

UNIVERSIDADE DO VALE DO RIO DOS SINOS
CIÊNCIAS EXATAS E TECNOLÓGICAS
PROGRAMA INTERDISCIPLINAR DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
APLICADA

**Utilização de Algoritmos Genéticos Em
Recomposição de Sistemas de Potência**

por

CARLOS ALBERTO MENDEL

São Leopoldo, abril de 2006

Carlos Alberto Mendel

***Utilização de Algoritmos Genéticos Em
Recomposição de Sistemas de Potência***

Dissertação apresentada à Universidade do Vale
do Rio dos Sinos (UNISINOS) como requisito
parcial para a obtenção do título de Mestre em
Computação Aplicada

Orientador:
Adelmo Luis Cechin

São Leopoldo

2006

AGRADECIMENTOS

Agradeço, de forma geral, a todos que se envolveram na elaboração deste trabalho.

Agradeço, em especial, ao orientador, prof. Adelmo Cechin e ao prof. J. V. Canto dos Santos, pelo incentivo e pelo interesse despertado pelo assunto. Também pela oportunidade de ter trabalhado com ambos, além de outros, em projeto relacionado ao assunto, junto à CEEE, empresa a qual também tenho a agradecer.

Agradeço à Universidade, que por mais de uma década, me influenciou e que continuará a influenciar, por muito tempo.

Por fim, mas não com menos importância, agradeço aos familiares, colegas e amigos, que foram relegados a segundo plano por diversas vezes e que, mesmo assim, sempre me apoiaram.

RESUMO

Este trabalho apresenta um método de Inteligência Artificial para ser utilizado em operações de Restauração de Sistemas de Potência, com o desenvolvimento de um sistema híbrido. O sistema utiliza um Algoritmo Genético (AG), que codifica seqüências de operações de chaveamento. Durante a evolução do AG, as operações sugeridas são avaliadas por um programa de Fluxo de Potência, que fornece dados para a função objetivo deste AG. Para este AG, uma nova representação de genoma foi criada, capaz de codificar estágios compostos de blocos de operações sequenciais. Juntamente a esta nova representação, foram desenvolvidos operadores de cruzamento e de mutação.

Palavras-chave: Restauração de Sistemas de Potência, Inteligência Artificial, Algoritmos Genéticos.

ABSTRACT

This work reports the development of an Artificial Intelligence system applied to Power Systems Restoration, with an hybrid approach. The system employs a Genetic Algorithm (GA) to generate sequences of operations which are analysed by a Power Flow program to verify and access their fitness. For this GA, a new genoma representation was developed, as well as two genetic operators, for crossover and mutation.

keywords: Power Systems Restoration, Hybrid Systems, Artificial Neural Networks, Genetic Algorithms, Power Flow.

SUMÁRIO

Lista de Figuras

Lista de Abreviaturas

1	Introdução	18
2	Conceitos Básicos	21
2.1	Sistemas de Potência	21
2.1.1	Componentes	21
2.1.1.1	Estações Geradoras	21
2.1.1.2	Linhas de Transmissão	22
2.1.1.3	Sistemas de Distribuição	23
2.1.2	Conceitos Básicos sobre Circuitos Elétricos	23
2.1.3	Representação Unifilar	26
2.1.4	Normalização - Grandezas por Unidade [p.u.]	27
2.1.5	Linhas de Transmissão	28
2.1.5.1	Quadripolos	28
2.1.5.2	Modelo de Linhas de Transmissão Longa	30
2.2	Fluxo de Potência Resumido	32
2.2.1	Matriz de Admitâncias	33
2.2.2	Classificação das Barras	36
2.2.2.1	Barra de Referência - <i>Slack Bus</i> ou <i>Swing Bus</i>	36
2.2.2.2	Barra de Carga ou Barra P-Q	36
2.2.2.3	Barra de Tensão Controlada ou Barra P-V	36

2.2.3	O Fluxo de Potências Generalizado	37
2.2.4	Métodos de Solução	37
2.2.4.1	Método de Newton-Raphson	37
2.2.4.2	Método Desacoplado Rápido	39
2.3	Algoritmos Genéticos	40
2.3.1	Componentes dos Algoritmos Genéticos	40
2.3.1.1	Cromossomos	41
2.3.1.2	Função Objetivo (<i>Fitness</i>)	41
2.3.1.3	Operadores de Algoritmo Genético (AG)s	41
2.3.1.4	<i>Schemata</i>	41
2.3.2	Descrição geral do algoritmo	42
3	Estado da Arte	44
3.1	Restauração de Sistemas	44
3.1.1	Métodos Analíticos	47
3.1.2	Sistemas Especialistas em Restauração de Sistemas de Potência (RSP)	48
3.1.3	Sistema Híbridos	53
3.1.4	Redes Neurais Artificiais (RNA)	54
3.1.5	Outras técnicas de Inteligência Artificial (IA)	55
3.1.6	Algoritmos Genéticos	55
3.1.7	Conclusões da Revisão	57
4	Metodologia	58
4.1	Descrição Geral	58
4.2	Algoritmos Genéticos	59
4.2.1	Representação do Genoma	59
4.2.2	Inicialização dos Genomas	62
4.2.3	Detalhamento da Representação do Genoma	65
4.2.4	Inter-operação com o Fluxo de Potência	67

4.2.5	Operador de Cruzamento	67
4.2.6	Operador de Mutação	68
4.2.7	Função Objetivo	69
4.2.7.1	Funções Implementadas	69
4.3	Fluxo de Potência	72
4.3.1	Compensação por não convergências	73
4.4	Modelos de Sistemas	74
4.5	Conclusão da Metodologia	75
5	Implementação	77
5.1	Descrição Geral	77
5.2	Modelo Descritivo	78
5.2.1	Representação em Linguagem UML	80
5.3	Fluxo de Potência	80
5.4	Algoritmo Genético	83
5.4.1	Implementação do Genoma	85
5.4.2	Inicialização de cada indivíduo	85
5.5	Exportação dos Dados	86
6	Experimentos e Resultados	87
6.0.1	Experimentos com Funções Objetivo	88
6.0.2	Variação de Parâmetros do AG	88
6.1	Resultados do Algoritmo Genético	88
6.1.1	Função Objetivo 0	88
6.1.1.1	Sistema IEEE 14 Barras	89
6.1.1.2	Sistema IEEE 30 barras	91
6.1.2	Função Objetivo 1	92
6.1.2.1	Sistema IEEE 14 Barras	93
6.1.2.2	Sistema IEEE 30 Barras	94

6.1.3	Função Objetivo 2	95
6.1.3.1	Sistema IEEE 14 Barras	96
6.1.3.2	Sistema IEEE 30 Barras	98
6.1.4	Função Objetivo 3	99
6.1.4.1	Sistema IEEE 14 Barras	99
6.1.4.2	Sistema IEEE 30 Barras	100
6.1.5	Função Objetivo 4	100
6.1.5.1	Sistema IEEE 14 Barras	101
6.1.5.2	Sistema IEEE 30 Barras	102
6.1.6	Sistema CEEE	103
6.1.6.1	Função Objetivo 1	103
6.1.6.2	Função Objetivo 2	105
6.1.6.3	Função objetivo 3	105
6.1.6.4	Função Objetivo 4	107
6.2	Avaliação da Performance	107
6.3	Conclusões sobre os resultados dos experimentos	108
7	Conclusão	109
	Anexo A – Formato PSAP	111
	Anexo B – Formato CDF	114
	Anexo C – Formato ANAREDE	120
	Anexo D – LICENÇA GALIB	125
	Anexo E – Gráficos dos Resultados - Completos	127
E.1	Sistema IEEE 14	127
E.1.1	Função 0	127
E.1.2	Função 1	128

E.1.3	Função 2	130
E.1.4	Função 3	133
E.1.5	Função 4	136
E.2	Sistema IEEE 30	138
E.2.1	Função 0	138
E.2.2	Função 1	138
E.2.3	Função 2	141
E.2.4	Função 3	144
E.2.5	Função 4	147

Referências

LISTA DE FIGURAS

1	Sistema de Potência	22
2	Representação Senoidal da Potência	26
3	Triângulo das Potência	26
4	Exemplo de um diagrama unifilar	27
5	Circuito π	29
6	Quadripolos em série	30
7	Linha de transmissão	30
8	Modelo exemplo de Fluxo de Potência	34
9	Circuito equivalente do modelo	34
10	Algoritmo Genético	42
11	Componentes do programa desenvolvido para RSP.	58
12	Cromossomo dividido em grupos pré-selecionados	60
13	Cromossomo dividido em grupos	60
14	Cromossomo sem divisão em estágios	61
15	Representação com múltiplos estágios e operações	61
16	Componentes da inicialização do AG	62
17	Processo de inicialização dos genomas	64
18	Sentido de ativação de linhas	66
19	Operador de Cruzamento	68
20	Operador de Mutação	69
21	IEEE-14	74
22	IEEE-30	76
23	Sistema CEEE-parcial	76

24	Componentes de Dados da Implementação	78
25	Componentes de Dados da Implementação	79
26	Diagrama UML: classes do Sistema	81
27	Diagrama UML: classes do AG	81
28	Diagrama de colaboração: AG	84
29	Algoritmo genético básico da <i>GAlib</i>	84
30	IEEE14-P:25-G:25-F:0-A:1-B:1(Completa)	89
31	IEEE14-P:50-G:50-F:0-A:1-B:1	91
32	IEEE14-P:50-G:50-F:0-A:1-B:1	91
33	IEEE30-P:25-G:25-F:0-A:1-B:1(Completa)	92
34	IEEE14-P:25-G:25-F:1-A:1-B:2(Completa)	93
35	IEEE30-P:25-G:25-F:1-A:1-B:2(Completa)	94
36	IEEE30-P:25-G:25-F:1-A:1-B:5-C:0	95
37	IEEE30-P:50-G:50-F:1-A:1-B:2-C:0	95
38	IEEE14-P:25-G:25-F:2-A:1-B:1-C:1(Completa)	96
39	IEEE14-P:50-G:100-F:2-A:1-B:1-C:0	97
40	IEEE14-P:50-G:100-F:2-A:1-B:1-C:1	97
41	IEEE30-P:25-G:25-F:2-A:1-B:2-C:1	98
42	IEEE30-P:50-G:100-F:2-A:1-B:1-C:1	98
43	IEEE14-P:25-G:25-F:3-A:0-B:2-C:1	99
44	IEEE14-P:25-G:25-F:3-A:1-B:2-C:1	99
45	IEEE30-P:25-G:25-F:3-A:0-B:2-C:1	100
46	IEEE30-P:25-G:25-F:3-A:1-B:2-C:1	100
47	IEEE14-P:25-G:25-F:4-A:1-B:5-C:1	101
48	IEEE14-P:50-G:100-F:4-A:1-B:5-C:1(Completa)	102
49	IEEE30-P:25-G:25-F:4-A:1-B:5-C:1	103
50	IEEE30-P:50-G:100-F:4-A:1-B:5-C:1	103

51	CEEE-P:25-G:25-F:1-A:1-B:1-C:0(Completa)	104
52	CEEE-P:25-G:25-F:2-A:1-B:1-C:1(Completa)	105
53	CEEE-P:25-G:25-F:3-A:1-B:1-C:1(Completa)	106
54	CEEE-P:25-G:25-F:4-A:1-B:5-C:1(Completa)	106
55	CEEE-P:50-G:50-F:4-A:1-B:2-C:1(Completa)	107
56	CEEE-P:50-G:50-F:4-A:1-B:5-C:1(Completa)	108
57	IEEE14-P:25-G:25-F:0-A:1-B:1	127
58	IEEE14-P:25-G:50-F:0-A:1-B:1	127
59	IEEE14-P:50-G:100-F:0-A:1-B:1	127
60	IEEE14-P:25-G:25-F:1-A:1-B:1-C:0	128
61	IEEE14-P:25-G:25-F:1-A:1-B:2-C:0	128
62	IEEE14-P:25-G:25-F:1-A:1-B:5-C:0	128
63	IEEE14-P:25-G:50-F:1-A:1-B:1-C:0	128
64	IEEE14-P:25-G:50-F:1-A:1-B:2-C:0	129
65	IEEE14-P:25-G:50-F:1-A:1-B:5-C:0	129
66	IEEE14-P:50-G:100-F:1-A:1-B:1-C:0	129
67	IEEE14-P:50-G:100-F:1-A:1-B:2-C:0	129
68	IEEE14-P:50-G:100-F:1-A:1-B:5-C:0	129
69	IEEE14-P:25-G:25-F:2-A:1-B:1-C:0	130
70	IEEE14-P:25-G:25-F:2-A:1-B:1-C:1	130
71	IEEE14-P:25-G:25-F:2-A:1-B:1-C:2	130
72	IEEE14-P:25-G:25-F:2-A:1-B:2-C:1	130
73	IEEE14-P:25-G:50-F:2-A:1-B:1-C:0	131
74	IEEE14-P:25-G:50-F:2-A:1-B:1-C:1	131
75	IEEE14-P:25-G:50-F:2-A:1-B:1-C:2	131
76	IEEE14-P:25-G:50-F:2-A:1-B:2-C:1	131
77	IEEE14-P:50-G:100-F:2-A:1-B:1-C:0	131

78	IEEE14-P:50-G:100-F:2-A:1-B:1-C:1	132
79	IEEE14-P:50-G:100-F:2-A:1-B:1-C:2	132
80	IEEE14-P:50-G:100-F:2-A:1-B:2-C:1	132
81	IEEE14-P:25-G:25-F:3-A:0-B:1-C:1	133
82	IEEE14-P:25-G:25-F:3-A:0-B:2-C:1	133
83	IEEE14-P:25-G:25-F:3-A:1-B:1-C:1	133
84	IEEE14-P:25-G:25-F:3-A:1-B:2-C:1	133
85	IEEE14-P:25-G:50-F:3-A:0-B:1-C:1	134
86	IEEE14-P:25-G:50-F:3-A:0-B:2-C:1	134
87	IEEE14-P:25-G:50-F:3-A:1-B:1-C:1	134
88	IEEE14-P:25-G:50-F:3-A:1-B:2-C:1	134
89	IEEE14-P:50-G:100-F:3-A:0-B:1-C:1	134
90	IEEE14-P:50-G:100-F:3-A:0-B:2-C:1	135
91	IEEE14-P:50-G:100-F:3-A:1-B:1-C:1	135
92	IEEE14-P:50-G:100-F:3-A:1-B:2-C:1	135
93	IEEE14-P:25-G:25-F:4-A:1-B:1-C:1	136
94	IEEE14-P:25-G:25-F:4-A:1-B:2-C:1	136
95	IEEE14-P:25-G:25-F:4-A:1-B:5-C:1	136
96	IEEE14-P:25-G:50-F:4-A:1-B:1-C:1	136
97	IEEE14-P:25-G:50-F:4-A:1-B:2-C:1	137
98	IEEE14-P:25-G:50-F:4-A:1-B:5-C:1	137
99	IEEE14-P:50-G:100-F:4-A:1-B:1-C:1	137
100	IEEE14-P:50-G:100-F:4-A:1-B:2-C:1	137
101	IEEE14-P:50-G:100-F:4-A:1-B:5-C:1	137
102	IEEE30-P:25-G:25-F:0-A:1-B:1	138
103	IEEE30-P:25-G:50-F:0-A:1-B:1	138
104	IEEE30-P:50-G:100-F:0-A:1-B:1	138

105	IEEE30-P:25-G:25-F:1-A:1-B:1-C:0	138
106	IEEE30-P:25-G:25-F:1-A:1-B:2-C:0	139
107	IEEE30-P:25-G:25-F:1-A:1-B:5-C:0	139
108	IEEE30-P:25-G:50-F:1-A:1-B:1-C:0	139
109	IEEE30-P:25-G:50-F:1-A:1-B:2-C:0	139
110	IEEE30-P:25-G:50-F:1-A:1-B:5-C:0	139
111	IEEE30-P:50-G:100-F:1-A:1-B:1-C:0	140
112	IEEE30-P:50-G:100-F:1-A:1-B:2-C:0	140
113	IEEE30-P:50-G:100-F:1-A:1-B:5-C:0	140
114	IEEE30-P:25-G:25-F:2-A:1-B:1-C:0	141
115	IEEE30-P:25-G:25-F:2-A:1-B:1-C:1	141
116	IEEE30-P:25-G:25-F:2-A:1-B:1-C:2	141
117	IEEE30-P:25-G:25-F:2-A:1-B:2-C:1	141
118	IEEE30-P:25-G:50-F:2-A:1-B:1-C:0	142
119	IEEE30-P:25-G:50-F:2-A:1-B:1-C:1	142
120	IEEE30-P:25-G:50-F:2-A:1-B:1-C:2	142
121	IEEE30-P:25-G:50-F:2-A:1-B:2-C:1	142
122	IEEE30-P:50-G:100-F:2-A:1-B:1-C:0	142
123	IEEE30-P:50-G:100-F:2-A:1-B:1-C:1	143
124	IEEE30-P:50-G:100-F:2-A:1-B:1-C:2	143
125	IEEE30-P:50-G:100-F:2-A:1-B:2-C:1	143
126	IEEE30-P:25-G:25-F:3-A:0-B:1-C:1	144
127	IEEE30-P:25-G:25-F:3-A:0-B:2-C:1	144
128	IEEE30-P:25-G:25-F:3-A:1-B:1-C:1	144
129	IEEE30-P:25-G:25-F:3-A:1-B:2-C:1	144
130	IEEE30-P:25-G:50-F:3-A:0-B:1-C:1	145
131	IEEE30-P:25-G:50-F:3-A:0-B:2-C:1	145

132	IEEE30-P:25-G:50-F:3-A:1-B:1-C:1	145
133	IEEE30-P:25-G:50-F:3-A:1-B:2-C:1	145
134	IEEE30-P:50-G:100-F:3-A:0-B:1-C:1	145
135	IEEE30-P:50-G:100-F:3-A:0-B:2-C:1	146
136	IEEE30-P:50-G:100-F:3-A:1-B:1-C:1	146
137	IEEE30-P:50-G:100-F:3-A:1-B:2-C:1	146
138	IEEE30-P:25-G:25-F:4-A:1-B:1-C:1	147
139	IEEE30-P:25-G:25-F:4-A:1-B:2-C:1	147
140	IEEE30-P:25-G:25-F:4-A:1-B:5-C:1	147
141	IEEE30-P:25-G:50-F:4-A:1-B:1-C:1	147
142	IEEE30-P:25-G:50-F:4-A:1-B:2-C:1	148
143	IEEE30-P:25-G:50-F:4-A:1-B:5-C:1	148
144	IEEE30-P:50-G:100-F:4-A:1-B:1-C:1	148
145	IEEE30-P:50-G:100-F:4-A:1-B:2-C:1	148
146	IEEE30-P:50-G:100-F:4-A:1-B:5-C:1	148

LISTA DE ABREVIATURAS

- RSP** Restauração de Sistemas de Potência
- KBS** Knowledge Based Systems
- EMS** Energy Management System
- AEPS** American Electric Power System
- SGE** Sistemas de Gerenciamento de Energia
- IA** Inteligência Artificial
- SE** Sistemas Especialistas
- ODE** Operator Decision Environment
- ANEEL** Agência Nacional de Energia Elétrica
- CEEE** Companhia Estadual de Energia Elétrica
- PM** Programação Matemática
- SCADA** Supervisory Control And Data Acquisition
- SAD** Sistema de Apoio à Decisão - do inglês DSS: *Decision Support Systems*
- RBC** Raciocínio Baseado em Casos - do inglês CBR: *Case Based Reasoning*
- RNA** Redes Neurais Artificiais
- AG** Algoritmo Genético
- PFP** Programa de Fluxo de Potência
- PECO** Philadelphia Electric Company
- PSAP** Power System Analysis Package
- CDF** Common Data Format
- CEPEL** Centro de Pesquisas de Energia Elétrica
- IEEE** Institute of Electrical and Electronics Engineers
- BFS** Breath First Search

1 INTRODUÇÃO

A sociedade brasileira, a exemplo do que já aconteceu nos demais países industrializados, se vê frente a uma ameaça oriunda de seu próprio desenvolvimento: nossa produtividade e vida moderna são extremamente dependentes do fornecimento confiável de energia elétrica. A importância desta confiabilidade no fornecimento leva as empresas do setor, governos e outros órgãos relacionados a realizarem grandes esforços para sua concretização.

Ainda temos lembranças dos recentes blecautes que trouxeram transtornos ao país inteiro no início deste século. Existe farta literatura descrevendo a seqüência de eventos e os motivos que levaram às escuras diversos estados brasileiros (RADIOBRÁS, 2002), (TEREZA, 2002), (MELLONI, 2002). O tempo que foi necessário para que o abastecimento de energia voltasse ao normal alarmou a sociedade, e lhe deu clara noção da importância de esforços para evitar que o ocorrido se repetisse.

Não obstante, sabe-se que a política de privatização do setor transferiu a estas empresas a responsabilidade por desenvolver planos de correção de contingências, para responder às diversas situações a que estes serviços estão expostos atualmente. Desta forma, a Agência Nacional de Energia Elétrica (ANEEL), a partir de 1996 (pela Lei 9427, de 27/12/1996), passou a desempenhar um papel de influência neste cenário, onde ela regulamenta, inclusive, montantes que estas mesmas empresas devem aplicar em pesquisas e desenvolvimentos no setor, além de fiscalizar a utilização destes recursos.

Dentre as diversas áreas de pesquisa em que as empresas e agências brasileiras e de outros países vêm aplicando seus esforços está a de restauração de sistemas. A Restauração de Sistemas de Potência se preocupa com um dos diversos estados de operação de um sistema de energia elétrica. O estado restaurativo é o último de uma cadeia de estados que representam a situação de normalidade, ou não, do sistema. Esta cadeia vai de um estado denominado de nível seguro, passando por um quinto estado, denominado de emergência, onde há a ocorrência de violações de limites operativos. Nesta situação, a eliminação das violações só é possível mediante o corte de carga do sistema. Deste estado de emergência, passa-se para um último estado, chamado de nível restaurativo. Este se caracteriza pela inexistência de violações de limites operativos e por perda de fornecimento. É neste nível que se encontram os sistemas de energia elétrica após a

ocorrência de blecautes.

Pode-se, assim, definir o conceito de RSP como um conjunto de ações de controle capaz de levar um sistema de energia elétrica de um estado restaurativo ao de operação em estado seguro. Ou seja, a partir de um estado em que, possivelmente, a rede elétrica tem sua integridade prejudicada, e onde há cortes de fornecimento, deve-se, idealmente, atingir o estado de operação seguro, com fornecimento normal e com todos os limites de operação respeitados.

RSP é um problema multiestágios, estágios nos quais se procura restabelecer o abastecimento de energia a um determinado grupo de cargas consideradas prioritárias. O tempo total de desabastecimento representa a restrição mais importante. Ele ainda envolve um grande número de fatores, incluindo-se a identificação e o escalonamento dos recursos de geração disponíveis, os equipamentos de conexão disponíveis (linhas, transformadores, bancos *shunt*) e os respectivos limites operativos de todos estes recursos, como o de injeção de potências ativas e reativas, e de capacidade de transmissão das linhas e transformadores (SANTOS; GARCIA, 2004).

O nível de complexidade envolvido na solução deste problema tem levado os pesquisadores da área a procurar por ferramentas capazes de lidar com as dificuldades encontradas. A revisão da bibliografia da área mostra a crescente utilização de diversas técnicas de otimização, como Programação Linear, Pesquisa Tabu; e de técnicas de IA, passando pelo uso de Sistemas Especialistas (SE), Raciocínio Baseado em Casos - do inglês CBR: *Case Based Reasoning* (RBC), RNA, Sistemas Multi-Agentes, AG, etc.

O objetivo deste trabalho é o emprego de técnicas de IA em RSP, especificamente com a aplicação de AGs. Foi desenvolvido um algoritmo capaz de gerar estratégias de restauração *offline*. Estas estratégias são expressas por meio de operações elementares a serem executadas sobre os principais componentes do sistema de potência. As operações codificadas compreendem um conjunto básico de operações, como a ativação de componentes ou a energização de linhas de transmissão. O AG utiliza um Programa de Fluxo de Potência (PFP) para a avaliação da sua *Função Objetivo*. O emprego simultâneo de IA e *Métodos Analíticos*, por parte do PFP, classifica o sistema como um *sistema híbrido*.

Para a realização deste AG, foram criados os mecanismos necessários para uma nova representação. Essencialmente, este AG procura otimizar uma seqüência de operações, executadas sobre os componentes do sistema elétrico, avaliado por uma função objetivo. Para isto, as seqüências foram codificadas em uma lista de grupos de operações. Cada posição desta lista contém um grupo de operações elementares, executadas sobre o sistema de potência. A lista é populada a partir de um mecanismo especialmente desenvolvido para garantir que, inicialmente, apenas as operações corretas para cada estágio fossem introduzidas. O cruzamento dos indivíduos foi desenvolvido de forma a garantir que, após o cruzamento, as operações subseqüentes ao ponto de cruzamento fossem válidas para ambos os envolvidos neste operação.

