	1.05
21	1.01	0.95	1.05
100	1.05	0.95	1.05
210	1.075	0.95	1.075
216	1.019	0.95	1.05
217	1.026	0.95	1.05
218	1.01	0.95	1.05
220	1.034	0.95	1.05
225	1.04	0.95	1.05
231	1.026	0.95	1.05
320	1.069	0.95	1.069
370	1.074	0.95	1.074
749	1.011	0.95	1.05
760	1.014	0.95	1.05
790	1.007	0.95	1.05
2950	1.044	0.95	1.05
2978	1.039	0.95	1.05
4501	1.032	0.95	1.05
4502	1.027	0.95	1.05
4503	0.992	0.95	1.05
4506	1.043	0.95	1.05
4509	1.043	0.95	1.05
4512	1.05	0.95	1.05
4513	1.013	0.95	1.05
4515	1.008	0.95	1.05
4516	0.996	0.95	1.05
4522	1.009	0.95	1.05
4524	0.975	0.95	1.05
4526	1.058	0.95	1.058
4527	1.015	0.95	1.05
4528	1.015	0.95	1.05
4529	1.014	0.95	1.05
4530	1.011	0.95	1.05
4532	1.01	0.95	1.05
4533	1.023	0.95	1.05

4534	1.066	0.95	1.066
4535	1.066	0.95	1.066
4536	1.066	0.95	1.066
4537	1.023	0.95	1.05
4538	1.023	0.95	1.05
4539	1.023	0.95	1.05
4542	1.026	0.95	1.05
4552	1.01	0.95	1.05
4554	1	0.95	1.05
4556	1.005	0.95	1.05
4559	1.005	0.95	1.05
4562	1.02	0.95	1.05
4564	0.98	0.95	1.05
4566	1.074	0.95	1.074
4567	0.975	0.95	1.05
4568	0.981	0.95	1.05
4569	0.988	0.95	1.05
4572	1.015	0.95	1.05
4576	1.001	0.95	1.05
4582	1.025	0.95	1.05
4583	1.045	0.95	1.05
4586	1.047	0.95	1.05
4589	1.029	0.95	1.05
4592	1.035	0.95	1.05
4593	1.023	0.95	1.05
4860	1.001	0.95	1.05
4861	1.001	0.95	1.05
4862	1.007	0.95	1.05

Tabela B.2 – Valores de Tensão Antes e Após a Otimização

Barra	Especificada	Sem Otimização	Após a Otimização
18	1.03	1.03	1.03
21	1.01	1.01	1.01
100	1.05	1.05	1.05
210	1.075	1.076	1.077
216	1.019	1.019	1.019
217	1.026	1.028	1.029
218	1.01	0.98	0.98
220	1.034	1.034	1.034
225	1.04	1.04	1.04
231	1.026	1.03	1.034
320	1.069	1.069	1.069
370	1.074	1.074	1.074
749	1.011	1.012	1.013
760	1.014	0.984	0.984
790	1.007	0.964	0.964
2950	1.044	1.044	1.045
2978	1.039	1.038	1.038
4501	1.032	1.039	1.04
4502	1.027	1.045	1.047
4503	0.992	1.008	1.009
4506	1.043	1.06	1.062
4509	1.043	1.06	1.062
4512	1.05	1.057	1.058
4513	1.013	1.004	1.013
4515	1.008	1.009	1.009
4516	0.996	0.996	0.996
4522	1.009	1.018	1.017
4524	0.975	0.996	0.992
4526	1.058	1.066	1.056
4527	1.015	1.03	1.023
4528	1.015	1.023	1.019
4529	1.014	1.022	1.018
4530	1.011	1.066	1.075
4532	1.01	1.01	1.01
4533	1.023	1.011	1.01
4534	1.066	1.015	1.012
4535	1.066	1.017	1.016
4536	1.066	1.013	1.012
4537	1.023	1.01	1.01

4538	1.023	1.01	1.01
4539	1.023	1.011	1.01
4542	1.026	0.998	0.998
4552	1.01	0.94	0.94
4554	1	1	1
4556	1.005	1.004	1.004
4559	1.005	1.004	1.004
4562	1.02	0.893	0.893
4564	0.98	0.974	0.974
4566	1.074	0.979	0.978
4567	0.975	0.968	0.968
4568	0.981	0.973	0.973
4569	0.988	0.979	0.978
4572	1.015	0.915	0.915
4576	1.001	0.996	0.996
4582	1.025	0.878	0.878
4583	1.045	0.968	0.968
4586	1.047	0.894	0.894
4589	1.029	0.88	0.88
4592	1.035	1.018	1.018
4593	1.023	1.013	1.013
4860	1.001	1	1
4861	1.001	1	1
4862	1.007	1.007	1.007

ANEXO C - Códigos de Execução do Arquivo de Entrada de Dados do ANAREDE Utilizados no Aplicativo

Neste anexo serão detalhados os campos e as funções dos códigos de execução do arquivo de entrada de dados do ANAREDE (referenciar), que foram utilizadas no aplicativo.

C.1 - DARE

Este código de execução é responsável pela leitura de dados de intercâmbio de potência ativa entre áreas, ver figura C.1.