O trabalho descrito nesta dissertação está estruturado da seguinte forma. No capítulo 2, tem-se um apanhado de conceitos, que auxiliarão o acompanhamento do que será exposto nos demais capítulos. No capítulo 3, faz-se uma revisão de diversos elementos da bibliografia da área, traçando-se uma linha temático-histórica dos desenvolvimentos e resultados obtidos por cada autor. A seguir, no Capítulo 4, descreve-se a Metodologia empregada no presente trabalho. A implementação desta metodologia será detalhada em capítulo a parte (Capítulo 5). Após a implementação, o Capítulo 6 detalha os experimentos realizados para o trabalho apresentado e os resultados obtidos com estes experimentos. A conclusão do trabalho será apresentada a seguir, no Capítulo 7.

2 CONCEITOS BÁSICOS

2.1 Sistemas de Potência

Ao conjunto de equipamentos utilizados para o fornecimento de energia elétrica aos consumidores chama-se de Sistema de Potência. Ele 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, num primeiro estágio, a energia obtida da natureza, sob diversas formas, seja transformada em energia elétrica e convenientemente convertida para o transporte, em um segundo estágio, através de um sistema de linhas de transmissão e, no estágio seguinte, de distribuição, até os 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.

O projeto de sistemas de potências é desenvolvido de forma que um número adequado de estações geradoras sejam interligadas por linhas de transmissão, de tal modo que *toda* a energia produzida por este conjunto possa ser utilizada em *toda* a extensão da área coberta pelo sistema. A figura 1 ilustra alguns dos componentes de um sistema de potência, desde a geração até os consumidores finais.

2.1.1 Componentes

Para atender aos consumidores, existe um grupo de três tipos de componentes principais em um Sistema de Potência: as *estações geradoras*, as *linhas de transmissão* e os *sistemas de distribuição* (EL-HAWARY, 2000). Esta divisão se dá devido à natureza da atividade executada em cada componente, e é estendida inclusive à divisão de atividades da exploração comercial da energia elétrica, onde também se tem três grupos distintos de empresas, as *Empresas Geradoras*, as *Transmissoras* e as *Distribuidoras*.

2.1.1.1 Estações Geradoras

Compreendem o conjunto de equipamentos utilizados para a geração da energia elétrica a partir de outro tipo de energia. Os tipos mais comuns são as centrais hidrelétricas, as centrais

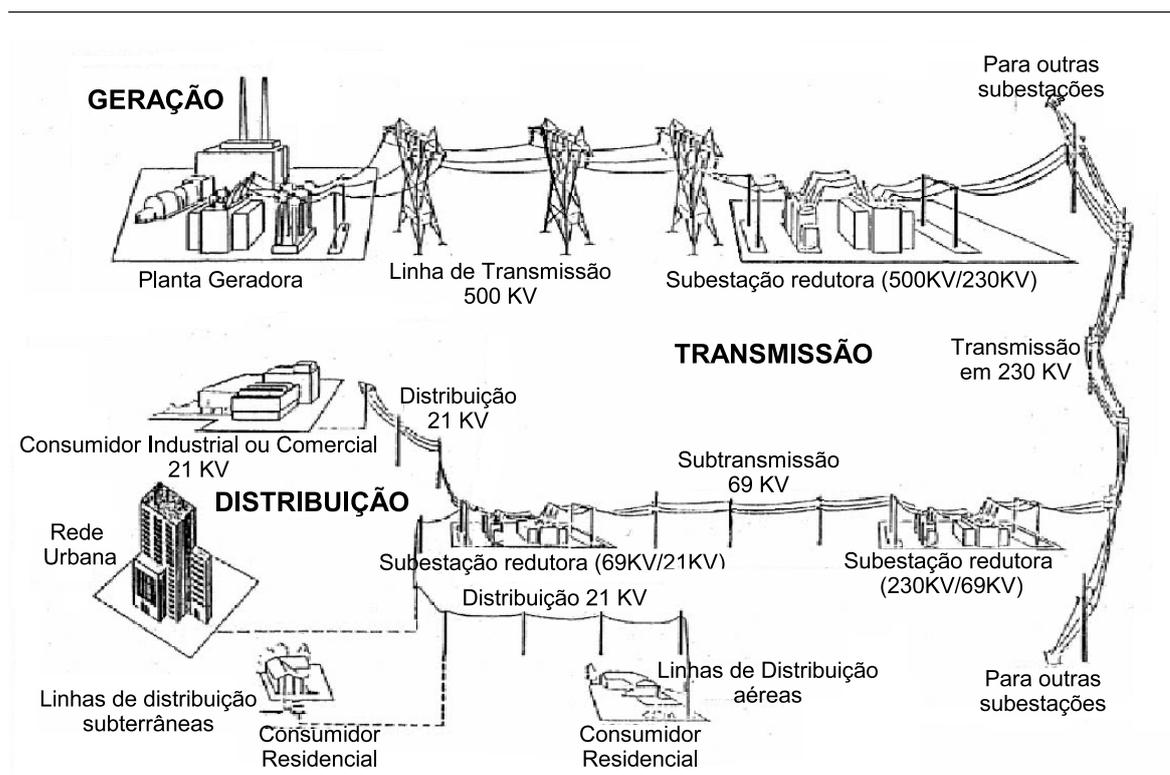


Figura 1: Esquema simplificado de um Sistema de Potência

termoelétricas, as eólicas e as usinas atômicas. As centrais hidrelétricas, de maior capacidade, situam-se afastadas dos grandes centros consumidores, requerendo extensas linhas de transmissão para abastecer as estações de distribuição.

No aspecto econômico, estas estações pertencem ou são administradas por uma empresa encarregada pela geração. Exemplos são a CEEE e a CGTEE, no Rio Grande do Sul.

2.1.1.2 Linhas de Transmissão

Uma linha de transmissão é formada por um conjunto de condutores elétricos capazes de permitir a passagem de grandes quantidades de energia com as menores perdas possíveis. Estes condutores estão, normalmente, suspensos em torres de transmissão localizadas ao longo da linha.

As linhas de transmissão podem ser classificadas em pelo menos dois aspectos. Um é o padrão da corrente conduzida, que classifica as linhas de transmissão em linhas de *Corrente Alternada* e *Corrente Contínua* (EL-HAWARY, 2000). Outro aspecto comumente utilizado para classificação de linhas é o nível da tensão existente entre os condutores da linha. Este nível de tensão foi evoluindo ao longo do tempo, e atualmente, está na faixa de algumas dezenas de milhares de volts até algumas centenas de milhares de volts e, desde os anos 90, atingindo mais

de um milhão de volts (EL-HAWARY, 2000).

A empresa que realiza as suas atividades no serviço de transmissão de energia elétrica pertence ao grupo das *Empresas Transmissoras*. No Rio Grande do Sul, um exemplo também é a CEEE.

2.1.1.3 Sistemas de Distribuição

O transporte da energia elétrica, das subestações até os consumidores finais, é feito pelos sistemas de distribuição. Nesta etapa, a tensão encontrada nas linhas é mais baixa, na ordem de centenas ou de alguns milhares a algumas dezenas de milhares de volts. A topologia das ligações entre as unidades desta etapa é predominantemente radial, com a existência de interconexões entre estes ramos radiais, o que permite a rápida restauração de energia em casos de faltas localizadas (LIU; LEE; VENKATA, 1988).

As empresas *Distribuidoras* são responsáveis por este último nível de serviço. Exemplos, no nosso estado, são a CEEE, a AES-Sul e a RGE.

2.1.2 Conceitos Básicos sobre Circuitos Elétricos

A lei fundamental empregada na análise de circuitos elétricos é a lei de Ohm, (equação 2.1).

$$v = R.i \quad (2.1)$$

onde v é a tensão vetorial, em volts, R é a resistência, em Ohms e i , a corrente, também vetorial, em ampères.¹

Em Sistemas de Potência, é mais freqüente expressar-se o fluxo de energia em valores de potência do que em valores de corrente. Como a potência é um produto direto da corrente circulando em um circuito ou elemento, esta substituição é razoável. Entretanto, como os sistemas de transmissão e distribuição são, predominantemente, de corrente alternada, algumas definições adicionais são necessárias.

A potência dissipada sobre um componente, em função da corrente elétrica ciculando por este e da tensão elétrica a qual está submetido, é dada pela equação (2.2), em valores instantâneos, além de expressa também em função da resistência, com o emprego de (2.1).

$$p(t) = v(t).i(t) = Ri^2(t) = \frac{v^2(t)}{R} \quad (2.2)$$

Os sistemas elétricos apresentam uma resposta à fontes de excitação (como fontes de tensão

¹As equações utilizadas neste trabalho seguem a convenção de representar grandezas vetoriais por caracteres maiúsculos e em negrito, e grandezas instantâneas, por caracteres minúsculos.

ou corrente). Esta resposta é composta de componentes transitórios e permanentes. Os componentes transitório tendem a zero, com o aumento do tempo, permanecendo apenas os permanentes. Este estado é conhecido como *regime permanente* (JOHNSON; HILBURN; JOHNSON, 1994).

Na engenharia elétrica, é comum a expressão de diversos valores de grandezas senoidais em termos de grandezas chamadas *fasores*, comumente utilizados na transformação das equações do domínio do tempo para o da frequência, onde podemos operar com equações algébricas, ao invés de equações diferenciais (JOHNSON; HILBURN; JOHNSON, 1994). Esta representação é válida apenas para o *regime permanente*. Considere-se, por exemplo, uma tensão senoidal dada por:

$$v(t) = V_m \cos(\omega t + \theta) \quad (2.3)$$

onde V_m representa a amplitude da tensão, ω representa a frequência angular e θ , o *ângulo de fase* ou simplesmente *fase*.

A partir da fórmula de Euler, esta tensão pode ser expressa apenas por sua amplitude V_m e pelo ângulo θ (JOHNSON; HILBURN; JOHNSON, 1994):

$$V = V_m e^{j\theta} = V_m \angle \theta \quad (2.4)$$

Define-se a relação entre a tensão e a corrente como sendo a impedância do circuito, aqui expressa em valores fasoriais (JOHNSON; HILBURN; JOHNSON, 1994):

$$Z = \frac{V}{I} = |Z| \angle \theta_Z = \frac{V_m}{I_m} \angle (\theta - \phi) \quad (2.5)$$

Considere-se, então, uma impedância $Z = |Z| \angle \theta_Z$ sobre a qual se aplica uma tensão $v(t)$, ao final, expressa fasorialmente por (EL-HAWARY, 2000):

$$v(t) = V_m \cos(\omega t) = V_m \angle \theta \quad (2.6)$$

A corrente instantânea circulando pela impedância Z é

$$i(t) = I_m \cos(\omega t + \phi) \quad (2.7)$$

onde ϕ é a defasagem da corrente em relação a tensão $v(t)$ e I_m , a amplitude da corrente, é dada por:

$$I_m = \frac{V_m}{|Z|} \quad (2.8)$$

A potência instantânea é dada por

$$p(t) = v(t)i(t) = V_m I_m [\cos(\omega t) \cos(\omega t + \phi)] \quad (2.9)$$

que reduz-se para

$$p(t) = \frac{V_m I_m}{2} [\cos \phi + \cos(2\omega t + \phi)] \quad (2.10)$$

Como a média de uma função senoidal, ao longo de um ciclo $2\omega t$, é zero, a potência média é dada por:

$$p_m = \frac{V_m I_m}{2} \cos(\phi) \quad (2.11)$$

Os valores rms de tensão e correntes, dados por $V_m = \sqrt{2}V_{(rms)}$ e $I_m = \sqrt{2}I_{(rms)}$, também fornecem a potência média, dada por:

$$p_m = V_{(rms)} I_{(rms)} \cos \phi \quad (2.12)$$

Num circuito onde estão presentes reatâncias e resistências, uma componente da corrente neste circuito transporta energia que é armazenada e descarregada, periodicamente, na reatância. Esta energia armazenada, ao circular pelo circuito, não altera sua potência média (EL-HAWARY, 2000).

A potência média em um circuito é chamada *potência ativa* (P), e a potência que alimenta os componentes reativos, *potência reativa* (Q), e são dadas por:

$$P = VI \cos \phi \quad (2.13)$$

$$Q = VI \sin \phi \quad (2.14)$$

onde V e I são os valores rms de tensão e corrente, e ϕ o ângulo de defasagem entre as duas grandezas. P é medido em Wats(W) e Q, em VAR.

Na figura 2 representam-se algumas das grandezas vistas. Em (a) pode-se ver a tensão V, a corrente I e a potência P, além do ângulo de defasagem ϕ , entre V e I. Em (b), vêm-se as potências P, Q e a potência instantânea total p(t).

Na figura 3 pode-se ver, em (a), a projeção do vetor V sobre o vetor I, considerando-se o ângulo de defasagem ϕ . Em (b), tem-se o vetor I projetado sobre o vetor V. Ao se multiplicar estes vetores pelas respectivas projeções do vetor oposto, ou pelo próprio vetor oposto, obtêm-se os valores de P, Q e VI. Os vetores V e I definem o que é conhecido como *triângulo das potências*.

Define-se, ainda, uma grandeza chamada *potência aparente* ou *complexa*, designada pela letra S, que tem como componentes P e Q, e que é dada por:

$$\begin{aligned} S &= P + jQ \\ &= VI(\cos \phi + j \sin \phi) \end{aligned} \quad (2.15)$$

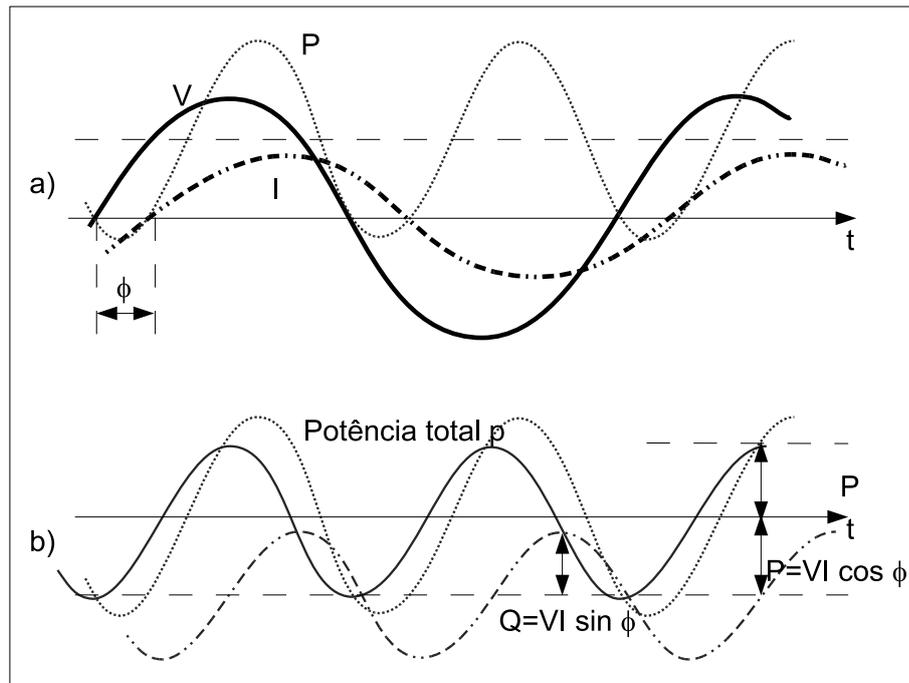


Figura 2: (a) Diagrama que ilustra o comportamento da tensão V , da corrente I e da Potência P . (b) Relação entre P , Q e a potência $p(t)$ (EL-HAWARY, 2000).

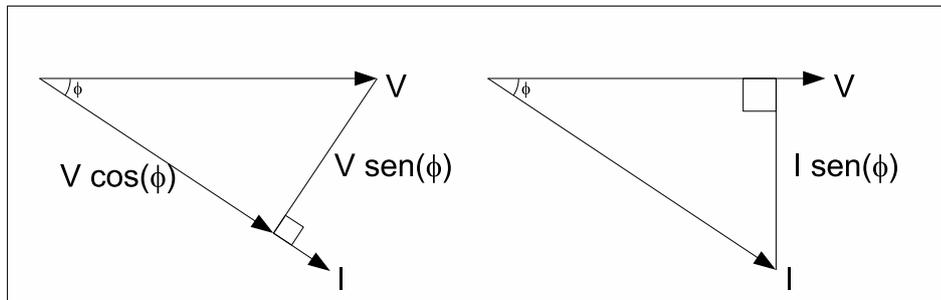


Figura 3: Projeções vetoriais da tensão V sobre a corrente I (a) e da corrente I sobre a tensão V (b) (EL-HAWARY, 2000).

Aplicando a identidade de Euler:

$$S = VIe^{j\phi}$$

$$S = VI\angle\phi$$

2.1.3 Representação Unifilar

Os Sistemas de Potência são, em sua maioria, de corrente alternada e trifásicos. Uma representação gráfica completa de um sistema à três fases seria de difícil elaboração, devido ao grande número de conexões e conjuntos de linhas a serem representados. Entretanto, em situações de equilíbrio entre as fases, estes sistemas podem ser representados e tratados matematicamente com uma das fases e com o neutro. Esta representação pode ser ainda mais simplificada com

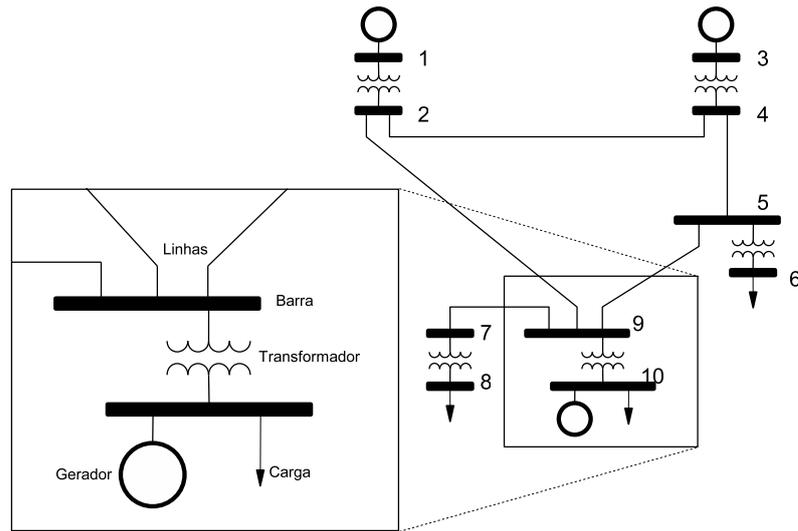


Figura 4: Exemplo de um diagrama unifilar. Representam-se neste exemplo geradores, barras, transformadores, cargas e linhas de transmissão

a supressão do neutro e com a utilização de símbolos padronizados para representar os componentes do sistema. Este diagrama simplificado é chamado de *diagrama unifilar* (EL-HAWARY, 2000). Um exemplo deste tipo de diagrama encontra-se na figura 4.

A função principal de um diagrama unifilar é fornecer uma representação concisa, objetiva, do sistema de potência. O nível de detalhes representado neste diagrama varia segundo o problema considerado, podendo por isto omitir a representação de componentes não relevantes do sistema na área de estudo.

2.1.4 Normalização - Grandezas por Unidade [p.u.]

Existe a possibilidade de se representar todas as grandezas de um sistema em relação a um valor base, ou referencial. Esta representação será feita em percentagem ou em valor por unidade [*p.u.*], e sua determinação segue a forma geral:

$$valor[p.u.] = \frac{valor\ real}{V_{ref}} \quad (2.16)$$

onde V_{ref} é o valor tomado como referência, ou base, de mesma dimensão que o valor real

A representação em valores por unidade é preferida à outras, pois não necessita maiores manipulações durante as operações matemáticas realizadas na análise de circuitos (EL-HAWARY, 2000). Assim, algumas das equações básicas da eletricidade podem ser reescritas para o cálculo de seus valores de base, conforme as equações (2.17) e (2.18)), apresentadas a seguir:

$$Corrente\ base\ (A) = \frac{potência\ aparente\ base\ (kVA)}{tensão\ base\ (kV)} \quad (2.17)$$

$$Impedância\ base\ (\Omega) = \frac{tensão\ base\ (V)}{corrente\ base\ (A)} \quad (2.18)$$

A escolha da base para estas grandezas é feita de modo a reduzir ao mínimo o trabalho de cálculo, de forma que conduza à obtenção de valores de tensão e corrente próximos de 1. Outros motivos para utilizar as grandezas em valores por unidade são: a) os fabricantes de equipamentos usualmente especificam os parâmetros de seus produtos desta forma; b) os valores por unidade de equipamentos de um mesmo tipo variam dentro de uma faixa estreita, ao passo que os valores normais variam amplamente; c) facilidade no manuseio de valores em cálculos de circuitos que apresentam transformadores; d) não existe variação nos valores de impedância dos transformadores em relação à maneira com que são interligados, o que não acontece quando se utilizam os valores ôhmicos normais.

Depois de obtidos os valores de base com as equações (2.17) e (2.18)), pode-se determinar os valores das principais grandezas de um sistema, com o auxílio de (2.16), que levam às equações:

$$I[p.u.] = \frac{\text{Corrente real}}{\text{Corrente base}} \quad (2.19)$$

$$Z[p.u.] = \frac{\text{impedância real em ohms}}{\text{impedância base em ohms}} \quad (2.20)$$

Neste trabalho não será necessária a conversão de unidades para o sistema *p.u.*, pois os valores já estão expressos nesta unidade.

2.1.5 Linhas de Transmissão

2.1.5.1 Quadripolos

Uma rede elétrica pode ser composta por 2 ou mais terminais. Entretanto, muitas das redes presentes em sistemas de potência são compostas por quatro terminais, distribuídos em dois pares. Um quadripolo típico pode representar, por exemplo, o modelo de um transformador ou linha elétrica.

A aplicação dos quadripolos aos sistemas de energia elétrica permite o estudo de um problema importante, chamado de *problema de transmissão*, onde se deseja saber a tensão e a corrente em um par do quadripolo em função de valores presentes no outro par (EL-HAWARY, 2000).

O problema de transmissão emprega um conjunto de equações da forma:

$$\begin{aligned} V_s &= AV_r + BI_r \\ I_s &= CV_r + DI_r \end{aligned} \quad (2.21)$$

onde A, B, C e D são chamados *parâmetros de transmissão*. Ou, em forma matricial:

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \quad (2.22)$$

Um quadripolo pode ser simétrico, caso ele possa ter seus terminais invertidos sem alterar o comportamento do sistema a que pertence. As linhas de transmissão são um caso de quadripolo simétrico. Para satisfazer o critério de simetria, um quadripolo deve ter

$$A = D \quad (2.23)$$

Na análise de sistemas de potência, um quadripolo simétrico de grande importância é o do circuito π simétrico (figura 5) (EL-HAWARY, 2000).

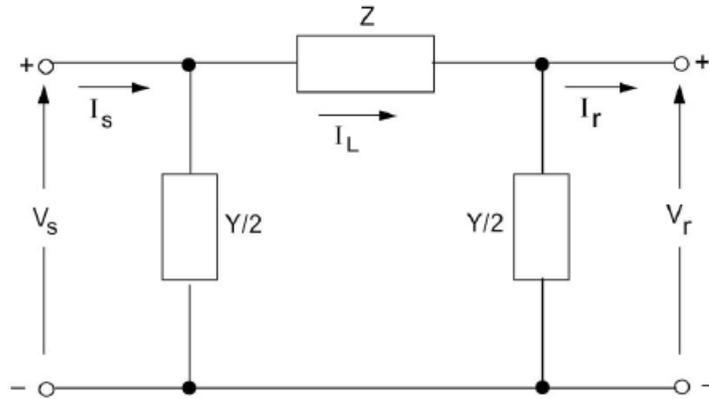


Figura 5: Circuito π simétrico

Da figura, pode-se verificar que:

$$A = \left(1 + \frac{ZY}{2}\right) \quad (2.24)$$

$$B = Z \quad (2.25)$$

$$C = Y \left(1 + \frac{ZY}{4}\right) \quad (2.26)$$

$$D = A \quad (2.27)$$

O emprego dos parâmetros de transmissão permite uma fácil manipulação de associações série de quadripolos. A figura 6 mostra dois quadripolos conectados em série.