O formato dos dados, assim como a descrição dos campos, é apresentado na tabela C.1.

```
X      .95  1.1
Z      .95  1.046
99
DARE
(Ar)  (xchg)  ( Identificacao da area ) (Xmin) (Xmax)
1      0.      *          FURNAS          *
2      0.      *          CEMIG G/T          *
3      0.      *          CEMIG D            *
4      0.      *          **** CTEEP - GRANDE SAO PAULO ****
5      0.      *          **** C T E E P          ****
6      0.      *          **** C T E E P - SECUNDARIA ****
7      0.      *          CPFL
8      0.      *          **** C E S P          ****
9      0.      *          LIGHT
10     0.      *          CERJ
11     0.      *          ESCELSA
12     0.      *          CELC
```

Figura C.1 – Código de Execução DARE

Tabela C.1 – Formato dos Dados de Área

Campo	Colunas	Descrição
Numero	01-04	Número da área
Intercâmbio Líquido	09-14	Valor líquido de intercâmbio da área, em MW (valor positivo para exportação e negativo para importação)
Nome	20-55	Identificação Alfanumérica da Área
Intercâmbio Mínimo	57-62	Valor mínimo de intercâmbio da área, em MW (valor positivo para exportação e negativo para importação)
Intercâmbio Máximo	64-69	Valor máximo de intercâmbio da área, em MW (valor positivo para exportação e negativo para importação)

C.2 – DBAR

Este código de execução é responsável pela leitura de dados das barras CA, ver figura C.2 .

O formato dos dados, assim como a descrição dos campos, é apresentado na tabela C.2.

```

FGER      30.  LFCV      1.  DCIT      10.  VSIT      10.  LPIT      50.  LFLP      10.
ACIT      10.  LCRT      24.  LPRT      60.  CSTP      500.
PDIT      30.  DMAX      5.  FDIV      2.  ICMN      .05.  VART      5.  TSTP      32.
ICIT      1.  APAS      90.  CPAR      70.  VAVT      2.  VAVF      5.  VMVF      15.
ICMV      2.  VPMF      5.  VPMF      10.  VSVF      20.  VINP      1.  VSUP      1.
VPVT      9999
DBAR
(No) 0 TB( nome )G( V)( A)( Pg)( Qg)( Qn)( Qm)( Bc)( P1)( Q1)( Sh)(A(vf)
10 1WANGRA-1--1GR8 980-41. 520. 6.31-211. 414. 28.5 16.9 441000
11 1WANGRA-2--1GR8 980-39.1350.55.04-600. 600. 67.9 47.1 441000
12 1WLCBARRET-6GR81040-4.91100.296.7-540. 420. 11000
14 1WFUNIL-1--1GR81000-53. 29.-4.87 -50. 50. 11000
15 1WFUNIL-2--2GR81030-67. 59.69.11-100. 100. 11030
16 1WFURNAS---8GR81050-14.1200.359.2-720. 480. 11000
17 1WFURNAS---8GR81050-14.1200.359.2-720. 480. 11000
    
```

Figura C.2 – Código de Execução DBAR

Tabela C.2 – Formato dos Dados de Barra

Campo	Colunas	Descrição
Número	01-04	Número de identificação da barra CA
Operação	06-06	A ou 0 – Adição de dados de barra E ou 1 – Eliminação de dados de barra M ou 2 – Modificação de dados de barra
Tipo	08-08	0 – Barra de carga (PQ injeções de potência ativa e reativa fixas). 1 – Barra de tensão regulada (PV injeções de potência ativa e magnitude de tensão fixas). 2 – Barra de referência. 3 – Barra de carga com limite de tensão (PQ injeções de potência ativa e reativa fixas enquanto que a magnitude de tensão permanece entre os valores limites).
Grupo de Base de Tensão	09-09	Identificação alfanumérica do grupo base de tensão
Nome	10-21	Identificação alfanumérica da barra
Grupo Limite de Tensão	22-22	Identificação alfanumérica dos grupos de limite de tensão
Tensão	23-26	Valor inicial da magnitude de tensão, em p.u. Para barra de tensão controlada, remotamente ou não, por geração de potência reativa ou por variação de TAP de transformador, este campo deve ser preenchido com o valor da magnitude de tensão a ser mantido constante
Ângulo	27-30	Ângulo de fase inicial da tensão da barra, em graus.
Geração Ativa	31-35	Valor de geração de potência ativa na barra, em MW.
Geração Reativa	36-40	Valor de geração de potência reativa na barra, em MW.
Geração Reativa Mínima	41-45	Valor do limite mínimo de geração de potência reativa na barra, em Mvar
Geração Reativa Máxima	46-50	Valor do limite máximo de geração de potência reativa na barra, em Mvar
Barra Controlada	51-55	Para barra de tensão regulada e de referência, com limites de potência reativa especificados, este campo destina-se ao número da barra cuja a magnitude está sendo controlada
Carga Ativa	56-60	Valor da carga ativa da barra, em MW
Carga Reativa	61-65	Valor da carga reativa da barra, em Mvar
Capacitor Reator	66-70	Valor total de potência reativa injetada na barra, Mvar, por bancos de capacitores/reatores
Área	71-72	Número da área à qual pertence a barra
Tensão para Definição de Carga	73-76	Valor em p.u. da tensão para qual foi medido o valor das parcelas ativa e reativa da carga definida nos campos Carga Ativa e Carga Reativa, respectivamente.

ANEXO D - Experimentos Realizados

D.1 – Experimento 1

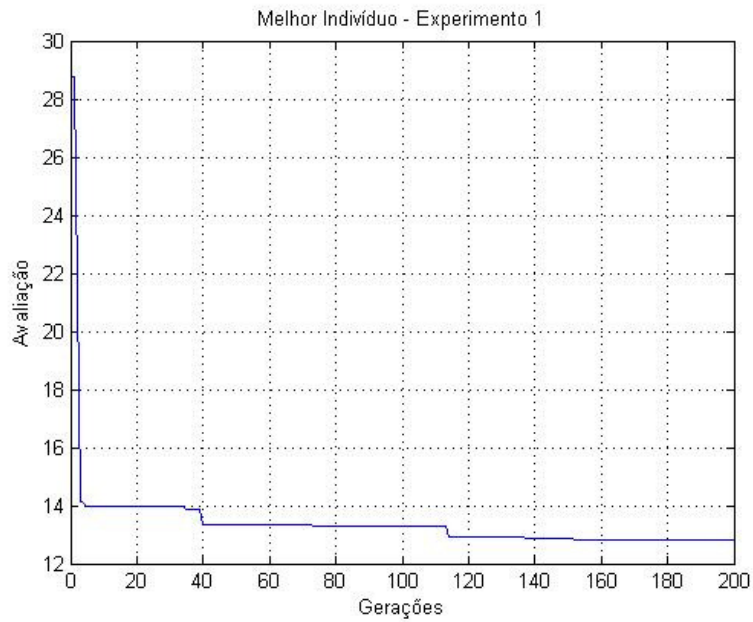


Figura D.1 – Melhor Indivíduo no Experimento 1

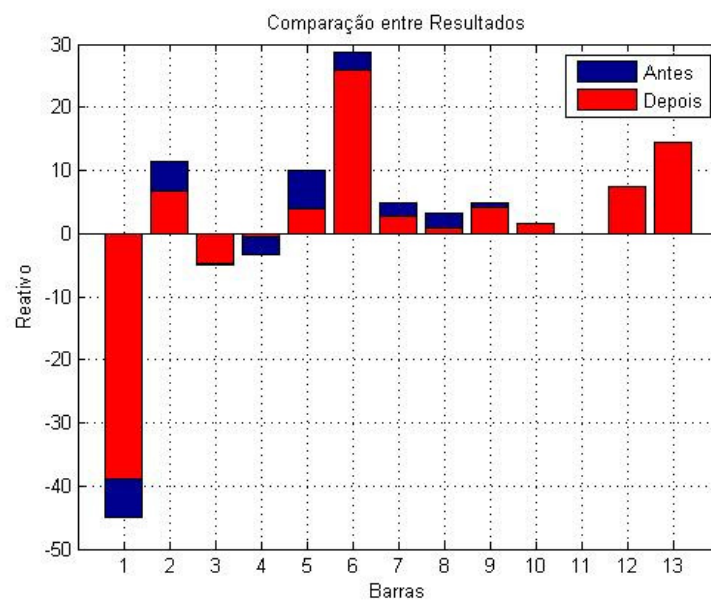


Figura D.2 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 1

D.2 – Experimento 2

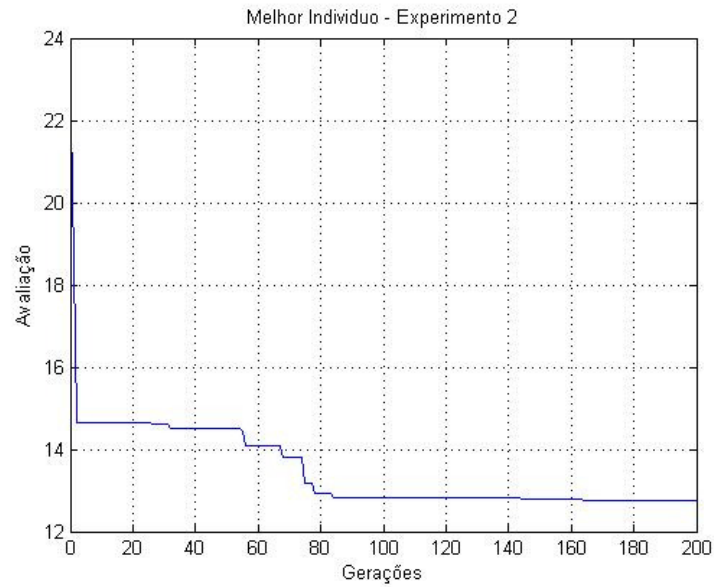


Figura D.3 – Melhor Indivíduo no Experimento 2