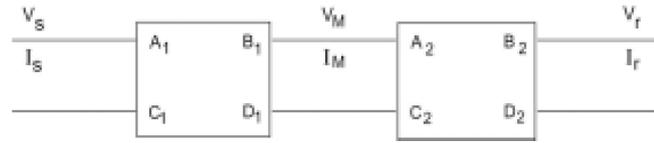


Figura 6: Quadripolos associados em série

Pode-se escrever que:

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} V_M \\ I_M \end{bmatrix} \quad (2.28)$$

$$\begin{bmatrix} V_M \\ I_M \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \quad (2.29)$$

$$\begin{bmatrix} V_s \\ I_s \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} V_r \\ I_r \end{bmatrix} \quad (2.30)$$

Esta equação se estende para os casos de n quadripolos conectados em série, com a obtenção de um conjunto de parâmetros de transmissão equivalente, através de multiplicações matriciais.

2.1.5.2 Modelo de Linhas de Transmissão Longa

Quando se consideram linhas de transmissão de comprimentos arbitrários, tem-se interesse em obter o equivalente desta linha a partir de seus dados unitários. Na figura 7 está representada uma linha, com detalhes de uma posição incremental (dx) qualquer, ao longo de seu comprimento, a uma distância x de sua extremidade.

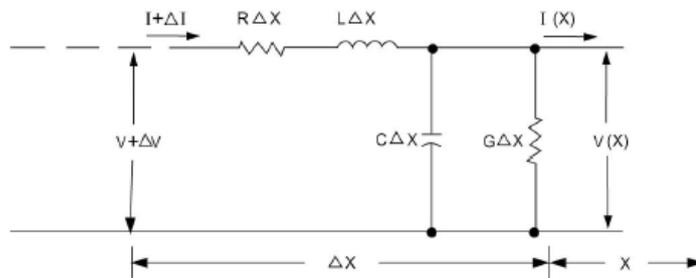


Figura 7: Modelo de uma linha de transmissão

Assumindo-se que esta linha está operando em regime senoidal permanente, e com balanço entre fases, faz-se uma análise das grandezas ΔV e ΔI . Aplicando-se as Leis de Kirchhoff, obtém-se (EL-HAWARY, 2000):

$$\Delta V = I(x)z\Delta x \quad (2.31)$$

$$\Delta I = V(x)y\Delta x \quad (2.32)$$

Introduz-se uma constante de propagação v , definida por:

$$v = \sqrt{zy} \quad (2.33)$$

A impedância equivalente, por unidade de comprimento, é dada por:

$$z = R + j\omega L \quad (2.34)$$

E a admitância *shunt*, por unidade de comprimento:

$$y = G + j\omega C \quad (2.35)$$

R e L são a resistência e a indutância série por unidade de comprimento, e G, a condutância e a capacitância *shunt*, em relação ao neutro, por unidade de comprimento.

As equações diferenciais que modelam a linha, quando $x \rightarrow 0$, são expressas por:

$$\frac{\partial^2 V}{\partial x^2} = v^2 V \quad (2.36)$$

$$\frac{\partial^2 I}{\partial x^2} = v^2 I \quad (2.37)$$

A solução da equação diferencial ordinária (2.36) é

$$V(x) = A_1 \exp(vx) + A_2 \exp(-vx) \quad (2.38)$$

Derivando $V(x)$ em relação a x , obtém-se a tensão sobre o segmento dx . Com esta obtém-se $I(x)$:

$$I(x) = \frac{A_1 \exp(vx) + A_2 \exp(-vx)}{Z_c} \quad (2.39)$$

Com a introdução da constante Z_c , a *impedância característica* da linha, dada por:

$$Z_c = \sqrt{\frac{z}{y}} \quad (2.40)$$

As constantes A_1 e A_2 podem ser avaliadas a partir das equações (2.38) e (2.39), em $x=0$:

$$\begin{aligned} V(0) &= A_1 + A_2 \\ Z_c V(0) &= A_1 - A_2 \end{aligned}$$

que permitem escrever:

$$V(x) = \frac{1}{2} \{ [V(0) + Z_c I(0)] \exp(vx) + [V(0) - Z_c I(0)] \exp(-vx) \} \quad (2.41)$$

$$I(x) = \frac{1}{2} \left\{ \left[I(0) + \frac{V(0)}{Z_c} \right] \exp(vx) + \left[I(0) - \frac{V(0)}{Z_c} \right] \exp(-vx) \right\} \quad (2.42)$$

que fornecem a tensão e a corrente a qualquer distância x a partir do final da linha.

Utilizando-se funções hiperbólicas:

$$\begin{aligned}\sinh \phi &= \frac{\exp(\phi) - \exp(-\phi)}{2} \\ \cosh \phi &= \frac{\exp(\phi) + \exp(-\phi)}{2}\end{aligned}$$

pode-se reescrever as equações (2.41) e (2.42) como:

$$V(x) = V(0) \cosh(vx) + Z_c I(0) \sinh(vx) \quad (2.43)$$

$$I(x) = I(0) \cosh(vx) + \frac{V(0)}{Z_c} \sinh(vx) \quad (2.44)$$

a partir das quais pode-se chegar aos seguintes parâmetros de transmissão:

$$\begin{aligned}A(x) &= \cosh vx \\ B(x) &= Z_c \sinh(vx) \\ C(x) &= \frac{1}{Z_c} \sinh(vx) \\ D(x) &= A(x) = \cosh vx\end{aligned} \quad (2.45)$$

2.2 Fluxo de Potência Resumido

A análise do fluxo de potência é essencial para o estudo de sistemas de potência e forma a base para outras análises. Por exemplo, ela desempenha um papel central no planejamento de expansões de sistemas. Além disto, ela é empregada em diversos estudos realizados durante a operação dos Sistemas de Potência.

O problema do Fluxo de Potência pode ser definido como:

Para uma dada Rede de Potência, com cargas cujos valores de potência complexa são conhecidos, e com um dado conjunto de especificações ou restrições, encontrar a solução para quaisquer tensões de barras desconhecidas e gerações não especificadas e, finalmente, para o fluxo de potência complexa nos componentes da rede. (GRIGSBY; HANSON, 2001)

Adicionalmente, as perdas em componentes individuais e na rede completa também podem ser calculadas, além da realização de uma verificação no sistema, localizando componentes em sobrecarga e tensões elétricas fora das tolerâncias especificadas.

Os passos necessários para a realização de um estudo de fluxo de potência são:

- Determinar os valores dos componentes passivos da rede.
- Determinar a posição e os valores de todas as cargas complexas.
- Determinar as restrições e especificações de geração.
- Desenvolver um modelo matemático descrevendo o fluxo de potência na rede.
- Resolver o perfil de tensões elétricas da rede.
- Resolver o fluxo de potência e as perdas da rede.
- Verificar as violações de restrições que ocorreram.

2.2.1 Matriz de Admitâncias

Para a etapa que consiste em desenvolver o modelo matemático do fluxo de potência da rede, a primeira providência a ser tomada é a obtenção da *Matriz de Admitâncias* das barras do sistema. Esta é 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 o sistema. 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 (GRIGSBY; HANSON, 2001):

- A admitância dos elementos *shunt* conectados entre o nodo k e a referência é adicionada à posição (k,k) da matriz.
- A admitância de um elemento conectado entre os nodos j e k é 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.

Estas regras darão lugar a duas equações (2.51 e 2.50), que serão apresentadas a seguir, com um exemplo (EL-HAWARY, 2000).

Exemplo:

Considere-se o sistema da figura 8, onde encontram-se alguns geradores e cargas. As barras 1, 2 e 4 possuem capacidade de geração. As barras 1, 2 e 3 possuem cargas. A barra 3 não possui geração, ao passo que a barra 4 não possui carga, apenas geração.

Utilizando o equivalente π para cada uma das linhas representadas, obtém-se o diagrama da figura 9. Neste diagrama, encontram-se as admitâncias do sistema, ignorando-se os geradores e as cargas (EL-HAWARY, 2000).

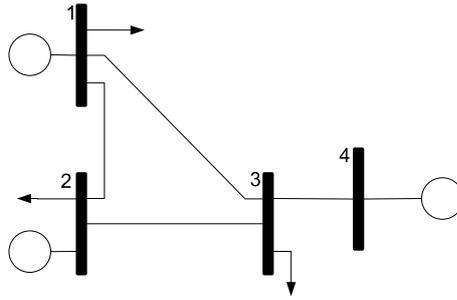


Figura 8: Sistema utilizado no exemplo de Análise de Fluxo de Potência

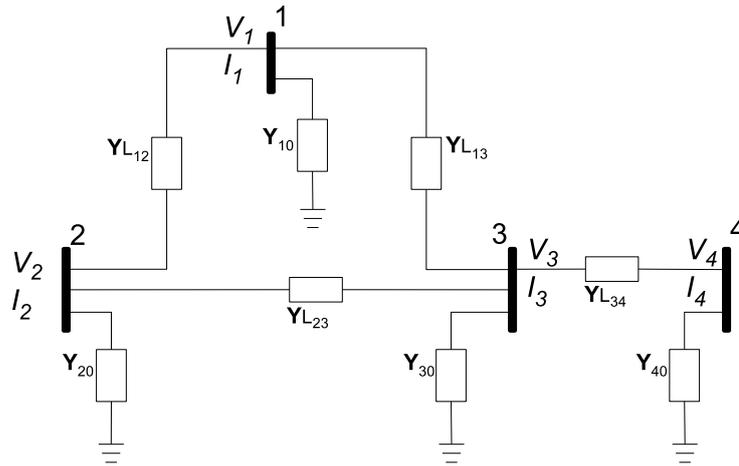


Figura 9: Circuito equivalente do modelo utilizado neste exemplo, desconsideradas suas fontes e cargas. As admitâncias Y_{10} , Y_{20} , Y_{30} e Y_{40} são admitâncias shunt, conectadas entre o respectivo barramento e o neutro.

Expandindo-se a equação da Lei dos Nós (JOHNSON; HILBURN; JOHNSON, 1994) para cada barra, obtém-se:

$$\begin{aligned}
 I_1 &= V_1 Y_{10} + (V_1 - V_2) Y_{L12} + (V_1 - V_3) Y_{L13} \\
 I_2 &= V_2 Y_{20} + (V_2 - V_1) Y_{L12} + (V_2 - V_3) Y_{L23} \\
 I_3 &= V_3 Y_{30} + (V_3 - V_1) Y_{L13} + (V_3 - V_2) Y_{L23} + (V_4 - V_3) Y_{L34} \\
 I_4 &= V_4 Y_{40} + (V_4 - V_3) Y_{L34}
 \end{aligned} \tag{2.46}$$

Empregando-se o descrito para a obtenção da Matriz de Admitâncias (2.2.1), pode-se determinar os elementos da matriz como (EL-HAWARY, 2000):

$$\begin{aligned}
Y_{11} &= Y_{10} + Y_{L_{12}} + Y_{L_{13}} \\
Y_{22} &= Y_{20} + Y_{L_{12}} + Y_{L_{23}} \\
Y_{33} &= Y_{30} + Y_{L_{13}} + Y_{L_{23}} + Y_{L_{34}} \\
Y_{44} &= Y_{40} + Y_{L_{34}} \\
Y_{12} = Y_{21} &= -Y_{L_{12}} \\
Y_{13} = Y_{31} &= -Y_{L_{13}} \\
Y_{23} = Y_{32} &= -Y_{L_{23}} \\
Y_{34} = Y_{43} &= -Y_{L_{34}}
\end{aligned} \tag{2.47}$$

onde o subscripto L_{jk} indica o valor da admitância da linha entre as barras j e k da figura 9.

Manipulando (2.46) e substituindo (2.47) nos leva a:

$$\begin{aligned}
I_1 &= Y_{11}V_1 + Y_{12}V_2 + Y_{13}V_3 + 0V_4 \\
I_2 &= Y_{21}V_1 + Y_{22}V_2 + Y_{23}V_3 + 0V_4 \\
I_3 &= Y_{31}V_1 + Y_{32}V_2 + Y_{33}V_3 + Y_{34}V_4 \\
I_4 &= 0V_1 + 0V_2 + Y_{43}V_3 + Y_{44}V_4
\end{aligned} \tag{2.48}$$

Como $Y_{14} = Y_{41} = 0$, pois as barras 1 e 4 não possuem conexão, assim como $Y_{24} = Y_{42} = 0$, pode-se escrever a equação (2.48) em notação matricial como (EL-HAWARY, 2000):

$$I_{BUS} = Y_{BUS}V_{BUS} \tag{2.49}$$

onde:

$$I_{BUS} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix}, V_{BUS} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} e Y_{BUS} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} \\ Y_{21} & Y_{22} & Y_{23} & Y_{24} \\ Y_{31} & Y_{32} & Y_{33} & Y_{34} \\ Y_{41} & Y_{42} & Y_{43} & Y_{44} \end{bmatrix}$$

Assim, estes resultados e a própria Matriz de Admitâncias pode ser generalizada para o caso de n barras como (EL-HAWARY, 2000):

$$Y_{ii} = \sum_{j=0}^n Y_{L_{ij}} \tag{2.50}$$

$$Y_{ij} = Y_{ji} = -Y_{L_{ij}} \tag{2.51}$$

Como, em geral, se especifica nas barras a Potência Aparente S_i , ou suas Componentes P_i e Q_i , ao invés das correntes, usamos as identidades $I_i^* = \frac{S_i}{V_i}$ e $S = P + jQ$ para obtermos (EL-HAWARY, 2000)

$$\frac{P_i - jQ_i}{V_i^*} = \sum_{j=1}^n Y_{ij} V_{ij} \quad (i = 1, \dots, n) \quad (2.52)$$

e re-escrevermos a equação (2.48) na forma de potência como (EL-HAWARY, 2000):

$$P_i - jQ_i = (V_i^*) \sum_{j=1}^n (Y_{ij} V_j) \quad (2.53)$$

2.2.2 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. Assim, as barras podem ser classificadas de acordo com quais destas quantidades serão especificadas (GRIGSBY; HANSON, 2001).

2.2.2.1 Barra de Referência - *Slack Bus* ou *Swing Bus*

A barra de referência do sistema é uma única barra para a qual se especificam a tensão V e o ângulo A, e onde as potências P e Q são desconhecidas. Deve-se escolher uma barra do sistema que possua fontes de potências ativa e reativa, já que esta barra deve ter capacidade de suprir os valores das potências ativa e reativa determinadas pelo Fluxo de Potência. A melhor escolha da barra de referência para um sistema depende de experiência e de conhecimento do sistema em questão.

2.2.2.2 Barra de Carga ou Barra P-Q

Uma Barra de Carga possui ambas as potências ativa e reativa especificadas. Apesar de que estas barras possam conter também geradores, com valores especificados de potências ativa e reativa, comumente, convém se especificar como barras P-Q as barras que possuam qualquer valor de potência complexa injetada.

2.2.2.3 Barra de Tensão Controlada ou Barra P-V

Todas as barras que possuam os valores de tensão e de potência real injetada são classificadas como barras P-V. A potência reativa injetada é uma variável que será calculada pelo Fluxo de Potência. Entretanto, elas devem ter uma fonte de potência reativa variável, tal como um gerador.

2.2.3 O Fluxo de Potências Generalizado

Para o caso genérico de n barras, pode-se empregar as equações da Matriz de Admitâncias (2.50 e 2.51) e a equação (2.53), repetida abaixo. Esta última equação define a *Equação Básica do Fluxo de Potência*. Esta é empregada diretamente para todas as barras de carga, e após as devidas manipulações, para as barras de tensão controlada (GRIGSBY; HANSON, 2001).

$$P_i - jQ_i = (V_i^*) \sum_{j=1}^n (Y_{ij} V_j) \quad (2.53)$$

De posse de um modelo matemático, pode-se seguir determinando as tensões do sistema, resolvendo-se a equação (2.53) para as $n-1$ 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-se o Fluxo de Potências entre as barras e as perdas de todo o sistema.

2.2.4 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 o Método Desacoplado Rápido, pois o Newton-Raphson tem sido o método mais difundido, e em função do Desacoplado Rápido ter sido empregado no programa de Fluxo de Potência utilizado neste trabalho (item 5.3), e deste ser uma extensão do próprio método de Newton-Raphson (GRIGSBY; HANSON, 2001).

2.2.4.1 Método de Newton-Raphson

O Método de Newton-Raphson (NR) tem sido empregado na solução de equações não-lineares em diversas áreas. O algoritmo será aqui descrito com conjunto de duas equações genéricas, por questões de simplicidade. Mas ele pode ser estendido para um número arbitrário de equações (GRIGSBY; HANSON, 2001).

Considere-se o conjunto de equações não lineares (2.54) e (2.55)

$$f_1(x_1, x_2) = k_1 \quad (2.54)$$

$$f_2(x_1, x_2) = k_2 \quad (2.55)$$

Considerem-se $x_1^{(0)}$ e $x_2^{(0)}$ como soluções inexatas para as equações (2.54) e (2.55) e $\Delta x_1^{(0)}$ e $\Delta x_2^{(0)}$ como correções necessárias para que $x_1^{(0)}$ e $x_2^{(0)}$ tornem-se soluções exatas. Assim, as equações $x_1^{(0)}$ e $x_2^{(0)}$ podem ser reescritas como:

$$f_1 \left(x_1^{(0)} + \Delta x_1^{(0)}, x_2^{(0)} + \Delta x_2^{(0)} \right) = k_1 \quad (2.56)$$

$$f_2 \left(x_1^{(0)} + \Delta x_1^{(0)}, x_2^{(0)} + \Delta x_2^{(0)} \right) = k_2 \quad (2.57)$$

Expandindo-se estas equações em Séries de Taylor em torno das estimativas (GRIGSBY; HANSON, 2001):

$$f_1 \left(x_1^{(0)}, x_2^{(0)} \right) + \left. \frac{\partial f_1}{\partial x_1} \right|^{(0)} \Delta x_1^{(0)} + \left. \frac{\partial f_1}{\partial x_2} \right|^{(0)} \Delta x_2^{(0)} + T.O.S. = k_1 \quad (2.58)$$

$$f_2 \left(x_1^{(0)}, x_2^{(0)} \right) + \left. \frac{\partial f_2}{\partial x_1} \right|^{(0)} \Delta x_1^{(0)} + \left. \frac{\partial f_2}{\partial x_2} \right|^{(0)} \Delta x_2^{(0)} + T.O.S. = k_2 \quad (2.59)$$

onde o índice (0) nas derivadas parciais indica a avaliação destas com os valores das estimativas iniciais $x_1^{(0)}$ e $x_2^{(0)}$, e *T.O.S.* indicam 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 (GRIGSBY; HANSON, 2001) e re-escrever as equações (2.58) e (2.59) em forma matricial como:

$$\begin{bmatrix} \left. \frac{\partial f_1}{\partial x_1} \right|^{(0)} & \left. \frac{\partial f_1}{\partial x_2} \right|^{(0)} \\ \left. \frac{\partial f_2}{\partial x_1} \right|^{(0)} & \left. \frac{\partial f_2}{\partial x_2} \right|^{(0)} \end{bmatrix} \begin{bmatrix} \Delta x_1^{(0)} \\ \Delta x_2^{(0)} \end{bmatrix} = \begin{bmatrix} k_1 - f_1 \left(x_1^{(0)}, x_2^{(0)} \right) \\ k_2 - f_2 \left(x_1^{(0)}, x_2^{(0)} \right) \end{bmatrix} \quad (2.60)$$

A matriz de derivadas parciais da equação (2.60) 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 ordens 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 (GRIGSBY; HANSON, 2001).

Em resumo, termos corrigidos para uma ℓ -ésima iteração são dados pela equação (2.61) e a estimativa de solução inicial é atualizada com os resultados da equação (2.62) (GRIGSBY; HANSON, 2001).

$$\begin{bmatrix} \Delta x_1^{(\ell)} \\ \Delta x_2^{(\ell)} \end{bmatrix} = \begin{bmatrix} \left. \frac{\partial f_1}{\partial x_1} \right|^{(\ell)} & \left. \frac{\partial f_1}{\partial x_2} \right|^{(\ell)} \\ \left. \frac{\partial f_2}{\partial x_1} \right|^{(\ell)} & \left. \frac{\partial f_2}{\partial x_2} \right|^{(\ell)} \end{bmatrix}^{-1} \begin{bmatrix} k_1 - f_1 \left(x_1^{(\ell)}, x_2^{(\ell)} \right) \\ k_2 - f_2 \left(x_1^{(\ell)}, x_2^{(\ell)} \right) \end{bmatrix} \quad (2.61)$$

$$x^{(\ell+1)} = x^{(\ell)} + \Delta x^{(\ell)} \quad (2.62)$$

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 inseridas 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 (2.61) especificamente para o Fluxo de Potência (GRIGSBY; HANSON, 2001), tem-se a equação (2.63):

$$\begin{bmatrix} \underline{\Delta A}^{(\ell)} \\ \underline{\Delta V}^{(\ell)} \end{bmatrix} = \begin{bmatrix} \left. \frac{\partial P}{\partial A} \right|^{(\ell)} & \left. \frac{\partial P}{\partial V} \right|^{(\ell)} \\ \left. \frac{\partial Q}{\partial A} \right|^{(\ell)} & \left. \frac{\partial Q}{\partial V} \right|^{(\ell)} \end{bmatrix}^{-1} \begin{bmatrix} \underline{P}_{prev} - \underline{P}^{(\ell)} \\ \underline{Q}_{prev} - \underline{Q}^{(\ell)} \end{bmatrix} \quad (2.63)$$

onde as barras nos valores das variáveis da equação (2.63) indicam vetores, estendendo o caso de duas equações visto como exemplo para o caso geral da Equação de Potência. Os valores indicados com o índice $prev$ representam as previsões de potências ativa e reativas injetadas no sistema. Os valores de $P^{(\ell)}$ e $Q^{(\ell)}$ representam os valores de injeções de potências ativa e reativa, calculados 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 tensões 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 soluções atinjam uma tolerância aceitável ou sejam executados um dado número máximo de iterações.

Uma vez que se tenha obtido uma solução adequada, pode-se prosseguir com o cálculo das injeções de potências reativas nas barras P-V e da injeção de potência complexa na barra de referência (GRIGSBY; HANSON, 2001).

2.2.4.2 Método Desacoplado Rápido

O Método Desacoplado Rápido simplifica o de Newton-Raphson, explorando as fortes relações existentes 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 altas tensões reais (GRIGSBY; HANSON, 2001). Nestes sistemas, o fluxo de potência ativa é muito mais sensível às mudanças nos ângulos de fase do que às 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 é 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 (2.63) como:

$$\underline{\Delta A}^{(\ell)} = [B']^{-1} [\underline{P}_{prev} - \underline{P}^{(\ell)}] \quad (2.64)$$

$$\underline{\Delta V}^{(\ell)} = [B'']^{-1} [\underline{Q}_{prev} - \underline{Q}^{(\ell)}] \quad (2.65)$$

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, mantidas 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 neste método, ele apresenta boas propriedades de convergência. O Método Desacoplado Rápido passou a ser largamente difundido, por ser computacionalmente menos intensivo que o de Newton-Raphson. Este método é empregado no programa de Fluxo de Potência que o presente trabalho utiliza (item 4.3).

2.3 Algoritmos Genéticos

Os Algoritmos Genéticos (AGs) são métodos heurísticos utilizados na obtenção de soluções aproximadas para problemas de difícil solução, utilizando a aplicação dos princípios de biologia evolucionária à Ciência da Computação. Eles empregam técnicas derivadas da biologia, tais como herança, mutação, seleção natural e recombinação, para a evolução de soluções que, a cada geração do algoritmo, apresentam resultados mais eficazes.

Sua implementação típica é realizada nos moldes de simulações computacionais nas quais uma população de indivíduos implementam representações abstratas (*genomas*) de soluções candidatas para um problema de simulação, normalmente compostas por um único *cromossomos*. Esta população evolui em direção à soluções melhores, partindo de indivíduos aleatoriamente gerados, e passa por uma série de gerações. O *genoma*, ou conjunto de *cromossomos*, de cada indivíduo de uma geração é avaliado por uma *função objetivo* que avalia sua performance (função de *fitness*). Em cada geração, um conjunto de indivíduos é selecionado da população, baseando-se no seu índice de performance e modificado para gerar uma nova população que passará por mais uma iteração do algoritmo. A representação abstrata destas soluções é normalmente realizada com *cromossomos* compostos de cadeias de bits (*genes*), de forma binária, apesar de poder empregar outras representações (MITCHELL, 1999).

O desenvolvimento dos AG foi liderado por Holland, nos anos 60 e 70, que teve a colaboração de colegas e alunos da Universidade de Michigan. Em seu livro *Adaptation in Natural and Artificial Systems*, Holland apresentou o AG como uma abstração da evolução biológica e criou uma base teórica para esta adaptação, no contexto dos AGs (HOLLAND, 1999 apud MITCHELL, 1999).

2.3.1 Componentes dos Algoritmos Genéticos

Apesar de não haver consenso em uma definição rigorosa de Algoritmos Genéticos, que seja aceita por todos da comunidade de computação evolucionária, pode-se dizer que todos os

métodos que se denominam como AGs possuem pelo menos os elementos seguintes em comum: populações de *cromossomos*, seleção de acordo com a performance (*fitness*), *cruzamento* para produzir novos filhos e a *mutação* dos novos filhos (MITCHELL, 1999).