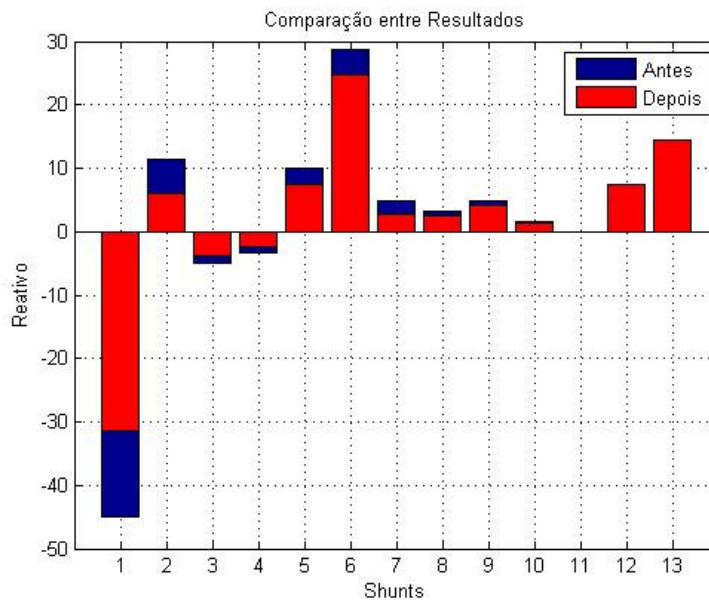


Figura D.4 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 2

D.3 – Experimento 3

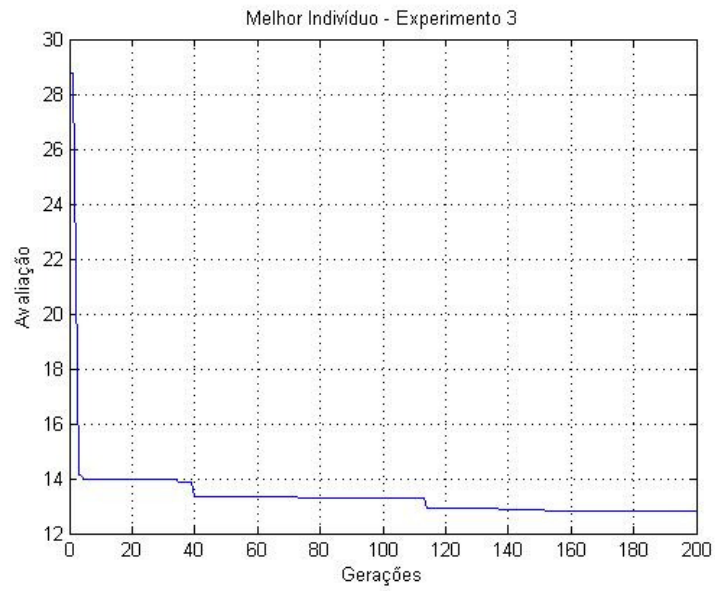


Figura D.5 – Melhor Indivíduo no Experimento 3

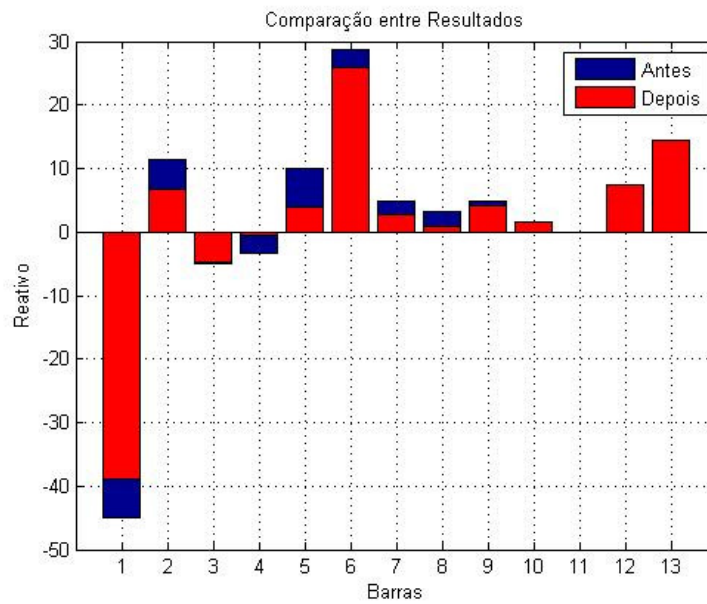


Figura D.6 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 3

D.4 – Experimento 4

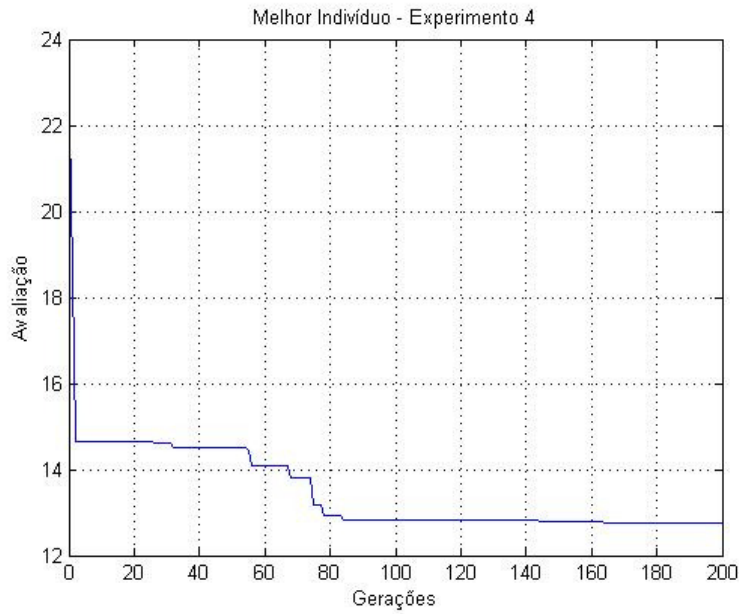


Figura D.7 – Melhor Indivíduo no Experimento 4

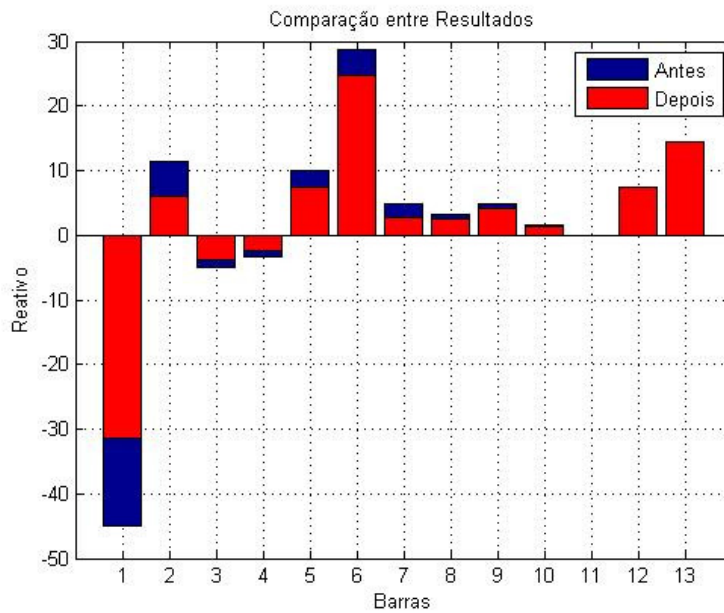


Figura D.8 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 4

D.5 – Experimento 5

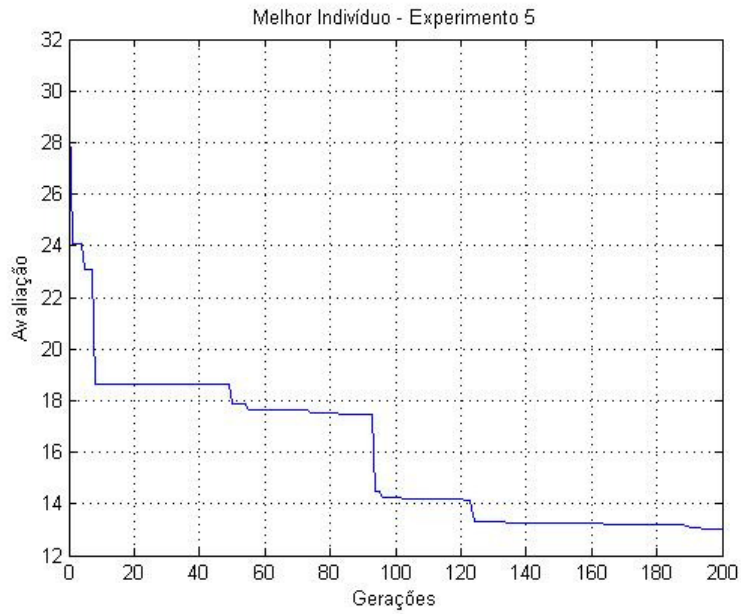


Figura D.9 – Melhor Indivíduo no Experimento 5

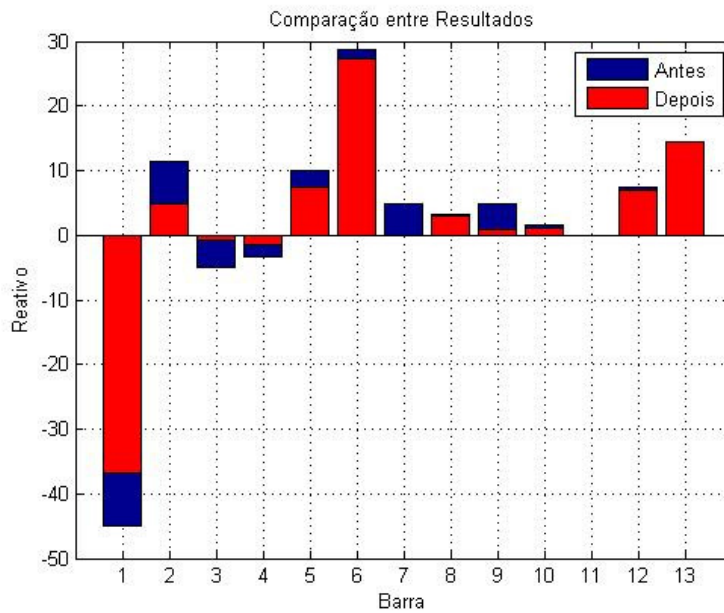


Figura D.10 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 5