2.3.1.1 Cromossomos

Os cromossomos em uma população de um AG tipicamente são modelados como cadeias de *bits*. Cada posição (*locus*) no cromossomo equivale a um *gene* que, em uma representação binária, tem dois possíveis valores (*alelos*): 0 ou 1. Cada cromossomo pode ser visualizado como um ponto no espaço de procura de soluções candidatas. Como já foi mencionado, o AG irá processar populações compostas de cromossomos, substituindo sucessivas populações por outras (MITCHELL, 1999).

2.3.1.2 Função Objetivo (*Fitness*)

Cada indivíduo de uma dada população tem seu *cromossomo* avaliado por uma função de performance que lhe concede um índice. Este índice depende do quão bem o cromossomo deste indivíduo resolve o problema cuja solução se procura encontrar. Esta função pode ter diversos tipos de objetivos, de acordo com a área de aplicação. Alguns dos mais comuns são minimização ou maximização de funções.

2.3.1.3 Operadores de AGs

Em sua forma mais simples, o AG envolve três operadores genéticos (MITCHELL, 1999):

- *Seleção*: este operador seleciona cromossomos da população para a reprodução. Quanto melhor o índice de performance do cromossomo, maiores as suas chances de seleção.
- *Cruzamento*: este operador seleciona, randomicamente, *loci* que indicarão pontos de intercâmbio de cromossomos entre os pais selecionados para a reprodução. Desta forma, os filhos de um par de cromossomos terão cromossomos de ambos os ancestrais, cruzados de forma randômica pelo operador.
- *Mutação*: este operador muda o alelo de um *locus* de um cromossomo, selecionados randomicamente, a uma taxa controlada, usualmente bastante baixa.

2.3.1.4 Schemata

A teoria formulada por Holland assume que, em um nível de descrição genérico, os AGs funcionam pela descoberta, ênfase e recombinação de bons *blocos construtivos* de soluções, de

forma altamente paralela. O ponto chave da descrição é de que boas soluções tendem a ser formadas por bons blocos construtivos, combinações de valores de *bits* que conferem maiores performances nas cadeias de bits onde eles estão presentes.

Holland introduziu a noção de esquemas (*schemata*) para formalizar a noção destes blocos construtivos. Um esquema é um conjunto de cadeias de *bits* que pode ser descrito por um modelo feito de dígitos, 0 ou 1, e por asteriscos indicando indiferença aos valores (*wild cards*). Por exemplo, o esquema $H = 1 * * * * 1$ representa o conjunto de todas as cadeias de 6 *bits* que comecem e terminem por 1. As cadeias que se enquadrarem neste modelo são chamadas instâncias de H. O esquema H possui dois *bits* definidos, ou diz-se ter ordem 2.

2.3.2 Descrição geral do algoritmo

O funcionamento de um algoritmo genético elementar pode ser entendido com auxílio da figura 10.



Figura 10: Algoritmo genético elementar

O primeiro passo do algoritmo é inicializar os *cromossomos* de uma população. Esta inicialização irá gerar um conjunto de possíveis soluções para o problema a ser resolvido. Inicia-se um ciclo de atividades, cujo primeiro passo é a seleção de indivíduos da população para a reprodução. Esta seleção é realizada com base nos valores obtidos por cada indivíduo quando submetidos a uma função objetivo (função de *fitness*).

Os indivíduos selecionados são cruzados entre si, gerando novos indivíduos para a população seguinte. Alguns dos novos indivíduos irão passar por mutações, à taxas controladas. Os novos indivíduos são introduzidos na nova população.

Os passos executados dentro deste ciclo são controlados por critérios de parada. Estes critérios, normalmente, envolvem ou o número de gerações ou critérios de performance, como a convergência dos melhores resultados.

3 ESTADO DA ARTE

A seguir serão revisados alguns dos elementos da bibliografia que serviram para instrumentar o projeto e o desenvolvimento desta proposta.

A revisão passa pelo processo de restauração de sistemas de potência, a partir dos primeiros estudos e tentativas de compreensão dos problemas envolvidos. Depois, ela segue com a revisão de alguns trabalhos baseados em processos analíticos. A aplicação de técnicas de Inteligência Artificial (IA) será descrita por conjuntos de artigos, com ênfases distintas. O grupo mais numeroso será o que descreve a utilização de Sistemas Especialistas (SE) no assunto, mas também haverá a revisão de trabalhos desenvolvidos com o uso de Redes Neurais Artificiais (RNA), Sistemas Multi-Agentes, Raciocínio Baseado em Casos - do inglês CBR: *Case Based Reasoning* (RBC), etc.

3.1 Restauração de Sistemas

Prabhakara *et al.* (JOHNSON; PRABHAKARA *et al.*, 1982), em um dos primeiros documentos tratando sobre as atividades comumente empregadas, em situações de desabastecimento, por um conjunto maior de companhias, ilustraram o que seria um programa de restauração genérico, para aplicação em quaisquer empresas. Um levantamento feito pelos autores entre algumas companhias mostrou que poucas empresas possuíam um plano de restauração formal completo. Pelo contrário, haviam partes de planos.

Dentre os planos existentes, destacava-se a existência de adaptabilidade a diferentes magnitudes de blecautes, da necessidade de alguma documentação formal, da descrição de prioridades de manobras. Também foram discriminadas algumas funções ou procedimentos auxiliares para fornecer informações úteis durante a restauração, como o *monitoramento*, que irá determinar quão rapidamente se tomará ciência da ocorrência do blecaute, da *avaliação* da situação, que irá determinar quais as proporções da ocorrência tratada, bem como sua classificação e a escolha dentre um conjunto de opções de planos, do *controle*, que irá evitar maiores danos aos equipamentos logo após o evento. Este controle poderia ser automático, remoto ou manual. As necessidades de *comunicação* também foram ressaltadas pelos autores como sendo essenciais para a eficiência do reabastecimento, consistindo tanto dos meios de comunicação e da informação

transmitida. Os meios de *transporte* também foram mencionados por permitirem que o pessoal de campo pudesse ser rapidamente deslocado aos diversos pontos de atuação. Por fim, os autores destacaram a importância do *pessoal* envolvido no processo, indicando a importância da supervisão, de treinamentos e de experiência para o sucesso e eficácia da implantação de quaisquer planos de restauração.

A restauração de sistemas, vista como um modo de operação extraordinário, requer planejamento cuidadoso e treinamento dos operadores. Adibi e Fink (ADIBI; FINK, 1994) escreveram um artigo que elucidou bem os problemas associados às operações de restauração, identificando diversas tarefas e relacionando com cada uma destas um conjunto de preocupações e problemas possivelmente associados. Após estas exposições, os autores indicaram um procedimento para se desenvolver um plano de restauração composto de um conjunto de passos que culmina com a geração de estratégias e táticas, que podem ser divididas em estratégias genéricas, e uma dentre duas estratégias distintas de escolha das companhias operadoras.

As estratégias genéricas são implementadas na preparação da rede para sua restauração, como por exemplo, abrir todos os interruptores para facilitar o diagnóstico do estado do sistema. A estratégia *build-down*, onde se energiza toda a rede de distribuição principal, para depois se passar a restauração balanceada de cargas e demais geradores. Complementarmente, a estratégia *build-up* consiste na divisão da rede completa em redes menores, contendo cada uma ao menos um gerador com capacidade de auto-inicialização (*black-start*), seguida da restauração de geradores adicionais, da sincronização das subredes geradas e da restauração seletiva de cargas (*load pick up*). Os autores ainda continuam mostrando a importância de quesitos auxiliares, como documentação adequada, treinamento, auditorias e atualizações dos planos obtidos. Eles indicaram a possibilidade de se estabelecer procedimentos gerais e uma lista de instruções para aprimorar a rápida restauração destes sistemas. Planos detalhados poderiam ser produzidos para atender às necessidades específicas destes sistemas.

O Grupo de Trabalho em Restauração de Sistemas de Potência (RSP) do Subcomitê de Operação de Sistemas, ligado à Sociedade de Sistemas de Potência do IEEE, emitiu um relatório, (ADIBI; FINK *et al.*, 1992b), no qual exploram considerações especiais a serem observadas durante as restaurações. Este relatório cobre um conjunto de cinco itens, passando pelo processamento de alarmes, operações de chaveamento, seqüência ótima de inicialização de unidades geradoras, sistemas de transmissão subterrâneos e finalizando com as limitações e capacidades de telecomunicações durante as restaurações. Para alguns destes itens foram apresentadas soluções que incluíam o auxílio por computador. Em especial, o capítulo versando sobre a seqüência ótima de inicialização de unidades geradoras levanta pontos bastante importantes, sobre os princípios a serem seguidos quando da priorização destas unidades, algumas vezes por questões estratégicas, outras por questões técnicas ou até de segurança dos equipamentos.

Huang *et alli* (HUANG; AUDETTE; HARRISON, 1995) apresentaram um método sistemático para o planejamento de restauração de sistemas de potência. A restauração na Hydro-Québec é executada de acordo com um conjunto de instruções de emergência detalhadas, que tem sua base preparada *offline* por planejadores de operações. A empresa divide seu sistema em cinco redes para a restauração, energizadas independente e simultaneamente, e aplica um conjunto de princípios para restaurar seu sistema: a) energizar as redes tão rápido quanto possível, sem necessariamente restaurar cargas neste estágio; b) minimizar o número de ajustes em variáveis de controle de estágio para estágio; c) sincronizar as redes após a energização; e d) restaurar as cargas em pequenos incrementos. Cada rede é energizada em estágios, cada estágio correspondendo ao fechamento de uma chave de circuito e a consequente energização de um componente de rede, como um transformador, uma linha, um reator *shunt* ou uma carga. Os autores seguem indicando os processos de restauração considerados convencionais e apresentando o seu método sistemático para o planejamento da restauração, que se utiliza de um programa de computador que provê uma solução otimizada para as variáveis de controle que satisfazem as restrições de operação, como os limites de potência e tensões.

Apesar de existirem diversas técnicas para a restauração de sistemas de potência, atestadas pelo grande número de artigos publicados sobre o assunto, todas as técnicas apontam para a existência de um conjunto de orientações gerais para seus desenvolvimentos, que independem das ferramentas utilizadas. Algumas destas orientações são descritas no artigo de Ancona (ANCONA, 1995), que as resumiu em 5 objetivos principais: a) restaurar com segurança, b) restaurar suave e deliberadamente, c) reduzir o tempo de restauração total, d) minimizar efeitos adversos para o público e e) manter flexibilidade para responder aos problemas. As metas principais da restauração devem ser, primeiramente, a reconstrução de um sistema estável e depois a restauração das cargas ainda não atendidas. Ancona indicou um guia genérico para os operadores, uma seqüência geral para auxiliar na preparação do plano de restauração de um sistema de potência específico e um guia para auxiliar na avaliação e melhoria das condições pré-existentes para as restaurações.

A obtenção da segurança durante o processo de restauração deve atender a uma série de restrições impostas pelo próprio sistema de potência, sua topologia, aspectos estáticos e dinâmicos. Adibi e Milanicz apresentaram em seu artigo (ADIBI; MILANICZ, 1995) a avaliação da performance das operações realizadas nos centros de operações. A performance foi indicada pelo percentual relativo entre as operações consideradas corretas e apropriadas e as que podiam ser definidas como corretas mas inadequadas. Os autores apontaram como principais causas de operações corretas mas inadequadas as mudanças do sistema de potência, e sua implicação na operação dos relês, equipamentos utilizados nas manobras de restaurações. Isto porque durante a restauração, o sistema passa por mudanças contínuas e fica, assim, sujeito a estas operações de relês corretas, porém inadequadas.

No Brasil, os blecautes dos últimos anos foram analisados por Gomes(GOMES, 2004). Ele utilizou de indicadores como as áreas afetadas, as cargas perdidas (ou não atendidas), o tempo de restauração, a severidade das perdas econômicas. Após rever diversas causas e lições aprendidas com cada evento descrito, o autor indica uma série de observações e recomendações. Das observações destaca-se a necessidade de programas de treinamento dos operadores nas mais diversas áreas e a simulação de cada distúrbio conhecido em computadores para garantir a compreensão de cada um deles, suas causas e possíveis alternativas, auxiliados pela análise dos modelos matemáticos associados. Das recomendações do autor, há diversas medidas ligadas aos aspectos físicos do sistema e também a implementação de ferramentas de software para avaliar o grau de segurança deste sistema. A simulação de falhas e o estudo de estratégias de restauração auxiliaria na definição das que se poderia indicar como mais adequada para cada situação, mantendo-se distância das opções sabidamente arriscadas para o sistema. Um conjunto de medidas específicas para a otimização do tempo de restauração foi sugerido, que passam pela revisão do planejamento do sistema brasileiro, sob a ótica da sua restauração, uso de ilhas locais, identificação de novos geradores para *black-start* e de subestações vitais para o sistema, além de uma rotina de testes periódicos para estes dispositivos e subestações e da criação de um programa de treinamentos para os operadores.

O processo de restauração do sistema brasileiro foi revisado pela (Operador Nacional do Sistema Elétrico) ONS, a pedido da Agência Nacional de Energia Elétrica (ANEEL), em especial nas áreas mais afetadas pelos últimos blecautes (Rio de Janeiro, São Paulo e Mato Grosso do Sul). Seus procedimentos de restauração prioritários foram verificados, e foram propostas restaurações alternativas para o caso de indisponibilidades nos processos prioritários (GOMES; LIMA; GUARINI, 2004). Neste artigo, os autores descrevem o processo de restauração do Sistema de Potência Brasileiro, descrevendo seus dois estágios: o de restauração fluente, onde não há a necessidade de comunicação entre as partes envolvidas, e o de restauração coordenada, que só é iniciado uma vez que um conjunto de pré-condições foi alcançado. É no processo de restauração fluente que o ONS apresentou um conjunto de recomendações a ser seguido, que garantem que após a geração de algumas áreas geo-elétricas se possa iniciar o processo de restauração coordenada. O artigo ainda segue mostrando o resultado de uma série de estudos de restauração realizados após cada manobra de restauração, e termina descrevendo o procedimento operacional principal para uma restauração de potência no Rio de Janeiro.

3.1.1 Métodos Analíticos

Wu e Monticelli(WU; MONTICELLI, 1988) propuseram um *framework* conceitual para o monitoramento e a avaliação durante as restaurações de sistemas, utilizando ferramentas analíticas. A proposta foi baseada na utilização de ferramentas semelhantes já existentes no monitoramento e avaliação de segurança, onde as funções de estimação de estado, análise de observabilidade e

identificação de dados corrompidos, entre outras, poderiam ser modificadas para sua aplicação nesta outra área. Para isto, os autores identificam as ferramentas analíticas já utilizadas em segurança de sistemas de potência que poderiam ser transportadas para o monitoramento de RSP, além de pesquisar as ferramentas analíticas que poderiam ser utilizadas para atividades de avaliação na restauração de sistemas, identificando suas áreas de aplicação efetivas. Eles ainda propuseram um Sistema Especialista para coordenar estas ferramentas durante a avaliação de restaurações. Os autores indicaram, como possíveis usos para a ferramenta, estudos de planejamento fora-de-linha, como simulador para o treinamento de operadores e como ambiente de operação em tempo real.

Adibi *et alli*(ADIBI; BORKOSKI; KAFKA, 1994) tratam dos requisitos de ferramentas analíticas para restauração. Neste artigo, os autores descrevem as necessidades de 9 tipos destas ferramentas, passando por programas de fluxo de potência, estabilidade transiente, dinâmica de longo termo, transientes de tensão, curto-circuitos, transientes eletromagnéticos e outros, até culminar em um programa para a coordenação de restaurações. Cada um deles possui um conjunto específico de requisitos, dos quais alguns foram analisados no artigo.

Santos e Garcia(SANTOS; GARCIA, 2004) mostraram um método de restauração analítica, projetado para realizar a restauração por estágios para atender cargas com prioridades diferentes. O método desenvolvido empregava um novo modelo de fluxo de potência linearizado, programação linear, uma expansão do conceito de rede fictícia e o próprio modelo convencional de fluxo de potência. O conceito de rede fictícia já fora utilizado anteriormente por Monticelli *et alli* para o planejamento da expansão de sistemas, e consistia na substituição dos ramos desconectados por impedâncias extremamente elevadas. O fluxo de potência linearizado consiste em tratar-se todo o sistema de potência como um circuito de corrente contínua.

O método fôra dividido em duas fases, uma tratando do problema da demanda ativa, e a outra da demanda reativa. Os autores realizaram testes sob o sistema IEEE-14 (sistema de referência adotado em diversos artigos) e no sistema SB-810, que representa os subsistemas elétricos das regiões Sul e Sudeste do Brasil. Em ambos os testes documentados, os autores relataram a restauração dos respectivos sistemas em dois estágios.

3.1.2 Sistemas Especialistas em RSP

O primeiro esforço para tratamento do assunto de RSP com o uso de técnicas de Inteligência Artificial foi o trabalho de Sakaguchi e Matsumoto(SAKAGUCHI; MATSUMOTO, 1983), onde os autores propuseram a criação de um sistema baseado em conhecimento (*Knowledge Based Systems (KBS)*) para o problema do controle da restauração. Os autores ilustraram a arquitetura básica da sua solução, indicando uma árvore de contexto para o KBS, e produzindo um sistema contendo um conjunto de 16 regras que, a partir da informação de uma lista de

barras a serem restauradas, determinava a seqüência de operações necessária para retornar estas barras ao seu estado funcional, respeitando os seus limites operacionais, que eram obtidos de um programa de *Fluxo de Potência* externo.

Nas vantagens apresentadas incluíam-se a programação heurística, a modularidade do conhecimento e a universalidade. A programação heurística foi apontada como solução para problemas desta natureza, onde a topologia de conexões de rede é mutável. A modularidade foi apontada como facilitadora para a compreensão do sistema, onde cada regra era única e isoladamente compreensível. A universalidade foi apresentada com o conceito de que bastaria a criação de uma nova base de conhecimento para se tratar de um sistema distinto, pois havia a universalidade na heurística. As desvantagens apresentadas foram a da incapacidade de validação completa do sistema, em parte causada pela própria modularidade apontada como vantagem, e a baixa velocidade da computação simbólica, que poderia ser resolvido, na época, com novas versões de compiladores ou máquinas mais rápidas.

Wollenberg e Sakaguchi (F.WOLLENBERG; SAKAGUCHI, 1987) analisaram a necessidade de aplicação de tecnologias de IA na operação de sistemas de potência. Os autores apresentaram as diferenças e as vantagens do uso destas técnicas sobre os programas numéricos convencionais. As principais diferenças apresentadas incluem a flexibilidade, a alta performance, no sentido de que os Sistemas Especialistas reproduzir o nível de performance de uma pessoa com experiência reconhecida no domínio de problema, e a compreensibilidade, já que os Sistemas Especialistas podem inclusive demonstrar a linha de raciocínio utilizada.

Os autores também discutiram sobre como implementar um sistema de Inteligência Artificial em um Energy Management System (EMS), apresentando opções de linguagens de programação. Ainda se apresentaram áreas de aplicações para sistemas de IA, com ênfase em Problemas de Controle em Tempo Real (processamento de alarmes, operações de chaveamento, controle de tensão, controle de RSP), Planejamento de Operações (planejamento de fluxos de potência, comprometimento de unidades) e Treinamento de Operadores (em ensino de pessoal e construção de cenários).

A análise e avaliação do conhecimento de um especialista foram a ponte que Komai *et alli* (KOMAI; MATSUMOTO; SAKAGUCHI, 1988) encontraram entre os métodos numéricos e baseados em conhecimento. Eles propuseram comparações entre os resultados obtidos em RSP entre os dois métodos, através da quantificação de suas eficiência e eficácia. Para isto, utilizaram de um algoritmo utilizando o método de otimização para a solução numérica e um SE e compararam as soluções obtidas por ambos os processos. Eles observaram que na maior parte dos casos, os resultados obtidos pelo método de IA foram idênticos aos numéricos e que, como o espaço de procura do método baseado em conhecimento era muito menor do que o do numérico, este era muito mais eficaz para achar uma solução ótima. Em outros casos, o resultado obtido pelo

método numérico não era operacionalmente possível, ao passo que os resultados apresentados pelos métodos de IA, nestes casos, eram sempre possíveis, embora pudessem não ser resultados ótimos. A conclusão dos autores foi que o conhecimento de um especialista pode guiar para uma solução ótima, com eficiência, em quase todos os casos, e que este conhecimento é eficaz para obter uma solução possível para problemas complexos. Como proposta, eles indicaram que uma integração entre os métodos pode trazer ainda mais vantagens para a solução baseada em conhecimento.

Em sistemas de distribuição, Liu *et alli* (LIU; LEE; VENKATA, 1988) propuseram a aplicação de técnicas de SE para sua restauração. Nestes sistemas, existe uma estrutura radial que permite o reabastecimento de zonas dos sistemas através do acionamento de caminhos alternativos, possíveis graças a existência de chaves, que normalmente estão abertas, mas que podem ser fechadas para a geração destes novos caminhos. Porém, em falhas de maiores dimensões, existem mais sub circuitos que precisam ser realimentados, o que complica a viabilidade das manobras, uma vez que se precisa atender a uma série de restrições de cada sistema. Foi neste cenário que os autores desenvolveram o SE, construído sobre uma base de 180 regras que podem ser utilizadas por operadores no planejamento de suas restaurações. Entre estas regras, os autores procuram atender a restrição adicional de encontrar configurações com as menores perdas em linhas possíveis.

Os autores validaram os resultados obtidos pelo sistema através de simulações computacionais. Eles também concluíram que a velocidade de computação do resultado foi satisfatória, obtendo planos em alguns segundos, o que julgaram adequado para esta aplicação. Complementaram suas conclusões frisando que o sistema possuía regras genéricas, ou portáteis para outros sistemas de distribuição, uma vez que elas não possuíam dependência em aspectos intrínsecos à rede analisada.

Kojima *et alli* (KOJIMA *et al.*, 1989b) discutiram sobre a geração de um protótipo de SE para a restauração de sistemas de potência, que teve seu processo dividido em duas fases, baseadas nas condições do sistema para sua restauração. Uma fase de energização, onde o principal objetivo era o de fornecer energia suficiente para a restauração, e uma fase de atendimento de cargas, onde se determinava uma configuração de sistema que não iria causar sobrecargas. Dependendo das condições dos sistemas, poderiam se executar diversas operações de restauração simultaneamente. Na fase de energização, a seqüência de operações proposta foi de verificação da situação do sistema após a falha; a determinação de uma política de restauração; a verificação da execução de operações, na qual se investigava o número de subestações que podiam ser restauradas simultaneamente; e na geração de procedimentos operacionais. Na fase de atendimento de cargas, os procedimentos adotados foram: a determinação de uma política de restauração; a determinação de métodos de eliminação de sobrecargas; e também na geração de procedimentos operacionais.

Kojima *et alli*(KOJIMA *et al.*, 1989a), em outro artigo, também propuseram um método de RSP bem como de um *framework* para orientação dos operadores durante a restauração. Para esta ferramenta, os autores pretendiam criar um sistema capaz de lidar com eventos ainda não vistos, aceitar diretivas dos operadores em qualquer etapa da restauração, além de ter boa performance em tempo real. Para atender a estas exigências, os autores indicaram o uso de técnicas baseadas em conhecimento e métodos analíticos combinados. A orientação dada aos operadores foi dividida em dois grupos, um com instruções de chaveamento, para o qual foram apresentadas 8 regras capazes de gerar uma seqüência; e outro com instruções de despacho de cargas, onde se levavam em consideração as características de geração de energia (sua resposta dinâmica) e a existência de prioridades na lista de cargas a ser restaurada. Foi apresentado um exemplo de uso do sistema, com um sistema contendo 36 geradores e 48 subestações. Para este exemplo, o protótipo apresentou resultados satisfatórios.

Kirschen e Volkmann(KIRSCHEN; VOLKMANN, 1991) introduziram uma separação na sua proposta. Eles propuseram um SE com uma estrutura hierárquica, separando a seleção do próximo objetivo (raciocínio estratégico) do desenvolvimento do plano para alcançar este objetivo (raciocínio tático). Esta separação visava habilitar o sistema a adaptar suas orientações no caso de ocorrerem problemas durante a restauração. O sistema implementado se chamava *Assistente de Restauração*, e visava a redução do tempo necessário para se completar a etapa de re-energização de subestações com as unidades de geração. Esta redução seria possibilitada, de acordo com os autores, por que o sistema iria apresentar ao operador planos de restauração que refletissem o estado real do sistema; também por retirar do operador e assumindo as tarefas ordinárias da restauração e por prover ao operador uma imagem clara do estado da restauração para lhe facilitar o processo de tomada de decisões.

O SE consistia, na verdade, de três módulos. O primeiro era o módulo estratégico, onde se criava um conjunto de objetivos, cada qual visando a ligação de uma linha de transmissão ou de segurança entre duas estações. Estes objetivos dariam lugar a um conjunto de tarefas. O segundo, o módulo tático, traduziria estas tarefas em listas de ações, que indicariam as manobras a serem executadas para concretizar cada tarefa. O terceiro módulo era o módulo de controle de execução, com o qual o operador poderia deixar a responsabilidade da execução do plano criado com os módulos anteriores. A execução à cargo do sistema evitaria uma série de erros ou atrasos que são possíveis quando se faz o controle manual das operações. Para isto, o SE interage com o operador nos níveis tanto de sistema quanto de equipamentos.

Matsumoto *et alli*(MATSUMOTO *et al.*, 1992) criaram um importante relatório onde reuniram os resultados dos artigos mais importantes até a época sobre RSP. Eles fizeram uma pesquisa sobre os sistemas de restauração, traçando um histórico sobre as publicações sobre o assunto. Ainda descreveram as técnicas baseadas em conhecimento para RSP, dividindo-as em três grupos: metodologia de auxílio operacional, metodologia em nível restaurativo e metodologia de

desenvolvimento de sistemas. O primeiro grupo relaciona-se com instrução ou suporte à decisão para os operadores. O grupo seguinte produziria planos e procedimentos para a restauração de sistemas. O último grupo de técnicas indicaria como se produziria um sistema baseado em conhecimento para a restauração de sistemas (como, por exemplo, se faria a aquisição de conhecimento e a integração com os ambientes já existentes).

Do mesmo Grupo de Trabalho em RSP é a autoria de outro artigo que mostrou novas tendências nos trabalhos de restauração (ADIBI; FINK *et al.*, 1992a). Neste artigo, surgem os conceitos de *restauração automatizada*, *restauração auxiliada por computador* e *restauração cooperativa*. Na primeira, o operador apenas observa as ações do sistema, e até a data ainda não havia tido nenhuma proposta de implementação. No segundo modelo, o operador é responsável por cada passo da restauração, e o sistema lhe fornece apoio informativo. A maior parte dos sistemas se enquadrava nesta definição. A terceira classe delegava responsabilidades alternadamente ao operador e ao sistema computacional, de acordo com a eficiência de cada um em cada processo.

No mesmo artigo, outro capítulo discutia o papel dos SE, e chegou à conclusão de que eles: deveriam prover assistência aos operadores, deixando-o responsável pelas principais decisões; proviam procedimentos sistemáticos para a rápida tomada de decisões em situações de stress e numa ampla gama de circunstâncias; que o raciocínio lógico assistido por programas algorítmicos podiam ser utilizados para identificar e priorizar as ações de controle; que a sua integração nos Sistemas de Gerenciamento de Energia (SGE)s teriam o potencial de redução significativa do tempo necessário para a restauração dos serviços; poderiam ser utilizados para planejar e testar procedimentos e estratégias de restauração.

Adibi *et alli* (ADIBI; KAFKA; MILANICZ, 1994), em outro artigo oriundo do mesmo subcomitê supra-citado, tratam especificamente sobre os requisitos dos SE em restaurações. Nele, os autores discretizam três períodos de tempo: integração das fontes de energia iniciais, de 30 a 60 minutos; integração de geração e transmissão (de 3 a 4 horas); minimização de clientes sem atendimento (de 8 a 12 horas). Em cada um destes períodos, os autores identificam quais os requisitos necessários à utilização de SE, em função das respectivas atividades e restrições predominantes.

No primeiro período, se deu ênfase a priorização no atendimento das interconexões críticas, que incluíam, por exemplo, unidades de geração térmica, que não sendo reativadas em intervalos de 30 a 60 minutos, deveriam ter sua entrada em operação postergada, por restrições técnicas. No segundo período, o principal papel de um SE seria o de estabelecer caminhos entre todas as unidades em operação e as ainda fora de operação, criando caminhos seguros entre elas, determinando seqüências de chaveamentos, fazendo previsões de tempo, verificando o atendimento de todas as restrições e, se possível, progressivamente enumerando os caminhos alternativos. Já

no terceiro período, as atribuições principais de um SE seriam: ordenação das grandes unidades geradoras térmicas e nucleares; energização das linhas de tensões alta e super-alta; abastecendo grandes cargas de baixa tensão e ativando os *corredores de energia* com os sistemas de potência adjacentes.

Liu *et alli*(LIOU; LIU; CHU, 1995) apresentaram um SE que passou a considerar a existência de *corredores de energia* como fonte suplementar à capacidade de geração própria de um sistema. A inclusão destes recursos externos aumentou a capacidade de geração e possibilitou o reabastecimento mais rápido de clientes. O trabalho contribuiu com a identificação dos papéis críticos desempenhados pelos corredores de energia durante as restaurações, com o desenvolvimento de um módulo que considerasse a disponibilidade destes recursos e pela implementação deste módulo integrado ao sistema já existente nas instalações dos autores.

Na França, Kostier *et alli*(KOSTIE *et al.*, 1998) apresentaram o projeto conceitual de um Sistema de Apoio à Decisão - do inglês DSS: *Decision Support Systems* (SAD), DAFFOR, visando atuar em blecautes nos seus sistemas, com considerações sobre execução em tempo real. O sistema possui dois *kernels*, um de raciocínio e o outro de atualização em tempo real. O *kernel* de raciocínio disparava um processo de simulação de fluxo de potência, que seria comparado com os dados oriundos do *kernel* de atualização em tempo real, o que permitia ao sistema determinar se estava diante de um evento ainda não visto anteriormente. O *kernel* de raciocínio podia, assim, adaptar sua operação à evolução das variáveis do sistema de potência, obtidas pelo *kernel* de atualização a partir dos dados do Supervisory Control And Data Acquisition (SCADA).

Bann *et alli*(BANN; IRISARRI *et al.*, 1991) descreveram em matéria para a *IEEE Expert* um ambiente para a integração de sistemas de Inteligência Artificial em um Sistema de Gerenciamento de Energia, que foi chamado Operator Decision Environment (ODE). Os autores desenvolveram um de ambiente que contemplava o modelo do sistema de potência, bem como o interfaceamento com os subsistemas de outros Sistemas de Gerenciamento de Energia Externos. O projeto do ODE foi dividido em camadas que isolavam o Banco de Dados, o acesso a este banco, as aplicações e as decisões. Entre as aplicações desenvolvidas para o ODE estavam um sistema de diagnóstico de falhas, um sistema de processamento inteligente de alarmes e um sistema de auxílio à restauração. Todas estas aplicações do sistema foram implementadas como sistemas especialistas. O artigo ainda apresenta um conjunto de estudos de casos de implementações.

3.1.3 Sistema Híbridos

Huang *et alli*(HUANG; GALIANA; VUONG, 1991) apresentaram uma metodologia que incorporava gráficos interativos, algoritmos de otimização, procedimentos de procura, regras heurísticas e simulações de fluxo de potência interativos. O processo de criação gráfica foi salientado como sendo de grande importância, por permitir a criação visual de planos de restauração. A pro-

posta procurava a definição de cenários ótimos de restauração, pela formulação matemática e por programação linear. O processo de otimização acelerava a restauração e garantia o atendimento a todas as restrições modeladas. A inclusão das regras heurísticas foi apontada como uma extensão futura, e teria o papel de auxiliar na subdivisão do sistema em ilhas e na criação automática de cenários de chaveamentos.

Cada cenário obtido no sistema definia um plano de restauração, com a seqüência de chaveamentos para sua implementação. Esta seqüência podia ser modificada e testada interativamente, com o uso de um simulador de fluxo de potência.

Nagata *et alli*(NAGATA; SASAKI; R.YOKOYAMA, 1995) apresentaram o desenvolvimento de um sistema híbrido, envolvendo metodologias de Programação Matemática (PM) e de SE, utilizando uma estratégia de decomposição-coordenação. Os autores alegam terem reduzido o tempo de computação para 30% do praticado anteriormente. O sistema desenvolvido focalizou-se na determinação de uma configuração alvo para a restauração, delegando o sequenciamento de operações para outros responsáveis.

Com a inclusão do conceito de um *custo de restauração*, o problema de restauração pôde ser expressado naturalmente como um problema de PM, e com isto, o número de regras do SE foi reduzido para um terço dos encontrados em sistemas similares, não híbridos.

3.1.4 RNA

Bretas e Phadke(BRETAS; PHADKE, 2003) propuseram a utilização de RNA na restauração, numa tentativa de redução do tempo para a obtenção de uma solução. Nesta proposta, o sistema foi decomposto em ilhas de restauração, cada qual responsável pelo desenvolvimento de um plano de restauração. A própria divisão do sistema em ilhas seria objeto de estudo separado. Os planos de cada uma das ilhas seriam compostos de uma seqüência de chaveamentos e de uma previsão de carga restaurada.

Cada ilha seria composta de duas RNA e de um programa de sequenciamento de chaveamentos. A primeira RNA forneceria uma previsão de carga, a partir de um valor normalizado de carga pré-distúrbio. A segunda RNA seria responsável pela determinação da configuração final da ilha, e teria, como entrada, um vetor composto pelo valor normalizado de previsão de carga, fornecido pela RNA anterior e por um total de três possíveis linhas de transmissão não disponíveis, por quaisquer motivos. O produto desta RNA seria a configuração final mencionada e um valor indicando o percentual de carga atendido para gerar uma configuração operacional possível. Testes realizados com um sistema composto de 162 barras foram bastante animadores.

Hsu e Huang(HSU; HUANG, 1995) desenvolveram um método baseado em reconhecimento de padrões com RNA para a restauração de sistemas de distribuição. A RNA desenvolvida recebia

em suas entradas dois tipos de informação:(a) o estado das cargas do sistema, com valores normalizados entre 0,1 e 0,9; (b) a capacidade disponível nos alimentadores do sistema. Já a camada de saída produzia o estado de ativação das diversas chaves presentes no sistema, que estabeleciam a restauração a partir dos alimentadores principais e dos laterais, ou alimentadores de suporte, característica do sistema de distribuição onde os autores desenvolveram o sistema.

O reconhecimento de padrões da proposta consistia no armazenamento de casos de restauração resolvidos pela RNA. O armazenamento era feito para o conjunto de dados composto pelo local da falha no sistema, o estado das cargas após a falha e o plano de restauração fornecido pela RNA. Quando da ocorrência de falhas, haveria a recuperação dos padrões considerados semelhantes, e a escolha do padrão aplicado seria realizada em função do número de operações de chaveamento necessárias.

3.1.5 Outras técnicas de IA

Islam e Chowdhury (ISLAM; CHOWDHURY, 2001) propuseram um sistema baseado em casos, RBC, visando o treinamento em operações de restauração. Este sistema seria capaz de resolver os problemas apresentados com base em problemas semelhantes, anteriormente introduzidos. Ele seria capaz de adaptar as soluções armazenadas para os novos problemas. Esta característica foi introduzida no sistema após o estudo de um sistema real em operação. O processo de adaptação do resultado de casos anteriores passava, ainda, por uma verificação feita em um programa de análise de fluxo de potência.

Nagata *et alli* (NAGATA; SASAKI, 2002; NAGATA *et al.*, 2002) publicaram artigo propondo o uso de multi-agentes em restauração. No artigo foram sugeridos dois tipos de agentes, os agentes de barramentos (*bus agents*) e um único agente facilitador. Os agentes de barramentos deveriam definir uma configuração alvo baseada apenas no seu próprio conhecimento e no de seus vizinhos, gerenciados pelo agente facilitador. Desta forma, cada agente de barramento possuía o conhecimento de uma conjunto de estratégias simples para restauração local, ao passo que o agente facilitador conduzia uma procura global. Os autores concluíram que a arquitetura seria promissora para sistemas de grande escala e mais complexos.

3.1.6 Algoritmos Genéticos

Outra linha de tratamento para o problema de RSP tem empregado Algoritmo Genético (AG)s. Seguem os resumos de alguns trabalhos na área.

Fukuyama *et alli* (FUKUYAMA; ENDO; NAKANISHI, 1996) desenvolveram um sistema híbrido, empregando SE e AG. O trabalho objetivava a restauração do abastecimento em redes de distribuição. O SE as operações de chaveamento necessárias para aumentar a disponibilidade de

energia na área afetada, caso não houvesse capacidade no local. O papel do AG era o de determinar operações de chaveamento para decompor as áreas afetadas pelo desabastecimento em áreas menores, alocadas em cada fonte de potência disponível.

Os autores continuam o artigo descrevendo o fluxo de operação do sistema desenvolvido, mostrando a operação do SE para disponibilizar potência extra para atender a área afetada, e descrevendo a representação da *string* utilizada no AG, as mudanças necessárias nos operadores genéticos para o problema de restauração e a função de *fitness* empregada.

Luan *et alli* (LUAN; IRVING; DANIEL, 2002) apresentaram uma aplicação de AG para a procura de uma estratégia de restauração de sistemas de distribuição ótima. A representação da *string* empregava *permutação inteira*, indicando a ordem das operações de chaveamento, e a natureza da operação (abrir ou fechar) era determinada pela teoria de grafos, sujeita às restrições de radialidade dos sistemas de distribuição.

A desconexão de cargas era possível pelo emprego de um gene especial, 0, introduzido na *string* para indicar que uma dada parte da rede deveria ficar desconectada.

Bretas *et alli* (BRETAS; DELBEN; CARVALHO, 1998) desenvolveram um método de restauração de sistemas de distribuição baseado em AGs que emprega uma nova forma de representação na *string*, utilizando cadeias de grafos. Nesta representação, cada entidade era representada por um pacote de *strings* (cromossomos), cujos *bits* (genes) alocavam os valores (alelos). Cada entidade representava um ramo radial do sistema a restaurar, capaz de alimentar todos os seus setores, ou seja, uma solução completa para o sistema com falha.

A definição das cadeias de grafos utilizadas nas *strings* partia da comparação entre Grafos desconexos e sem ciclos. A representação destes grafos por cadeias de grafos reduziu em muito a representação do sistema, e paralelamente o tempo de processamento. A utilização dos grafos e a observância de restrições nas operações do AG garantiram que somente soluções possíveis fossem produzidas.

Em seguida, cita-se o trabalho de Santos *et alli* (SANTOS *et al.*, 2004), que sugeriram a possibilidade de implementação de um AG para o problema de RSP baseado nas soluções de expansão de sistemas, o que indicou que a metodologia empregada pelos autores poderia ser um bom indício de caminho a seguir, no emprego de AGs. O artigo indica o uso de um AG para a produção de estratégias *off-line* que poderiam gerar instruções e procedimentos para RSP.

Os autores ainda exploram o uso de processamento distribuído para a solução de RSP, baseados na divisão do sistema em ilhas, em especial nos estágios iniciais de uma restauração massiva. Foram realizados experimentos que indicaram o proveito desta característica dos sistemas de potência, em especial durante a recomposição. Os resultados obtidos mostraram um ganho de velocidade (*speedup*) sensível a partir de 3 máquinas no sistema.

3.1.7 Conclusões da Revisão

Após a análise do conjunto de referências apresentado, salientam-se alguns pontos que levaram a dissertação ora apresentada, e que deram origem a algumas de suas linhas de desenvolvimento.

Há, em primeiro lugar, uma grande ocorrência de trabalhos que produzem planos de restauração *offline*. A frequência deste tipo de tratamento indica que a procura de soluções ótimas ainda requer quantidades de tempo incompatíveis com o tempo de resposta esperado destes sistemas. Os trabalhos apontados por (MATSUMOTO *et al.*, 1992) como utilizados para a produção de planos e procedimentos para restauração, por (HUANG; AUDETTE; HARRISON, 1995), com a preparação de bases *offline* e por (BRETAS; PHADKE, 2003), também com a criação de seqüências de operações de chaveamento *offline*, entre outros autores, levaram a inclusão deste conceito na proposta. A divisão do Sistema de Potência em subsistemas, ou ilhas de restauração, também é prática comum entre as pesquisas revisadas. (HUANG; GALIANA; VUONG, 1991), (HUANG; AUDETTE; HARRISON, 1995), (BRETAS; PHADKE, 2003) praticaram ou indicaram esta divisão, que também foi considerada para a dissertação.

Em síntese, estas são as idéias que formaram a base do que se está apresentando nesta dissertação.

4 METODOLOGIA

Este capítulo apresenta a metodologia empregada no desenvolvimento do Algoritmo Genético (AG), bem como de sua integração com o Programa de Fluxo de Potência (PFP). Alguns dos sistemas de potência aos quais o AG será aplicado são também apresentados.

4.1 Descrição Geral

O sistema desenvolvido é constituído por um conjunto de dados, ferramentas e atividades, interligados de forma a compor um conjunto de informações e de processamento destas. Estes elementos e suas conexões estão esquematizadas num diagrama, apresentado na figura 11.

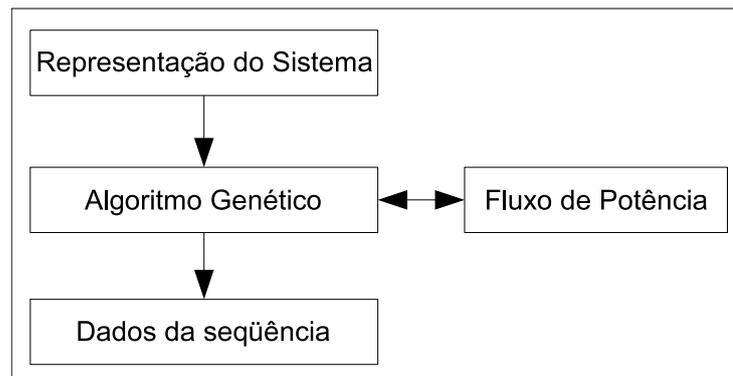


Figura 11: Componentes do programa desenvolvido para RSP.

Na figura 11, apresentam-se quatro componentes principais. Tem-se, inicialmente, a representação dos sistemas de potência. Esta representação é criada a partir da leitura e interpretação de arquivos que descrevem estes sistemas. No capítulo 5.2 serão apresentados os formatos de representação de sistemas de potência que foram utilizados no trabalho.

Os dados desta representação serão acessados por um *Algoritmo Genético*(AG), o segundo bloco da figura 11. Este AG possui uma população de cromossomos que representam possíveis soluções para o problema. Cada solução codifica uma seqüência de operações a serem realizadas sobre os elementos do Sistema de Potência. Durante a evolução deste AG, cada etapa da seqüência codificada altera o sistema de potência, que é apresentado ao PFP, para que este forneça uma avaliação do seu estado após cada novo estágio. Este passo indica ao AG se a

etapa foi bem sucedida e, caso positivo, quais os valores das variáveis de estado obtidas após sua conclusão. Os valores destas variáveis de estado são consultados pela Função Objetivo do algoritmo (4.2.7). A partir deste ponto, segue-se o fluxo normal do AG, com a seleção de pares e os seus cruzamentos, obtendo-se uma nova população a cada iteração do algoritmo.

O bloco seguinte explicita o PFP, que fornece a avaliação necessária ao funcionamento do AG, quando sua Função Objetivo é executada. À medida em que o AG altera a seqüência de operações codificada em cada cromossomo, criando uma sucessão de estados distintos para o sistema de potência, estes estados são apresentados ao PFP, que os avalia através da solução do fluxo de potência, consecutivamente, para cada estado, acumulando as informações referentes às variáveis monitoradas pela função objetivo.

Os cromossomos da população representam soluções para a restauração do sistema. Estas soluções são expressas por meio de seqüências de operações. O quarto bloco da figura 11 representa os dados destas seqüências de operações. Elas existem, internamente ao AG, codificadas diretamente nos cromossomos, e também em formato textual, em um arquivo externo, criado após o encerramento do AG. Este arquivo contém apenas a descrição da seqüência de operações do cromossomo escolhido como o mais apto pelo AG, ou seja, a melhor solução encontrada.

4.2 Algoritmos Genéticos

O principal componente do programa implementado é um Algoritmo Genético. Este tem por função gerar seqüências de operações elementares, ou blocos de operações, que devem levar o sistema de potência de um estado inicial de blecaute total até um estado onde houvessem mais consumidores atendidos.

É necessário mencionar, neste ponto, que optou-se pela solução apenas de casos de blecautes totais. Esta foi uma restrição introduzida na proposta do trabalho por se entender que estes seriam os casos mais simples para se comprovar a eficácia do método. A implementação dos casos genéricos, de distúrbios e interrupções localizadas, ou mesmo generalizadas, não implicando em blecautes totais, foi preterida para trabalhos posteriores.

4.2.1 Representação do Genoma

Para o AG encontrar a seqüência mais adequada dentro de seu espaço de busca, deve-se encontrar uma representação de cromossomo adequada para o problema. Durante a revisão da bibliografia da área, o emprego de AGs para a determinação de seqüências de operações só foi verificado em sistemas de distribuição, como em (FUKUYAMA; ENDO; NAKANISHI, 1996), (LUAN; IRVING; DANIEL, 2002) e (BRETAS; DELBEN; CARVALHO, 1998). Os sistemas de distribuição tem topologia predominantemente radial, o que levou estes pesquisadores ao emprego de repre-

representações que explorassem as características desta topologia. Como a topologia de sistemas de distribuição e de transmissão são sensivelmente diferentes, o emprego de AGs para transmissão requer a formulação de novas representações de *cromossomos*.

Inicialmente, um conjunto de representações foi idealizado para o trabalho. Uma primeira representação foi criada com a divisão da cadeia em grupos de cromossomos, cada um indicando um determinado estágio de restauração (figura 12). Dentro de cada estágio, teríamos a inclusão de barras especialmente selecionadas para este estágio. Esta seleção seria feita por algum processo que determinasse quais as prioridades de atendimento, como, por exemplo, foi proposto em (HUANG; AUDETTE; HARRISON, 1995) e (KOJIMA *et al.*, 1989b), ou para a geração de estratégias *build-down*, com a re-energização de uma rede de distribuição principal (ADIBI; FINK, 1994).



Figura 12: Representação com cromossomo dividida em grupos pré-selecionados, indicando estágios

Outra representação possível, seguindo o mesmo raciocínio, poderia não realizar uma pré-seleção das barras. Desta forma, o próprio AG iria determinar em qual estágio cada barra teria seu estado alterado. Na figura 13 pode-se ver uma representação nestes moldes. O cromossomo continuaria a ser dividido no mesmo número de estágios, agora com iguais dimensões dentro da cadeia. O AG, durante sua evolução, iria indicar quais barras seriam ativadas em cada estágio. Mas o número de estágio ainda deveria ser pré determinado.

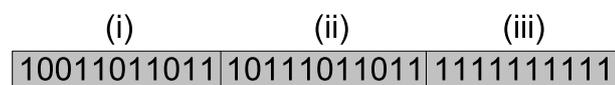


Figura 13: Representação com cromossomo dividido em grupos, indicando estágios

Para se ter uma representação com um número indeterminado de estados, poderia ser utilizada uma representação semelhante à presente na figura 14. Neste caso, o cromossomo não possuiria divisões, e o estágio de ativação de cada barra seria codificado diretamente no vetor de barras. Seria possível a codificação direta do número do estágio de ativação de cada no alelo presente nos *loci* (posições - 2.3.1.1) do cromossomo, como 1 para estágio 1, 4 para 4, etc. Entretanto, esta representação seria muito sensível aos operadores genéticos, porque a mudança de um único bit no número codificado alteraria profundamente o estágio equivalente. Outra possibilidade seria, como indicado na própria figura 14, uma representação binária, obtendo-se o respectivo estágio da quantidade de bits ativos no número presente em cada posição do vetor de barras.

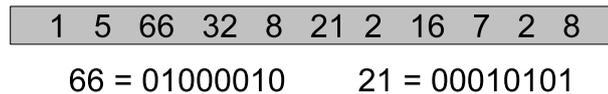


Figura 14: Cromossomo com a indicação direta do estágio de ativação das barras

Assim, neste exemplo (figura 14), o terceiro *gene*, de valor decimal 66, por possuir apenas 2 *bits* iguais a 1, indicaria o estágio 2. O sexto *gene*, igual a 21, indicaria o estágio 3.

Durante a implementação, uma outra representação foi criada. Esta representação é diferente de todas as demais encontradas durante o estudo do estado da arte, representando uma grande contribuição deste trabalho. De fato, esta nova representação propicia um desenvolvimento bastante satisfatório da etapa genética, em especial para o tratamento de problemas de sequenciamento.

Nesta nova proposta, cada cromossomo codifica uma seqüência *variável* de estágios. Cada estágio tem, por sua vez, um tamanho também variável e indica um conjunto, ou bloco, de operações que são realizadas durante sua extensão. Estas operações podem ter sua execução simulada sobre barras e sobre as linhas de transmissão do sistema de potência.