D.6 – Experimento 6

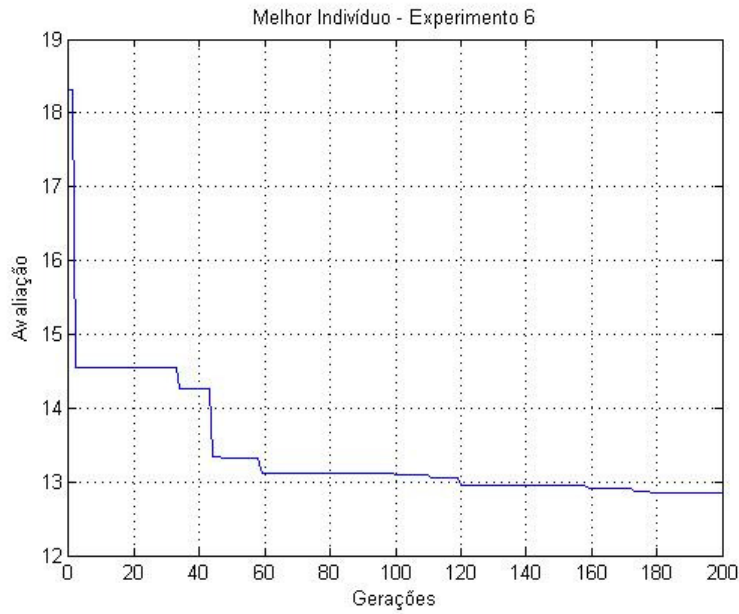


Figura D.11 – Melhor Indivíduo no Experimento 6

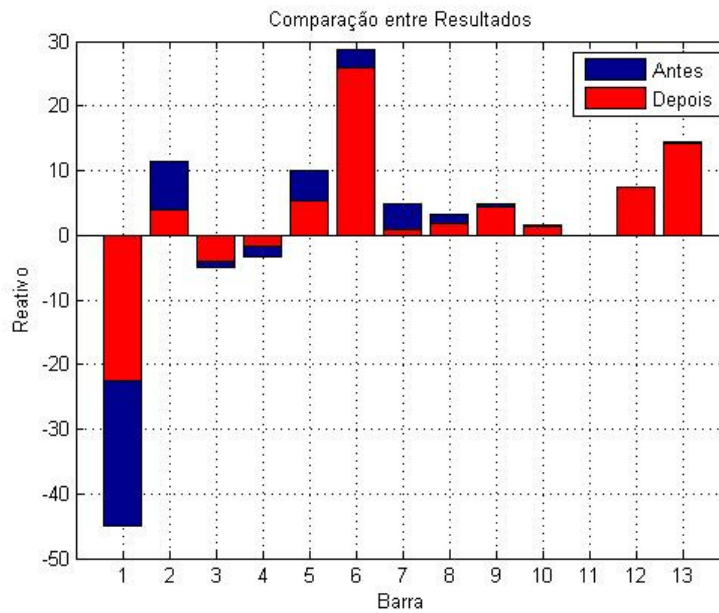


Figura D.12 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 6

D.7 – Experimento 7

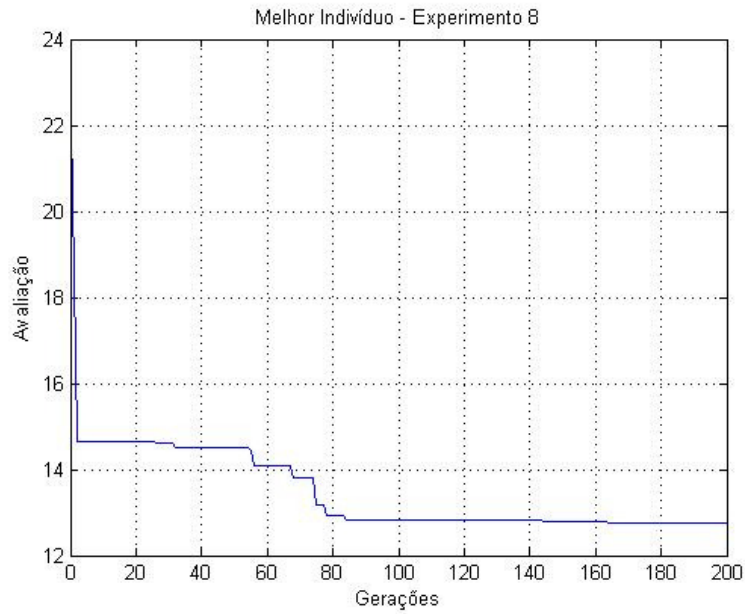


Figura D.13 – Melhor Indivíduo no Experimento 7

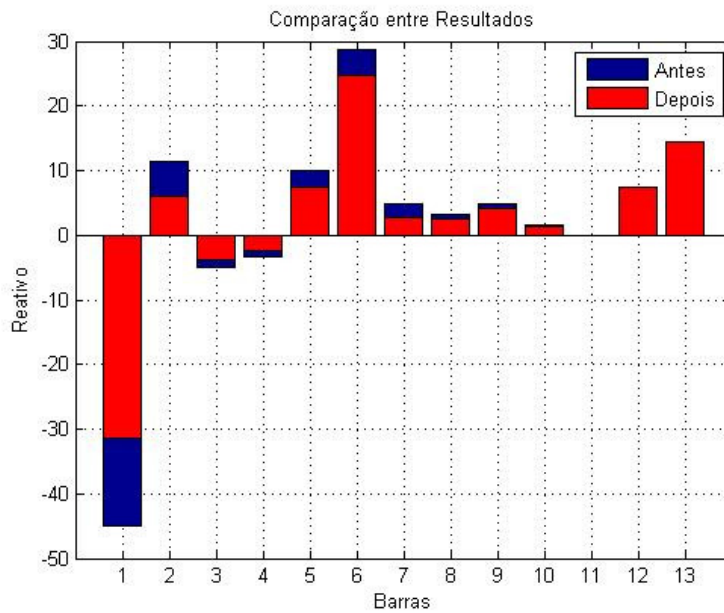