Um operador de cruzamento foi especialmente desenvolvido para esta nova representação, a fim de evitar que os novos indivíduos, gerados nesta operação, possuíssem estágios indevidos após o cruzamento (4.2.5). Da mesma forma, um novo operador de mutação também foi codificado, capaz de realizar mutações adequadas a esta representação (4.2.6).

A figura 15 apresenta dois indivíduos hipotéticos, utilizando uma representação meramente ilustrativa para sua compreensão inicial.

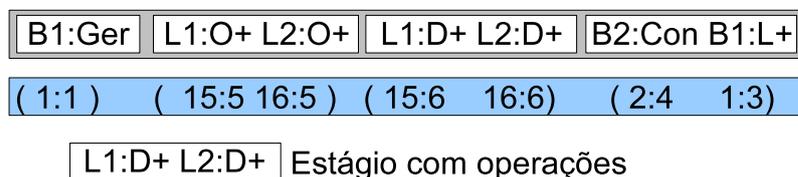


Figura 15: Representação com cromossomos divididos em estágios, com conjuntos de operações discretas. Na figura, vê-se a mesma representação em forma mnemônica e na forma numérica.

Na figura 15, pode-se ver os estágios como elementos discretos dentro de cada string (nos blocos internos da figura), e cada estágio contendo um grupo de operações. Cada operação é indicada com um par de identificadores, no formato *endereço:operação*.

Os endereços indicam onde as operações serão realizadas. Se a operação for executada sobre a barra n , seu endereço será expresso como Bn . Se for executada sobre a linha de transmissão n , o endereço será Ln .

As operações de barra incluem a ativação da carga conectada à barra (operação $L+$), a ativação do gerador associado à barra (operação Ger), ou a ativação do capacitor reator associado à barra (operação $C+$). As operações sobre as linhas restringem-se a ativação de suas extremidades de origem e destino (operações $O+$ e $D+$).

As diversas operações alteram o estado de barras e linhas. O estado inicial das barras é o de ausência de energia (um caso de blecaute total). As barras necessitam de alguma forma de energização. Esta energia pode ser fornecida por meio de geradores próprios, com capacidade de autogeração (*blackstart*), ou por meio de linhas de transmissão já energizadas. Quando uma barra é energizada, subentende-se que esta está *ativada*. A partir do estado ativado, as barras podem alimentar suas cargas, ativar geradores adicionais e reatores, ou energizar outras linhas de transmissão.

Ou seja, a partir da observação do estado atual de uma barra, algumas das operações sobre ela são habilitadas ou desabilitadas.

4.2.2 Inicialização dos Genomas

Esta representação demandou a inclusão de mais componentes no AG, a fim de se obter uma inicialização adequada. A figura 16 ilustra os componentes do AG utilizados na inicialização dos cromossomos iniciais da população.

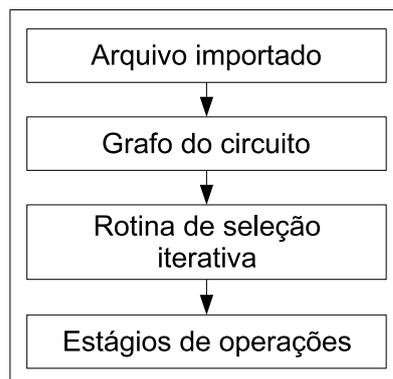


Figura 16: Componentes do programa envolvidos na inicialização do AG

A partir da leitura dos dados do Sistema de Potência do arquivo descritor, representado pelo primeiro bloco da figura 16, cria-se um grafo para representá-lo, onde, a partir das barras que possuem capacidade de geração própria (unidades *black start*), se incluem as barras diretamente conectadas a esta e assim, sucessivamente, conectando-se cada barra ainda não conectada nesta árvore. Como resultado, se obtém um grafo representando a conectividade do sistema.

Cada nodo do grafo representa uma das barras do sistema de potência, e os arcos representam linhas de transmissão entre estas barras. Os nodos são pesquisados em largura Breath First Search (BFS), de acordo com a topologia do grafo. Os componentes do sistema de potência

conectados às barras referenciadas pelos nodos são selecionados aleatoriamente. Cada um destes componentes selecionados gera uma operação, determinada pelo tipo de componente e pelo seu estado atual. A cada iteração da rotina, um novo conjunto de operações é gerado, até que todo o grafo tenha sido pesquisado e o número de nodos ainda desconectados seja inferior a um limite sorteado a cada novo cromossomo. Cada nodo pode gerar mais do que uma operação, em estágios diferentes, fazendo com que a barra referenciada passe por estados progressivos. Desta forma, são geradas diferentes seqüências de operações na população inicial, com diferentes números de nodos remanescentes.

Uma demonstração do processo de inicialização dos genomas pode ser visualizada na figura 17, utilizando-se, como exemplo, o sistema IEEE-14 barras. A figura ainda apresenta uma seqüência de grafos indicando o estado de alguns dos componentes deste sistema.

A barra 1, destacada no sistema e no estágio (1) da figura, foi arbitrada como possuindo capacidade de inicialização própria (*black start capable*). Os seis grafos seguintes representam estágios do processo de inicialização do genoma. No primeiro, apenas o nodo equivalente à barra 1 do IEEE 14 barras é ativada, sendo alimentada pelo seu próprio gerador.

A ativação desta primeira barra permite a energização das linhas a ela conectadas. No estágio (2), as duas linhas de transmissão conectadas à barra 1 são ativadas. Esta conexão foi representada com os pontos escuros nas extremidades das linhas.

No estágio seguinte (3), dentre os nodos agora passíveis de ativação, o algoritmo sorteia aleatoriamente, se deve ou não ativar as barras de potência referenciadas por cada um deles. No caso representado, o algoritmo determinou a ativação da barra 2 e a não ativação da barra 5. Este é um processo aleatório, controlado por parâmetros. A ativação da barra 2 é efetivada após a alteração do estado da linha de transmissão entre ela e a barra 1, representada pelo ponto escuro sobre a extremidade contígua a esta barra. Analogamente ao sistema real, a energização da barra 2 só foi possível pela transmissão da energia através da linha mencionada.

As linhas de transmissão ligadas a uma barra ativada também passam por um sorteio aleatório para determinar sua ativação. Isto pode ser visualizado no estágio (4), onde, das três linhas conectadas a barra 2 ainda não ativadas, a linha entre esta barra e a barra 5 permaneceu sem ativação.

No estágio (5), o algoritmo continua visitando todos os nodos diretamente conectados a nodos já ativados por meio de linhas de transmissão já energizadas, e determina se estes devem ser ativados, pelo sorteio aleatório. No exemplo, as duas linhas, ativadas no estágio anterior, energizam as duas barras a elas conectadas, as barras 3 e 4. Além disto, ainda se energiza a linha de transmissão entre as barras 2 e 5, que ficara desativada no estágio (4).

No último estágio representado no exemplo (6), o algoritmo demonstra a ativação da barra

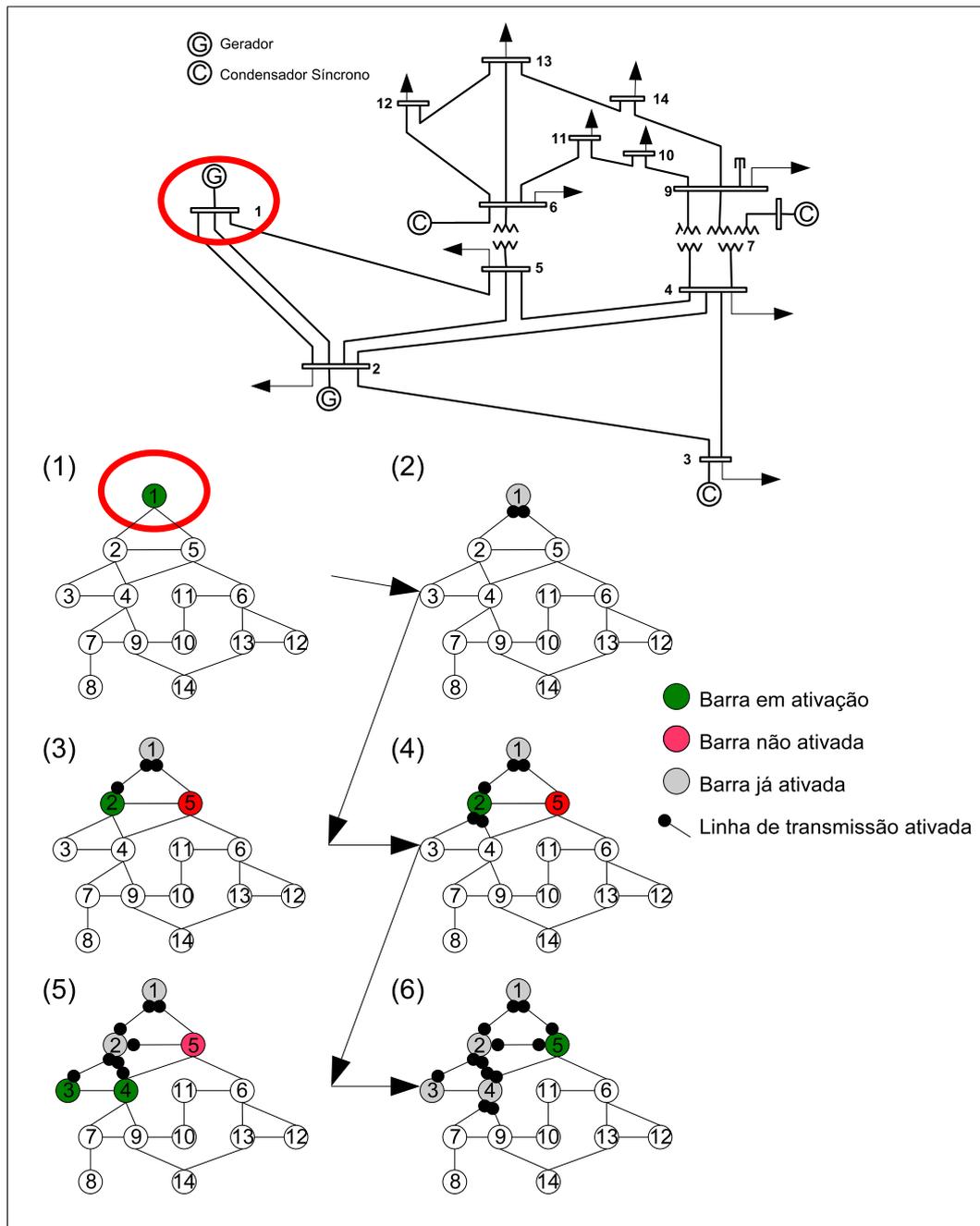


Figura 17: Demonstração do processo de inicialização dos genomas. O grafo é pesquisado e as barras correspondentes são ativadas aleatoriamente. Barras não ativadas em um determinado estágio têm nova oportunidade de serem ativadas, pois o processo é iterativo.

5, que havia permanecido sem ativação desde o estágio (3). As duas linhas de transmissão que chegam a esta barra, oriundas de barras já ativadas, tem sua ativação ilustrada, indicando que a energia possa ser transmitida de forma distribuída, por elas. O reprocessamento de nodos do grafo que já foram visitados é determinado aleatoriamente. Neste caso, a busca BFS reinicia e o algoritmo avalia o estado de cada barra, como se o estivesse fazendo pela primeira vez, mas

agora alguma barras e linhas já se encontram ativadas. ¹

Cada operação determinada durante estas etapas é codificada no cromossomo, de acordo com cada estágio do algoritmo. A inicialização continua até que o número de nodos inativos remanescentes seja inferior a um limite aleatoriamente escolhido. Esta escolha aleatória gera indivíduos com diferentes níveis de atendimento das cargas do sistema.

Neste exemplo não foram demonstradas as ativações de cargas nas barras que as possuem. Sua ativação, ou seja, o atendimento a estas cargas, seria determinado pelo mesmo sorteio, a partir do estágio subsequente a ativação da própria barra.

A aleatoriedade na ativação das barras, linhas e demais elementos permite a obtenção de indivíduos com diversidade de soluções. Como foi ilustrado no estágio (6) da figura (17), esta aleatoriedade permite que, em cada nível da busca BFS, algumas barras não sejam ativadas. O algoritmo reprocessa o grafo, a procura destas barras remanescentes, sendo que isto ocorre em duas circunstâncias: após terminada uma busca em largura, e por interrupções ocasionadas durante esta procura. Estas interrupções são sorteadas aleatoriamente, fazendo com que a rotina ative um número indeterminado de barras e reinicie para ativar as remanescentes. A rotina é definitivamente interrompida quando o número de barras não conectadas remanescentes for inferior a um limite. Este limite garante que o algoritmo seja interrompido e é escolhido aleatoriamente a cada novo indivíduo inicializado.

4.2.3 Detalhamento da Representação do Genoma

Na nova representação desenvolvida, o genoma, composto por um único cromossomo, passou a codificar as operações e os estágios em que estas deveriam acontecer em uma lista. As operações foram divididas em dois grupos, operando sobre linhas e barras. Utilizou-se uma propriedade de estado em cada um destes componentes, barras e linhas. A utilização destes estados mostrou-se como uma analogia que auxiliava a garantir que cada elemento realmente passasse por uma seqüência progressiva de configurações, até atingir um estado *aceitável* ².

Para as linhas, os estados possíveis incluíam:

- não conectada;
- origem conectada;
- destino conectado;

¹Esta característica é um indício de que o algoritmo deva funcionar nos casos onde não haja blecaute total, apesar de não ter sido esta uma definição do escopo do trabalho.

²*Aceitável* no sentido de que, para as barras, a simples conexão desta já poderia ser um estado terminal. O fato de sua carga estar sendo atendida ou não seria função de desempenho de cada genoma, e não uma restrição do processo. Uma barra desconectada, por outro lado, poderia não ser desejável, pois poderia inviabilizar caminhos para a transmissão de energia.

- conexão total.

A razão de se modelar diferentemente a conexão em cada extremidade de uma linha está associada ao fato de que o algoritmo poderia criar conexões em ordens tais que uma barra poderia ser ativada antes de outra, e vice-versa, em indivíduos diferentes da população. Este efeito pode ser visualizado na figura 18, onde temos barras diferentes ativadas, em a) e b). Em a), com as barras A e B ativadas, a primeira extremidade da linha BC a ser conectada é sua origem, coincidente com B. Em b), como as barras inicialmente ativadas são C e D, a extremidade de BC a ser ativada primeiramente é o seu destino. Origem e destino, neste caso, dizem respeito somente a ordem em que as barras são citadas no arquivo do qual o sistema foi importado, nada relacionado ao sentido de ativação.

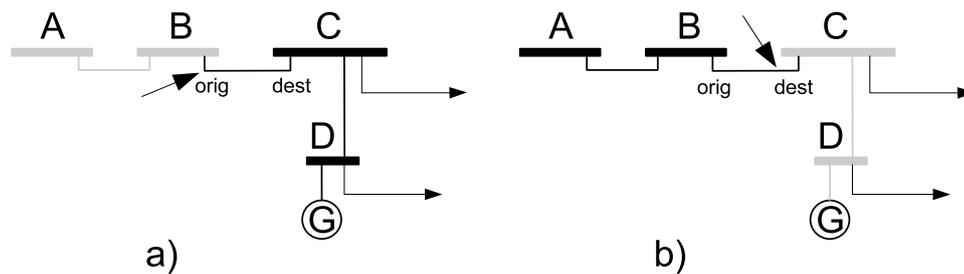


Figura 18: Exemplo da mudança do sentido de ativação de linhas de transmissão, em situações diferentes. A presença de diferente barras ativadas determina que uma linha tenha a ativação inicial realizada ora em uma de suas extremidades, ora em outra.

Para as barras de potência, os estados contemplados na modelagem foram:

- barra desconectada;
- barra conectada - quando a barra possui potência oriunda de alguma de suas linhas de transmissão ou a um gerador;
- gerador conectado - que pode ser distinto da conexão da própria barra;
- carga conectada;
- reator conectado.

Os três últimos estados somente são importantes para as barras que possuírem os respectivos componentes envolvidos: geradores, cargas ou reatores, e somente serão atingidos por uma barra a partir do estágio seguinte ao qual ela foi acionada.

4.2.4 Inter-operação com o Fluxo de Potência

Cada genoma é avaliado pelo AG que, no nosso caso, necessita simular cada estágio codificado. Esta simulação é realizada pelo PFP, através das funções de acesso e controle desenvolvidas e descritas no capítulo de implementação (seção 5.3).

Durante esta simulação, cada estágio do genoma é simulado, com a ativação seletiva de nodos e linhas, conforme determinado pelo AG. Este processo envolve:

- Inicializar a interface: como o PFP tem alocação estática, é necessário limpar-se os seus vetores internos.
- Configurar barras ativas: todas as barras que tiverem estado igual ou superior ao de *barra ativada* são incluídas e configuradas no PFP.
- Configurar linhas ativas: todas as linhas que foram necessárias e ativadas, para permitir que o conjunto de barras já configurado fosse ativado, são configuradas também.
- Executar o algoritmo de fluxo de potência e verificar-se o retorno da função. Caso esta não tenha encontrado uma solução convergente, isto é indicado ao AG.

Utilizando o exemplo da figura 17, a seqüência de 6 estágios demonstrada iria gerar 6 execuções do PFP. Antes de cada execução, o PFP seria configurado com as barras presentes na simulação de cada estágio.

Por exemplo, o estágio (4) da figura 17 iria configurar o PFP com duas barras (1 e 2) e com uma linha, presente entre estas barras. A barra 5, assim como as linhas presentes entre ela e as demais, não seriam incluídas neste estágio, por não terem sido determinadas como ativas.

Já no estágio (6) da mesma figura, o conjunto seria composto pelas barras 1, 2, 3, 4 e 5. Da mesma forma, apenas as linhas presentes entre estas barras e com ambas as extremidades ativadas seriam configuradas no PFP.

4.2.5 Operador de Cruzamento

Foi necessária a implementação de um novo operador de cruzamento para o genoma codificado, tendo em vista a maior complexidade da representação utilizada. Este operador deveria evitar problemas como seqüências com operações indevidas. Exemplos de operações indevidas são a tentativa de ativação da carga ou de um gerador de uma barra quando esta ainda não possuísse potência disponível.

Para isto, foi implementado um processo de procura entre os estágios de cada indivíduo selecionado para o cruzamento. Os estados das barras presentes nos dois cromossomos são com-

parados, e quando houver equivalência entre estes, são armazenados em uma lista. A equivalência foi determinada a partir da conexão das barras. Ou seja, se as barras comparadas estiverem, no mínimo, conectadas nos dois cromossomos, são consideradas equivalentes. O processo pode ser visualizado na figura 19.

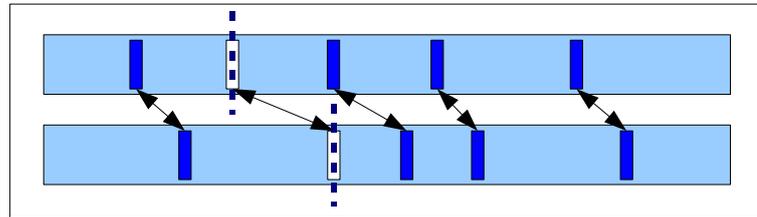


Figura 19: Localização de estados equivalentes entre os estágios de dois indivíduos da população. Podem ser detectados diversos pontos cujos estados das barras são equivalentes. Os pares formados pelos dois estágios são armazenados e um deles é escolhido aleatoriamente. As linhas tracejadas indicam os pontos onde as duas seqüências serão divididas e cruzadas entre si.

Uma consequência intuitiva deste operador de cruzamento é de que as seqüências resultantes terão número de operações diferentes. Algumas tenderão a serem menores e, possivelmente, mais eficientes que as demais.

Os índices dos estados equivalentes são armazenadas em listas. Ao final da comparação, uma posição da lista é sorteada aleatoriamente e determina os pontos onde as duas seqüências serão interrompidas e onde ocorrerá o cruzamento.

4.2.6 Operador de Mutação

O operador de mutação implementado sorteia, aleatoriamente, uma posição da seqüência de operações. No estágio descrito por esta posição, sorteia novamente uma barra do sistema e verifica se esta já está conectada. Caso esta já tenha sido alimentada, ele ativa a carga ou o gerador presente na barra, se este ainda não o tiver sido feito em estágio anterior. Caso a operação já tenha sido efetuada anteriormente, o operador sorteia se deve eliminar a operação anterior ou se deve movê-la para outro estágio, submetido à restrição de que, neste estágio, a mesma barra já tenha sido ativada. Se neste estágio a barra selecionada ainda não estiver sendo alimentada, haverá novo sorteio, até que se identifique uma barra na qual se possa operar.

Uma demonstração das operações implementadas no operador de mutação pode ser visualizada na figura 20. Nesta figura, estão representados os conceitos de inclusão e exclusão de operações e de alteração do estágio em que uma operação deva ocorrer.

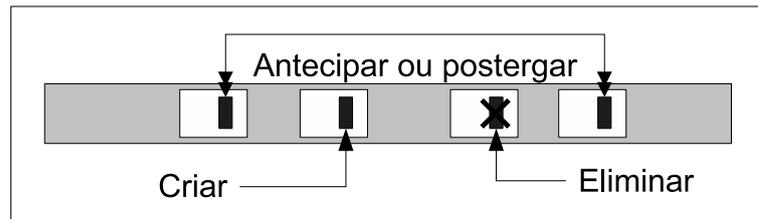


Figura 20: Representação das possibilidades de mutações. As áreas claras representam os estágios, e as escuras, as operações. As operações podem ser excluídas, incluídas ou ter seu estágio alterado, resguardadas as restrições do sistema.

4.2.7 Função Objetivo

A função de *fitness*, ou função objetivo, foi planejada de forma a considerar diversas variáveis do sistema, a cada estágio determinado pelo AG. Isto implicou em uma nova execução do Fluxo de Potência (4.3) a cada estágio sugerido, após as suas operações serem adequadamente transferidas ao modelo apresentado para o programa de Fluxo de Potência (4.2.4).

Mais detalhadamente, durante a avaliação de cada cromossomo pelo AG, o programa itera entre todas as barras ativas em cada estágio, transferindo os dados destas para o Fluxo de Potência. As linhas de transmissão necessárias para a ativação das barras já transferidas também são configuradas no Fluxo de Potência. Somente as barras e linhas necessárias para a execução do estágio em questão são transferidas para o programa de Fluxo de Potência. Isto foi garantido durante a determinação de cada estágio, pela procura seletiva das barras com possibilidade de ativação dentro do grafo do sistema, ainda no processo de inicialização do genoma (seção 4.2.2)

Procurou-se permitir a existência e a seleção de funções objetivo diferentes e parametrizáveis, para permitir uma análise mais detalhada da influência de cada variável do sistema nos resultados do AG. Desta forma, o programa permite a seleção e o controle das funções nele codificadas. Os detalhes da implementação das funções objetivo estão descritos no ítem 4.2.7.1.

4.2.7.1 Funções Implementadas

Após a execução da seqüência de estágios codificada no indivíduo, a função objetivo é pesquisada. Estão implementadas cinco funções objetivo, descritas a seguir. Note-se que estas funções foram implementadas progressivamente e representam evoluções na solução do problema.

4.2.7.1.1 Função objetivo 1

A primeira função objetivo implementada pode ser visualizada na equação (4.1). Sua intenção é o atendimento do máximo de potência com o mínimo número de estágios. Assim, à medida que mais cargas ativas e reativas forem atendidas nos primeiros estágios da solução codi-

ficada no cromossomo, maior o valor obtido com esta função. Pelo contrário, as cargas atendidas em estágios posteriores são desvalorizadas.

$$F_{O_1} = \sum_{e=1}^N \frac{P_{AT_e} A + P_{RE_e} B}{e} \quad (4.1)$$

onde N é o número de etapas da solução, P_{AT_e} e P_{RE_e} são as potências ativa e reativas totais atendidas na etapa e e A e B são constantes de ajuste.

4.2.7.1.2 Função Objetivo 2

A função objetivo seguinte, equação (4.2), tenta reduzir o número de estágios de operações

$$F_{O_2} = \sum_{b=1}^M \frac{(P_{AT_b} + P_{RE_b}) A}{1 + B * N} \quad (4.2)$$

onde M é o número de barras do sistema de potência, P_{AT_b} e P_{RE_b} são as potências ativa e reativa atendidas pela barra b , no último estágio, e A e B são constantes de ajuste.

Pelo controle das constantes A e B , é possível modificar-se o comportamento do AG, aumentando-se ou diminuindo-se a influência do número de estágios da solução.