Figura D.14 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 7

D.8 – Experimento 8

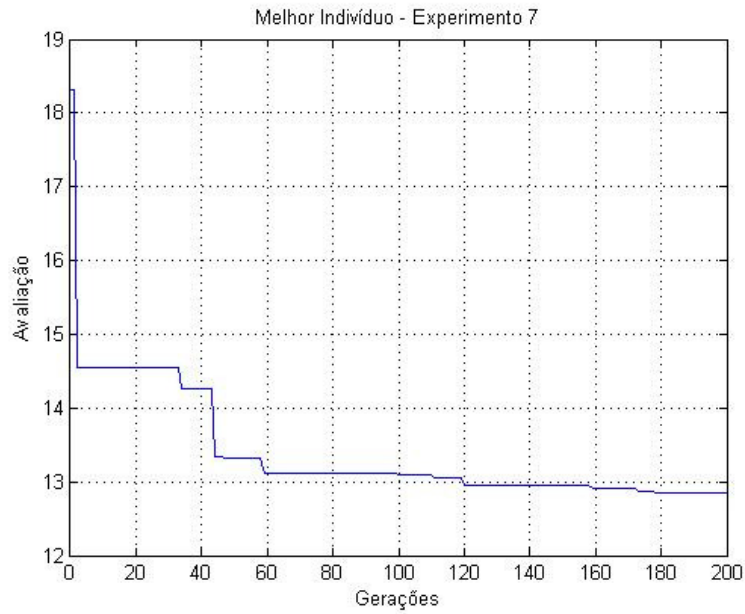


Figura D.15 – Melhor Indivíduo no Experimento 8

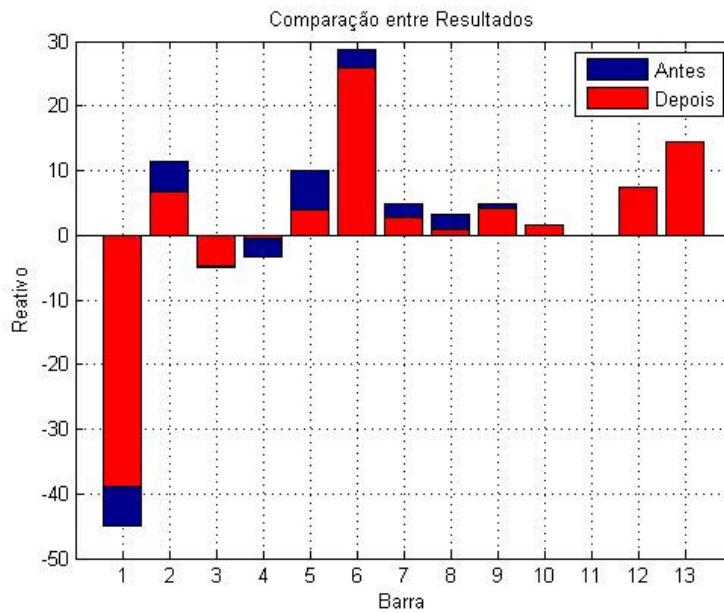


Figura D.16 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 8

D.9 – Experimento 9

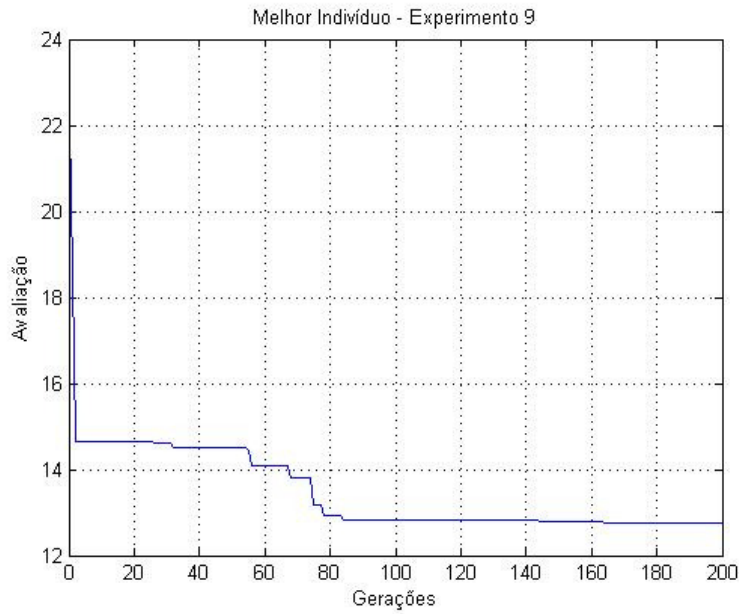


Figura D.17 – Melhor Indivíduo no Experimento 9