4.2.7.1.3 Função Objetivo 3

$$F_{O_3} = \frac{\sum_{b=1}^M A(P_{AT_b} + P_{RE_b}) - B(\Delta P)}{1 + C * N} \quad (4.3)$$

$$\Delta P = |P_{GE_b} - P_{AT_b}|$$

onde M é o número de barras do sistema, P_{AT_b} e P_{RE_b} são as potências atendidas pela barra b , P_{GE_b} é a potência gerada na barra b e A , B e C são constantes de ajuste.

Na terceira função objetivo, equação (4.3), procurou-se incluir as faltas de potências, devidas ao não acionamento de geradores nos estágios mais adequados, ou as sobras, devidas ao acionamento em estágios inadequados. O número de estágios da solução permaneceu sendo avaliado pela função.

4.2.7.1.4 Função Objetivo 4

Outra função objetivo implementada encontra-se na equação (4.5). Esta é um melhoramento da Função Objetivo 3, permitindo um controle individual sobre os aspectos de faltas e excessos

de potência geradas, através dos parâmetros A e B .

$$\begin{aligned}
 FO_4 &= \frac{\sum_{b=1}^M P_{AT_b} + P_{RE_b} - A.P_{Ex_b} - B.P_{Fl_b}}{1 + C * N} \\
 P_{Ex_b} &= \begin{cases} 0, & \text{se } P_{Ge_b} \leq P_{AT_b} \\ P_{Ge_b} - P_{AT_b}, & \text{caso contrario} \end{cases} \\
 P_{Fl_b} &= \begin{cases} 0, & \text{se } P_{AT_b} \leq P_{Ge_b} \\ P_{AT_b} - P_{Ge_b}, & \text{caso contrario} \end{cases}
 \end{aligned} \tag{4.4}$$

onde M é o número de barras do sistema, P_{AT} e P_{RE} são as potências atendidas nas barras, G_{AT} e G_{RE} são as potências geradas nas barras e A , B e C são constantes de ajuste.

A função continua avaliando, também, o número de estágios de operações da solução codificada no indivíduo.

4.2.7.1.5 Função Objetivo 5

Uma última função objetivo foi implementada, descrita pela equação 4.6. Procurou-se mais uma vez valorizar o pronto atendimento às cargas, mantendo-se as restrições de geração. Nesta função objetivo, a potência média atendida no estágio é utilizada para valorizar a resposta.

$$\begin{aligned}
 FO_5 &= \frac{\overline{L_{At}} \times (L_{At_N} - A * \overline{Fl_{At}}) + \overline{L_{Re}} \times (L_{Re_N} - B * \overline{Fl_{Re}})}{1 + C * N} \\
 \overline{L_{At}} &= \frac{\sum_{n=1}^N \sum_{m=1}^M L_{At_{nm}}}{N} \\
 \overline{L_{Re}} &= \frac{\sum_{n=1}^N \sum_{m=1}^M L_{Re_{nm}}}{N} \\
 \overline{Fl_{At}} &= \frac{\sum_{n=1}^N Fl_{At_n}}{N} \\
 \overline{Fl_{Re}} &= \frac{\sum_{n=1}^N Fl_{Re_n}}{N} \\
 Fl_{At_n} &= \begin{cases} 0, & \text{se } \sum_{m=1}^M Ge_{At} \geq \sum_{m=1}^M L_{At} \\ \sum_{m=1}^M L_{At} - \sum_{m=1}^M Ge_{At}, & \text{caso contrario} \end{cases} \\
 Fl_{Re_n} &= \begin{cases} 0, & \text{se } \sum_{m=1}^M Ge_{Re} \geq \sum_{m=1}^M L_{Re} \\ \sum_{m=1}^M L_{Re} - \sum_{m=1}^M Ge_{Re}, & \text{caso contrario} \end{cases}
 \end{aligned} \tag{4.5}$$

e onde M é o número de barras do sistema, N é o número total de estágios da solução, L_{At} e L_{Re} são as cargas ativas e reativas atendidas em cada estágio n e barra m , Ge_{AT} e Ge_{RE} são as potências ativa e reativa geradas na barra m , no estágio n , e A , B e C são constantes de ajuste.

A inclusão da potência atendida média tenta fazer com que, a cada estágio, o sistema atenda o máximo de cargas ativas e reativas possíveis, mantendo o valor destas médias mais elevado. A presença das faltas de geração na função tenta manter a geração em níveis suficientes para atender as cargas demandadas.

4.3 Fluxo de Potência

O programa utiliza um componente que executa o cálculo do Fluxo de Potência do sistema carregado. Uma descrição da metodologia envolvida neste cálculo foi apresentada na seção 2.2. O componente utilizado é um programa desenvolvido por Canto dos Santos, na linguagem de programação *Fortran*, e teve seu uso gentilmente permitido pelo autor.

Em sua forma original, o programa é executado a partir de um *prompt do sistema operacional*, não necessitando de nenhuma parametrização, pois calcula automaticamente o fluxo de potência de sistemas descritos em arquivos. Os dados obtidos pelo algoritmo são armazenados em arquivos de saída, e as mensagens enviadas durante a execução são apresentadas na tela do computador além de também serem armazenadas em outro arquivo de registros específico.

Optou-se por não utilizar o programa com esta interface por questões de performance. Como o AG iria acessar as funções deste programa com grande frequência, o *overhead* seria grande. Assim, o programa foi alterado e transformado em uma biblioteca estática, de forma a ser possível ligá-lo a outro programa, em ambiente *windows* e escrito em *C++*. Para isto, algumas alterações foram necessárias, pois as informações que, originalmente, eram lidas de um arquivo precisavam ser transferidas através de chamadas de funções entre a parte escrita em *C++* e o programa, agora biblioteca, em *Fortran*.

Estas funções permitem a definição de todos os parâmetros dos elementos do sistema. Foram criadas funções para a definição e alteração de barras e linhas de transmissão, além de funções para reinicialização das estruturas internas e controle de execução.

Os resultados obtidos pela biblioteca *Fortran* podem ser acessados diretamente da parte escrita em *C++*, pois as estruturas existentes na biblioteca são todas estáticas e globais.

Este programa tem um papel determinante no funcionamento do sistema. Ele fornece a avaliação do comportamento do Sistema de Potência, considerando um conjunto específico de dados que lhe são passados. Conforme descrito em 4.2.7, a cada estágio pertencente a um certo

indivíduo da população do AG, o programa de Fluxo de Potência é configurado com as barras e linhas de transmissão que, para este determinado estágio, foram selecionados e tiveram seu estado alterado.

4.3.1 Compensação por não convergências

As seqüências codificadas nos genomas, devido aos operadores de cruzamento e mutação, podem apresentar não convergências quando de sua aplicação ao PFP. Estes operadores podem introduzir seqüências de operações que são consideradas normais pela codificação destes operadores, que só se atem a aspectos topológicos, mas que afetam aspectos não modelados.

Durante a execução do PFP, estas operações causam a não convergência do Fluxo. Esta situação pode ser detectada. Dentre as opções possíveis, optou-se pela penalização dos indivíduos que apresentassem não convergências. Esta penalização iria reduzir a influência destes genomas nos demais da população, ao mesmo tempo que permitiria o aproveitamento das operações codificadas até o ponto da não convergência.

Caso o genoma vencedor ainda apresente alguma não convergência, a análise da seqüência permite tomar a medida mais apropriada nos pontos da seqüência onde verificou-se esta não convergência. Por isto, estas seqüências, mesmo apresentando não convergências, não são sumariamente descartadas.

O mecanismo de penalização emprega um multiplicador, normalmente mantido em 1, que corrige o valor da função objetivo. Quando uma não convergência acontece, a este multiplicador é atribuído um novo peso que é função da posição relativa do estágio onde a não convergência ocorreu:

$$\begin{aligned} Pos_r &= \frac{Nro. Estágio}{Total de estágios} \\ ajuste &= ajuste \times Pos_r \end{aligned}$$

onde *Nro.Estágio* é representada a posição do estágio atual dentro do conjunto de estágios, de dimensão *Total de estágios*, e *ajuste* representa a correção aplicada, inicialmente igual a 1.

À medida que mais não convergências acontecem em um mesmo genoma, o valor obtido pela função objetivo vai sendo reduzido exponencialmente. Um limite para esta correção foi empregado.

4.4 Modelos de Sistemas

Utilizaram-se alguns modelos de Sistemas de Potência, dentre os disponíveis, para o desenvolvimento deste trabalho. Os modelos contemplados, durante o desenvolvimento e as experiências, foram:

- IEEE 14 barras
- IEEE 30 barras
- Sistema CEEE

O sistema IEEE 14 barras (figura 21) é um modelo referencial, utilizado para *benchmark* no desenvolvimento de diversos trabalhos na área. Foi justamente o fato da sua utilização em diversos experimentos que levou a sua escolha, além de ser um modelo extremamente simples, de fácil depuração durante os experimentos a serem implementados.

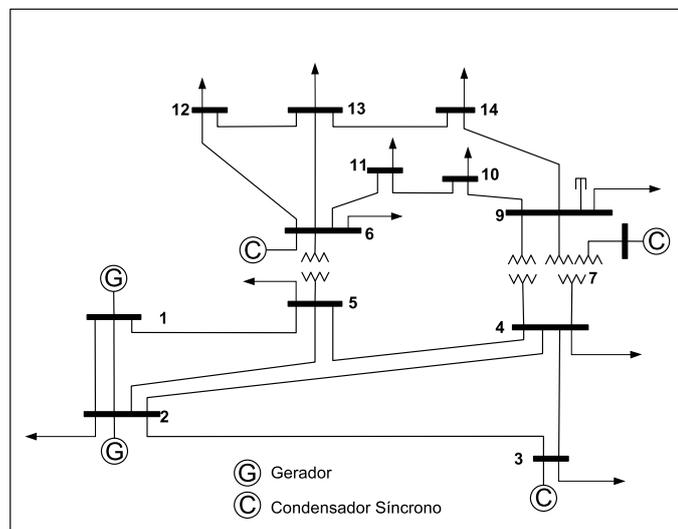


Figura 21: Sistema IEEE-14 barras

O IEEE 30 barras (figura 22), semelhantemente ao IEEE14, já foi alvo de diversos experimentos na área, e de uma complexidade levemente superior ao IEEE14. O intuito é o de utilizá-lo, também, na comparação de resultados e na depuração dos experimentos, caso necessária.

Estes dois sistemas representam partes do Sistema da American Electric Power System (AEPS), no Centro-Oeste Norte-americano, como se apresentavam em fevereiro de 1962 (IEEE-14) e Dezembro de 1961 (IEEE-30). Ambos foram obtidos do repositório existente no sítio *Power System Test Case Archive*, da Universidade de Washington.

O Sistema CEEE (figura 23) representa todo o sistema de transmissão da Companhia Estadual de Energia Elétrica (CEEE) no Estado. Sua inclusão deve-se ao fato de se ter acesso aos

seus dados, gentilmente fornecidos pela Empresa para o desenvolvimento deste trabalho. Sua dimensão permite classificá-lo como um sistema de médio porte.

4.5 Conclusão da Metodologia

A metodologia tenta uma solução simplificada para o problema de sequenciamento das operações necessárias para a RSP. A ênfase do trabalho foi aplicada na codificação do genoma, na inicialização e operações genéticas.

Do ponto de vista de sistemas de potência, muito deixou de ser implementado. Como dito, esta opção foi feita para simplificar e permitir a concentração no problema de sequenciamento.

Dentre os temas e atividades não implementadas, pode-se citar, pelo menos:

- *Ativação parcial das cargas.* As cargas são ativadas integralmente. Não se introduziu a possibilidade de ativação gradual destas cargas.
- *Priorização de cargas.* Todas as cargas apresentam o mesmo nível de importância para o AG. Uma solução alternativa poderia priorizar o atendimento a cargas escolhidas.
- *Utiliza-se apenas um Fluxo de Potência.* O Fluxo de Potência não leva em conta os aspectos dinâmicos do sistema de potência. O uso de programas analíticos auxiliares, como os citados em (ADIBI; BORKOSKI; KAFKA, 1994), poderia ser implementado para avaliar, conjuntamente ao PFP, a Função Objetivo.
- As capacidades das linhas de transmissão não são consideradas. O excesso de carga nas linhas de transmissão poderia, por exemplo, ser levado à função objetivo, permitindo que as soluções que respeitassem estes limites fossem valorizadas.

5 IMPLEMENTAÇÃO

5.1 Descrição Geral

O programa foi escrito em linguagem *C++*, no ambiente de desenvolvimento Visual Studio, da Microsoft. Este ambiente disponibiliza um conjunto de ferramentas de programação e depuração, além de um *framework* para desenvolvimento que possibilita a criação rápida de programas com uma interface gráfica bastante evoluída.

Para processar os dados gerados pelo Algoritmo Genético (AG), haviam diversas possibilidades de ferramentas. Optou-se pelo *Sistema R*, em função deste já possuir uma série de ferramentas para o tratamento, a manipulação e a análise de dados, tanto algorítmica quanto estatisticamente.

O *sistema R* é desenvolvido dentro do projeto GNU¹. Oferece uma grande diversidade de ferramentas, técnicas e recursos gráficos nas áreas de estatística e de suas aplicações. Além de ser código aberto, ele é executado em uma grande variedade de sistemas operacionais, inclusive no *MS Windows*. Além de tudo, ainda oferece a possibilidade de ser ligado à programas escritos em outras linguagens, como C/C++, e ter sua funcionalidade exposta a estes programas externos.

O emprego do *sistema R* deve-se à facilidade de criação de rotinas para manipulações de dados, sua importação e da abundância de extensões ao sistema original, nas mais diversas áreas. Algumas destas extensões são bibliotecas de funções específicas para sistemas neurais. Estas bibliotecas são denominadas *pacotes* (no original, *packages*).

A Seguir, serão descritos os detalhes de implementação do programa. Inicialmente serão apresentadas as principais estruturas de dados e as principais operações do programa. A seguir serão apresentados o programa de Fluxo de Potência e as alterações nele realizadas. O Algoritmo Genético será detalhado na seção seguinte.

¹O Projeto GNU é uma iniciativa de Software Livre de grande atuação, que influenciou a criação de muitos softwares, científicos ou não. Tem o apoio da Fundação de Software Livre (FSF). GNU é um acrônimo recursivo que significa GNU não é Unix (GNU's Not Unix).

5.2 Modelo Descritivo

Nesta seção apresenta-se a arquitetura utilizada no desenvolvimento do programa de geração de seqüências. As principais estruturas de dados e rotinas serão detalhadas, assim como o modelo UML das classes envolvidas.

Inicialmente, apresenta-se uma descrição das principais estruturas de dados do programa. Estas são os dados originais, presentes nos arquivos importados, os dados internos decorrentes da importação destes arquivos, os dados das seqüências geradas (figura 24).

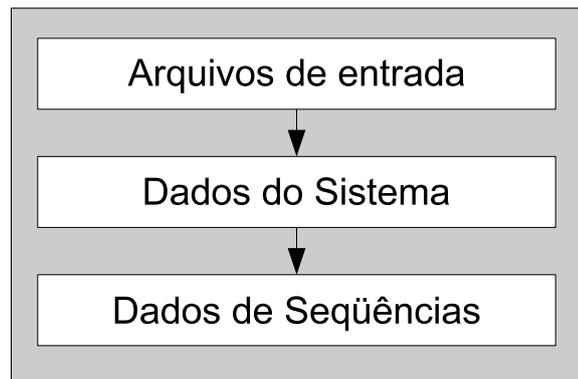


Figura 24: Diagrama da composição das estruturas de dados do programa de geração de seqüências

O programa foi dotado de capacidade de importação de três formatos de arquivos distintos:

- *PSAP* - formato da empresa Philadelphia Electric Company (PECO), que o desenvolveu para uso no seu sistema Power System Analysis Package (PSAP).
- *ANAREDE* - na verdade, uma variação do formato empregado pelo sistema POWER-MOD, da PECO, utilizado pelo Programa de Análise de Redes, ANAREDE, do Centro de Pesquisas de Energia Elétrica (CEPEL).
- *CDF* - Common Data Format (CDF), desenvolvido e difundido pelo Institute of Electrical and Electronics Engineers (IEEE).

Apesar de haverem diferenças entre estes diversos formatos, a informação que cada um deles dispõem para os sistema é, essencialmente, a mesma. Todos os dados necessários para a execução do AG são oriundos destes arquivos. As diferenças entre eles são, basicamente, morfológicas. Os formatos estão descritos nos anexos A, B e C.

Estes dados originais irão incorporar uma estrutura de dados interna. Esta estrutura é composta, na verdade, por classes em *C++*. Estas classes representam uma abstração dos

objetos reais encontrados em sistemas de potência, agregando informações de relacionamentos entre elas e funções para operar nos respectivos dados.

Desta forma, um grupo de classes foi criado para representar o sistema de potência. Um outro grupo de classes armazena os dados das seqüências, durante e após o crescimento do AG. Este grupo está representado no próximo bloco da figura 24.

Após o crescimento do AG, a seqüência escolhida é novamente simulada pelo programa de Fluxo de Potência. Nesta etapa, os valores de todas as variáveis de estado do sistema são armazenadas, em cada estágio simulado. O programa compõe, então, o arquivo com os resultados do AG. Este arquivo concatena os valores destas variáveis de estado (a partir do blecaute total) com os valores das variáveis de controle fornecidos pela seqüência, em cada estágio.

Para que haja a transformação e geração dos dados nos formatos da figura 24 são necessárias algumas operações elementares. A figura 25 ilustra as principais operações.

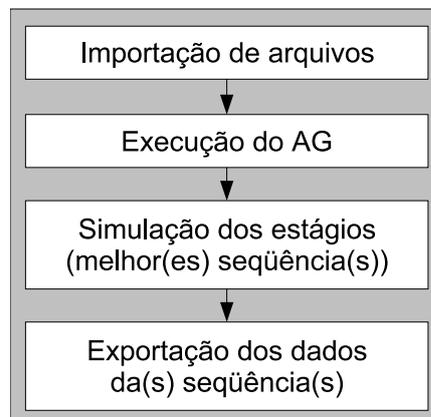


Figura 25: Diagrama da composição das estruturas de dados do programa de geração de seqüências

Os dados necessitam ser transformados de seus formatos originais, em arquivo, para a representação interna do sistema. Isto é realizado através de uma rotina de importação, com funções de leitura e transformação adequadas a cada formato interpretado.

Após a existência de um sistema nas estruturas internas do programa, pode-se processar o crescimento do AG. A execução e o controle do algoritmo são fornecidos pela biblioteca galib, e serão detalhadas no ítem 5.4.

Uma vez terminada a execução do AG, tem-se acesso ao melhor indivíduo encontrado durante o crescimento da população. Este indivíduo tem seus estágios processados, ou seja, simulados, pelo programa de Fluxo de potência e, em cada estágio processado, os valores das variáveis de estado são exportados para o arquivo de dados.

5.2.1 Representação em Linguagem UML

O programa foi desenvolvido sob a arquitetura *Document-View*, fortemente encorajada pelo *framework* MFC, da *Microsoft*. Esta arquitetura estimula a separação de funcionalidades entre a visualização dos dados e a própria estrutura e manipulação destes dados.

As classes ligadas a visualização dos dados não agregam funcionalidade ao programa, e por isso não serão apresentadas. As classes ligadas ao documento, e por sua vez, a estrutura de dados, foram divididas em dois grupos: um ligado ao próprio sistema de potência e o outro ao AG. Os dois grupos de classes podem ser visualizados nos diagramas das figuras 26 e 27.

Na figura 26, o grupo de classes superior do diagrama implementa funções de acesso a estrutura e relacionamento entre os objetos do modelo. O grupo de classes inferior é responsável por manter os dados dos componentes do sistema de potência.

Todas as classes herdam de uma classe básica que fornece serviços de identificação de cada objeto, além de possuir um índice que será utilizado para interagir com o programa de Fluxo de Potência.

A classe CSGrafo mantém estruturas que permitem o acesso a cada elemento do sistema de potência. A classe CSNodo possui uma lista de todos as arestas que incidem sobre si.

Na implementação do Algoritmo Genético, o uso da biblioteca galib permitiu a especialização de classes já existentes. A classe especializada foi a GAGenome, a partir da qual se derivou a classe CSGenome (figura 27).

A coordenação do AG está a carga da classe CSController. Esta classe instancia um objeto GAGeneticAlgorithm, que contem toda a funcionalidade para o crescimento do AG.

A interface entre o programa principal e o programa de Fluxo de Potência é realizada pela classe CIntFortran. Utilizou-se um padrão de projeto conhecido como *façade*, ou fachada, onde se procura agrupar um conjunto complexo de classes ou funções em uma interface única e mais simples.

5.3 Fluxo de Potência

O programa de Fluxo de Potência utiliza o Método Desacoplado Rápido, este, uma variação do Método de Newton-Raphson, ambos explicados nas seções 2.2.4.1 e 2.2.4.2.

O programa original iniciava sua operação lendo em um arquivo os dados das barras e linhas do sistema de potência. Após a inicialização de algumas variáveis, cria-se uma estrutura interna que armazena a topologia do sistema em matrizes. As matrizes são, então, ordenadas. Após esta etapa, calculam-se as matrizes de derivadas B' e B'' (equações 2.64 e 2.65), e as matrizes

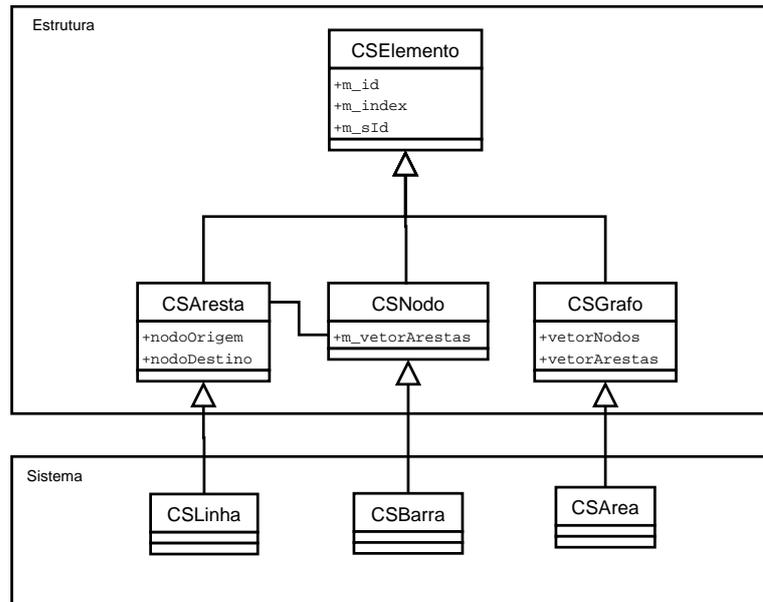


Figura 26: Diagrama UML das principais classes envolvidas com o sistema de potência

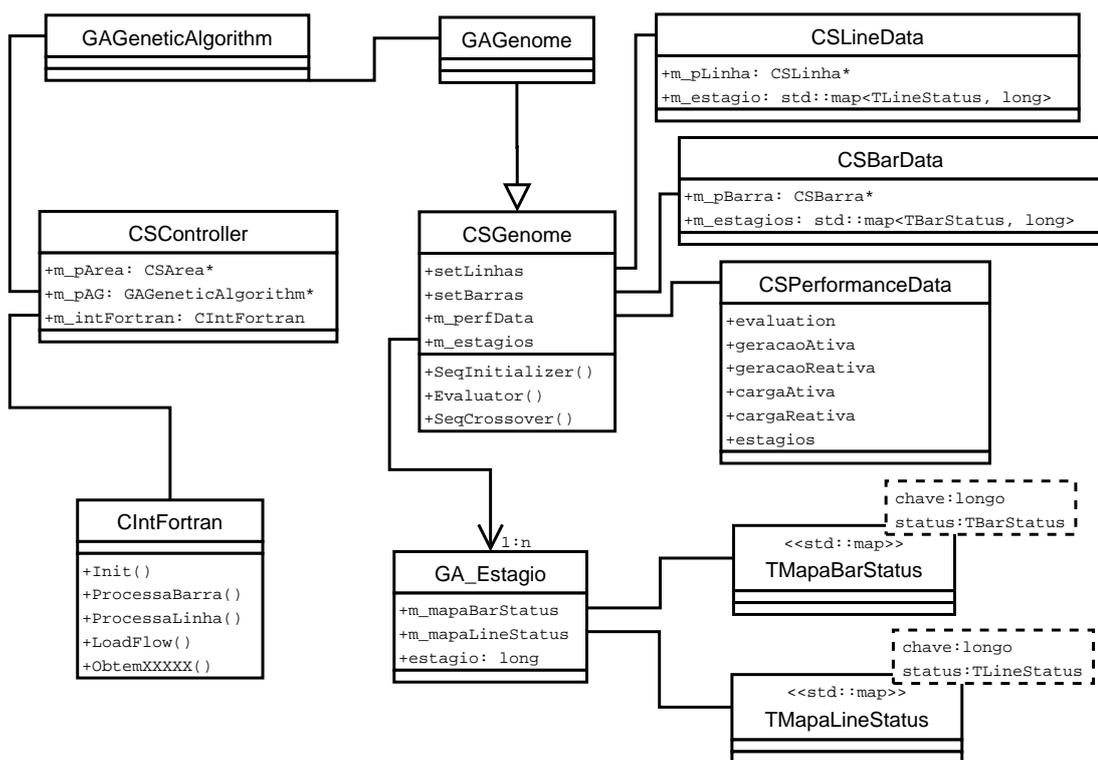


Figura 27: Diagrama UML das principais classes envolvidas no algoritmo genético

calculadas passam por um processo de redução.