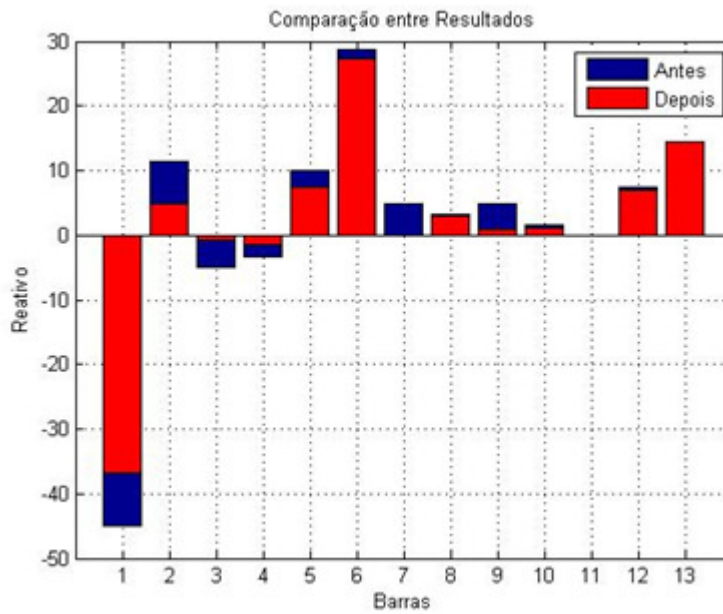


Figura D.18 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 9

D.10 – Experimento 10

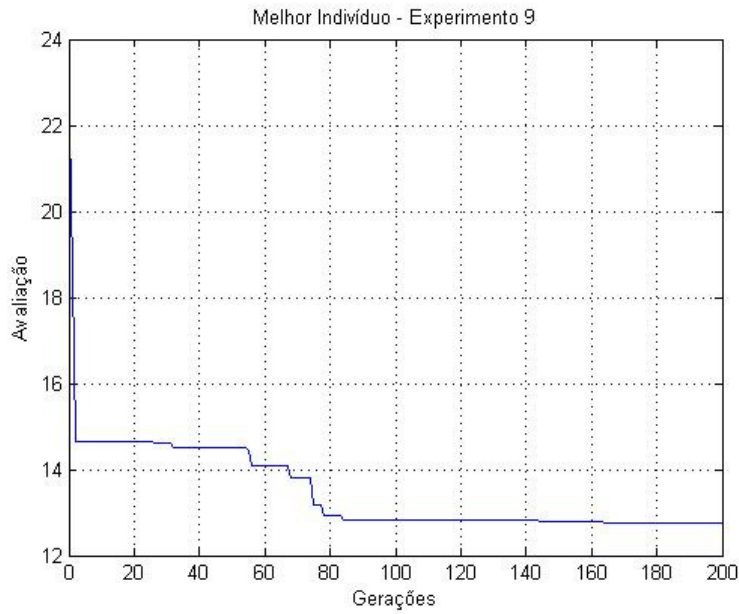


Figura D.19 – Melhor Indivíduo no Experimento 10

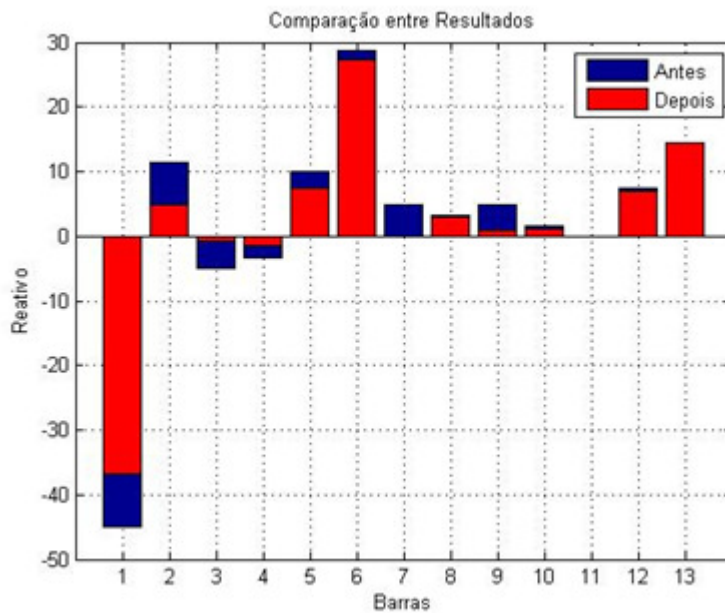


Figura D.20 – Comparação entre os Valores de Reativos Antes e Depois da Otimização para o Experimento 10