O programa continua, agora calculando os desbalanços de potência ativa e reativa nas barras do sistema. É preciso verificar a convergência da solução, o que é feito após o cálculo dos desbalanços. A partir dos desbalanços de potência, pode-se calcular as demais incógnitas de cada barra, de acordo com seu tipo.

Uma vez que o programa já calculou os desbalanços de potência das barras, resta calcular o fluxo entre elas, e as perdas.

Como apresentado na metodologia (ítem 4.3), optou-se pela utilização do Programa de Fluxo de Potência (PFP) internamente ao programa principal. Para isto, o primeiro precisou ser convertido em uma biblioteca estática. A linguagem Fortran, utilizada para o desenvolvimento do PFP, permite uma alteração rápida entre o formato executável, de um programa convencional, para o de uma biblioteca. A utilização do PFP como uma biblioteca evitou a necessidade de transferência de dados através de arquivos, o que tornaria o método extremamente lento.

O processo de leitura do Sistema de Potência a partir de um arquivo, nos moldes já existentes no PFP, foi substituído por funções que permitiriam definir o sistema algorítmicamente, a comando do programa principal.

Dentro do PFP, foi feita uma re-organização das variáveis globais do programa, na sua maioria, grandes matrizes de números reais, de forma a permitir uma auto-documentação destas variáveis e uma automação do processo de transferência daquelas que seriam necessárias na interface C++ - Fortran.

Em contrapartida, no programa principal, criou-se uma classe, CIntFortran, que concentraria as funcionalidade necessárias em todo o código em linguagem C++. Esta classe expôs métodos para inicialização, definição de sistemas e solução do Fluxo de Potência destes.

A principais funções definidas foram:

- *Init()* - (Re)inicializa as variáveis do PFP. Precisa ser chamada a cada nova solução, antes da definição de barras e linhas.
- *ProcessaBarra()* - Utilizada para definir os dados estáticos das barras do sistema.
- *ProcessaLinha()* - Utilizada para definir os dados estáticos das linhas do sistema. Caso alguma das barras nas quais a linha incide não tenha sido definida, uma exceção é lançada.
- *LoadFlow()* - Utilizada para a solução do Fluxo de Potência do sistema definido pelo conjunto de barras e linhas, após configurado com as respectivas funções.

Além das funções principais descritas, o acesso aos dados das barras e linhas é executado através de funções específicas.

O isolamento obtido com o uso do padrão *façade* permite a rápida substituição do PFP por outra versão deste, ou mesmo por outro programa. Neste caso, o impacto das alterações estaria confinado a classe CIntFortran.

5.4 Algoritmo Genético

No o desenvolvimento do algoritmo genético foi utilizada a GALib, uma biblioteca de componentes C++ para algoritmos genéticos. A biblioteca foi desenvolvida no Departamento de Engenharia Mecânica do Instituto de Tecnologia de Massachusetts (MIT) (WALL, 1996).

A *GAlib* oferece uma estrutura de classes que implementam grande parte das atividades necessárias ao crescimento de um algoritmo genético, desde a criação de genomas básicos ao controle da evolução de uma população.

Esta estrutura auxilia a atender dois dos requisitos básicos para a implementação de um AG:

- Definição de uma representação
- Definição de operadores genéticos

Para definir a representação, a GALib implementa quatro classes especializadas de genomas: GAListGenome, GATreeGenome, GAArrayGenome e GABinaryStringGenome. Estas classes herdam de GAGenome, adicionando uma estrutura de dados à classe ancestral, de acordo com o indicado pelo nome de cada classe. Ainda é possível a herança de cada uma destas quatro classes especializadas, ou mesmo da classe básica, GAGenome.

A biblioteca ainda define alguns tipos básicos de algoritmos genéticos, como o simples, o incremental e o de estado estável (*steady-state*). Estes algoritmos diferem na maneira como criam novos indivíduos e como repõem os indivíduos das gerações anteriores durante a evolução.

Os algoritmos genéticos da biblioteca contém as estatísticas, estratégias de reposição e parâmetros para controle do algoritmo. Contém um objeto população: um *container* de genomas. Contém também os operadores de seleção e de escalonamento.

Para a implementação, desenvolveu-se a estrutura de classes ilustrada na figura 27. Durante a execução do algoritmo, os objetos envolvidos trocam mensagens, ou simplesmente, fazem chamadas de funções. As principais chamadas realizadas podem ser vistas na figura 28, um diagrama de colaboração UML.

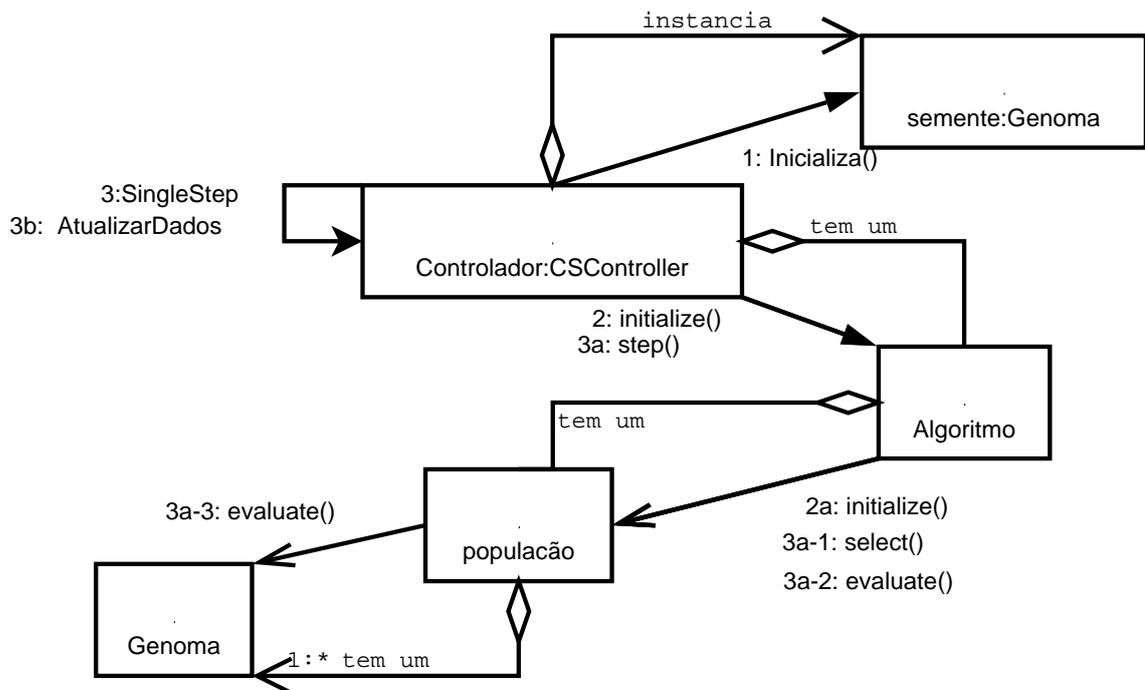


Figura 28: Diagrama de Colaboração ilustrando a seqüência de chamadas de funções e os relacionamentos entre objetos do módulo genético

Neste diagrama vê-se que o objeto *controlador* instancia um objeto *genoma* que será clonado no processo de inicialização da população. o *controlador* é encarregado de executar cada passo do AG, após o qual ele atualiza os elementos de Interface com o Usuário (UI) do programa. O objeto *algoritmo*, por sua vez, encapsula toda a operação da população, desde a criação e inicialização até a aplicação dos operadores genéticos. Cada *genoma* pertencente a esta população.



Figura 29: Fluxograma de uma algoritmo genético básico, como implementado pela *GAlib*.

O algoritmo completo implementado pela biblioteca pode ser visualizado na figura 29. Após a inicialização da população, têm início um processo iterativo que passa pelos passos:

- Seleção para reprodução: na biblioteca, é realizada pela classe `GAPopulation`.
- Cruzamento dos pais: também possui implementações padrão na biblioteca, dentro de cada classe padrão derivada de `GAGenome`.
- Mutação dos filhos: implementação padrão em cada classe `GAGenome` especializada.
- Inserção dos filhos na população: implementada no nível da classe `GAPopulation`.
- Verificação dos critérios de parada e controle do laço de iteração

Todas estas funções, além de possuírem diversas implementações padrão que podem ser escolhidas pelo desenvolvedor, ainda podem ser especializadas de acordo com as necessidades que surjam em cada projeto.

Os genomas instanciados pelo programa são instâncias da classe `CSGenoma`, e para eles foram escritas algumas funções específicas, incluindo operadores, descritas a seguir.

5.4.1 Implementação do Genoma

Descreve-se, primeiramente, os detalhes de implementação da classe `CSGenoma`. Conforme mencionado anteriormente (ítems 4.2 e 5.2.1), a classe `CSGenoma` herdou da classe `GATreeGenome`. Originalmente, a representação escolhida previa apenas um grafo representando o sistema de potência, onde cada nodo do grafo indicaria apenas o estágio no qual a barra correspondente seria ativada. Durante este estágio de desenvolvimento, utilizou-se a classe `GATreeGenome`, em cuja árvore interna foram armazenadas as barras com o respectivo estado de ativação. Esta representação falhava em controlar estados diversos, pois apenas a ativação era contemplada. Como optou-se por determinar nos estágios as diversas operações pelas quais cada barra deveria passar, outra representação foi necessária.

Na representação do genoma implementada, cada estágio armazenado na lista dos indivíduos é uma instância da classe `GA_Estagio`. Estes objetos possuem listas de estados de linhas e de barras, e representam estágios discretos com todas as operações a serem realizadas durante sua extensão. A determinação dos estados de cada estágio é realizada durante a inicialização dos genomas, e será detalhada a seguir.

5.4.2 Inicialização de cada indivíduo

Descreve-se, neste ítem, o método de inicialização mencionado no ítem 4.2.2. Inicialmente, cada genoma possui a estrutura da árvore que representa o grafo de conexões do sistema de

potência, a partir dos geradores auto-alimentados (unidades *black-start*). Esta estrutura está implementada na árvore membro da classe *GATreeGenome*, remanescente da implementação anterior, substituída.

Para a inicialização final de cada genoma, preenche-se uma lista com o identificador de cada nodo da árvore, que ainda não foi completamente conectado. Esta lista é iterativamente pesquisada, e as barras e linhas de transmissão associadas a cada nodo da árvore vão tendo seus estados alterados, progressivamente, à medida que se avança na estrutura da árvore. Quando um nodo passa a estar completamente conectado, é removido da lista de nodos não conectados, até que o número de nodos remanescentes nesta lista seja inferior a um limite, randomicamente escolhido para cada individuo da população. Esta randomicidade garante que o processo de inicialização execute adequadamente, sem laços infinitos, já que é possível que sempre restem nodos sem conexão. O processo de cruzamento deverá garantir que os indivíduos com os melhores conjuntos de nodos conectados tenham performance superior aos demais, ao passo que aqueles que apresentaram conjuntos maiores de nodos sem conexão sejam penalizados e eliminados da população.

Mesmo sendo removidos da lista de nodos a conectar, os nodos já conectados ainda tem sorteadas as possibilidades de conexão de seus geradores, cargas e capacitores. A cada nodo visitado, seu estado é pesquisado e sua progressão, dentro dos estágios possíveis para ele, é sorteada aleatoriamente. Existem variáveis de controle para se alterar o comportamento destes sorteios aleatórios.

5.5 Exportação dos Dados

Os dados da seqüência escolhida pelo AG como a melhor solução para o sistema são exportados para arquivos, em modo texto, com os campos separados por algum delimitador. Selecionou-se o ponto-e-vírgula (;) como delimitador.

A estrutura dos dados no arquivo de exportação foi escolhida de forma a permitir a importação destes dados para dentro do *Sistema R*. O formato escolhido listava todos os dados de entrada para a rede e, a seguir, as respostas esperadas.

Além dos dados da seqüência vencedora, os dados gerados durante a evolução do AG também são armazenados em arquivos que podem ser importados no *Sistema R*. Estes dados são processados para a avaliação do comportamento do AG e geraram as figuras presentes neste relatório.

6 *EXPERIMENTOS E RESULTADOS*

Neste capítulo, descrevem-se os experimentos realizados com o sistema desenvolvido. Cada experimento terá seus passos detalhados e seus objetivos expostos.

Os experimentos focalizam a descoberta das seqüências de operações sobre os componentes do Sistema de Potência. Sob este prisma, são verificados, além da evolução, medida em cada experimento pela respectiva função objetivo, também o comportamento das grandezas elétricas durante esta evolução e o comportamento destas grandezas na seqüência de operações vencedora de cada experimento.

Durante a evolução de cada experimento, acompanham-se as grandezas elétricas totais de cada sistema. Os valores registrados referem-se às potências ativa e reativa do melhor genoma de cada geração. Como são estas as grandezas que originam as avaliações de cada genoma, pode-se também verificar os possíveis motivos para que os valores obtidos de cada função objetivo apresentem seus respectivos comportamentos.

Após a evolução de cada experimento, os melhores genomas são detalhados. Este detalhamento se dá pela representação gráfica das mesmas grandezas elétricas apresentadas nos resultados evolutivos. Como cada estágio de restauração codificado nos genomas corresponde a um grupo de operações executadas sobre o sistema de potência, o qual dispense um determinado tempo para sua execução, pode-se traçar uma analogia entre a execução consecutiva destes estágios e o tempo de restauração decorrido. A análise deste conjunto de dados revela o perfil temporal da solução representada por cada genoma.

Os experimentos passam por duas classificações distintas. Uma diz respeito ao próprio Algoritmo Genético (AG), onde se dividem os experimentos em três grupos com diferentes combinações de tamanho de população e número de gerações. O objetivo desta divisão é verificar a influência da alteração destas propriedades na qualidade das respostas obtidas.

Outra classificação possível diz respeito à métrica que é utilizada na formulação de cada função objetivo. Cada uma das 5 funções objetivo codificadas no algoritmo passa por um conjunto de experimentos. Estas funções são controladas por variáveis, anteriormente mencionadas como constantes de ajuste (seção 4.2.7). A variação destes parâmetros permite verificar o quanto cada um deles influi na qualidade das respostas obtidas e também na performance de suas res-

pectivas funções objetivo.

6.0.1 Experimentos com Funções Objetivo

A tabela 1 relaciona os experimentos realizados na etapa evolutiva.

Tabela 1: Relação dos parâmetros utilizados nos experimentos realizados.

Função Objetivo	Parâmetro da Função			Sistema de Potência		
	A	B	C	IEEE 14	IEEE 30	CEEE
0	1	1	-	30	X	
	1	1	0	X	X	X
1	1	2	0	X	X	
	1	5	0	X	X	
	1	1	0	X	X	
2	1	1	1	X	X	X
	1	1	2	X	X	
	1	2	1	X	X	
	0	1	1	X	X	
3	0	2	1	X	X	
	1	1	1	X	X	X
	1	2	1	X	X	
	1	1	1	X	X	
4	1	2	1	X	X	X
	1	5	1	X	X	X

6.0.2 Variação de Parâmetros do AG

Os experimentos mencionados na tabela 1 são repetidos para as combinações de população e gerações citadas na tabela 2, à exceção dos experimentos do sistema CEEE, que seguem seqüência própria, conforme seção 6.1.6.

Tabela 2: Relação do tamanho das populações e do número de gerações utilizados em cada grupo de experimentos

População	Gerações
25	25
50	50
50	100

6.1 Resultados do Algoritmo Genético

6.1.1 Função Objetivo 0

O primeiro conjunto de experimentos utiliza a função objetivo 0, apresentada na seção 4.2.7.1.1, equação 4.1, abaixo repetida:

$$FO_1 = \sum_{e=1}^N \frac{P_{AT_e} A + P_{RE_e} B}{e} \quad (4.1)$$

6.1.1.1 Sistema IEEE 14 Barras

O sistema IEEE 14 representa uma parte do sistema da American Electric Power System (AEPS), no Centro Oeste Norte-americano. As potências máximas a serem atendidas neste sistema são de 259 MW e 75 MVAR. Ele possui 14 barras de potência e 20 linhas de transmissão.

6.1.1.1.1 População: 25 - Gerações: 25

O primeiro teste foi realizado com uma população de 25 indivíduos, executando por 25 gerações. Os resultados podem ser visualizados na figura 30.

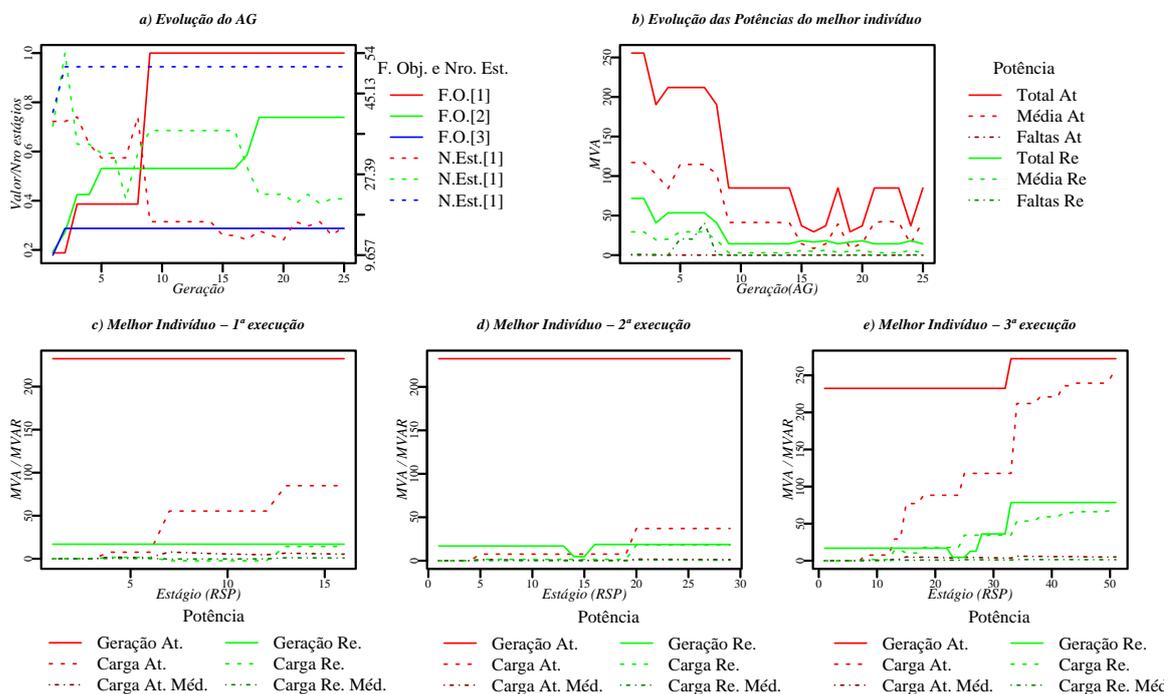


Figura 30: Resultados para o sistema IEEE14, com a população=25, 25 gerações, função objetivo=0, com parâmetros A=1 e B=1.

As séries de figuras que segue estão padronizadas, seguindo dois modelos diferentes de layout, e aqui são apresentadas detalhadamente. Nas demais figuras apresentadas, comentam-se apenas as observações pertinentes aos resultados específicos.

A subfigura 30a representa a evolução do AG. No eixo ordenado, tem-se as gerações. No 1º eixo coordenado, o valor normalizado que cada indivíduo apresenta, para o experimento em

questão, e no 2^0 , o número de estágios. As duas séries de dados representam, em linhas contínuas, a evolução da função objetivo e, em linhas tracejadas, o número de estágios encontrado em cada geração. Nesta subfigura, observa-se que o valor obtido pelos genomas, ao longo das gerações, aumentou. Este comportamento é esperado para todos os experimentos, pois indica a evolução de cada população de genomas. Entretanto, o número de estágios de restauração, apesar de também reduzir-se ao longo das gerações, o faz de modo irregular. Isto também é esperado, já que nesta função objetivo não se está otimizando esta variável.

A subfigura b, ainda na parte superior da figura 30, representa o comportamento de algumas grandezas elétricas durante a evolução do AG, e permite acompanhar visualmente a relação entre estas grandezas e o valor da função objetivo, para cada experimento. As grandezas apresentadas nesta subfigura são as potências atendidas nos estágios finais de cada genoma, as potências médias atendidas e a média de faltas de potência ao longo dos estágios de cada genoma. Especificamente, nestes experimentos, verifica-se que as potências finais atendidas e as potências atendidas médias caem abruptamente nos últimos estágios do experimento.

Este comportamento é resultado da não convergência do Programa de Fluxo de Potência (PFP), a partir de certo estágio da evolução do AG. Como a função objetivo em questão *não* considera os valores finais destas potências, mas sim sua evolução ao longo dos estágios da solução, esta não convergência do PFP passa despercebida pela função.

A linha inferior da figura 30 apresenta, nas subfiguras c, d e e, o comportamento dos melhores indivíduos de cada uma das três execuções de cada experiência. No eixo ordenado, indicam-se os estágios de *restauração* propostos e, no eixo das coordenadas, os valores de algumas das grandezas elétricas mais importantes, obtidas através da solução do Fluxo de Potência, com as soluções sugeridas pelo AG. As grandezas escolhidas dizem respeito às potências ativa e reativa, apresentando a geração comandada, em linhas contínuas, a carga atendida no estágio final de cada solução, em linhas tracejadas, e a carga atendida média ao longo dos estágios de cada solução, com linhas mistas, traço-ponto.

6.1.1.1.2 População: 50 - Gerações: 50

A figura 31 apresenta os resultados do mesmo experimento, repetido com o aumento do número de indivíduos e de gerações. Nesta figura, apresenta-se, de forma mais compacta, os resultados deste experimento. Na subfigura *a*, vêem-se a evolução das respostas da função objetivo e o número de estágios presente em cada solução apresentada. Na subfigura *b*, o comportamento das potências ativa e reativa atendidas, as finais em linhas contínuas, as médias em linhas tracejadas e as médias de faltas de gerações, em linhas traço-ponto.

O aumento do número de gerações não modificou significativamente os resultados obtidos

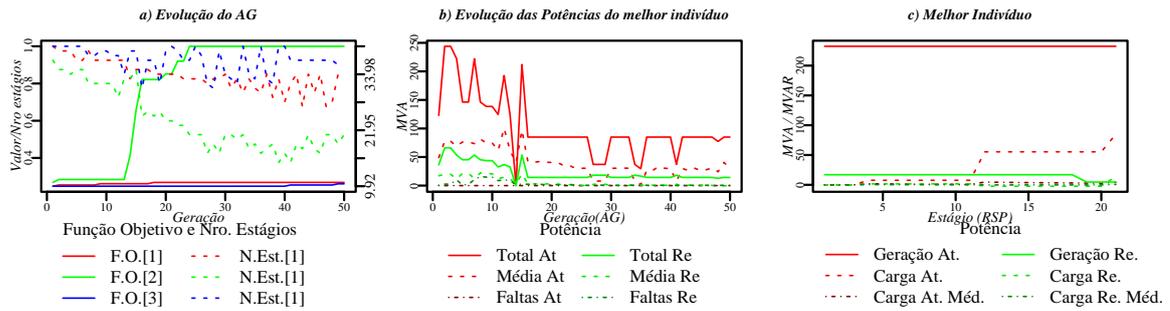


Figura 31: Sistema IEEE14, população=50, gerações=50, função=0, A=1 e B=1.

com esta função objetivo. Na verdade, o melhor cromossomo da população produziu resultados inferiores aos obtidos com população e indivíduos iguais a 25 (figura 30), o que reflete a inadequação desta função objetivo.

6.1.1.1.3 População: 50 - Gerações: 100

Mantendo-se a população em 50 indivíduos, mas com 100 gerações, não se obteve melhores resultados. Os resultados podem ser visualizados na figura 32. Observando-se a subfigura 32b, vê-se que uma das funções convergiu, mas seu resultado é tão inferior ao do experimento com população e gerações iguais a 25 quanto o experimento anterior. Novamente, como a função é insensível ao último estágio de cada solução, e como é deste estágio que são obtidos os dados presentes nesta subfigura, este resultado pode ser tido como normal.

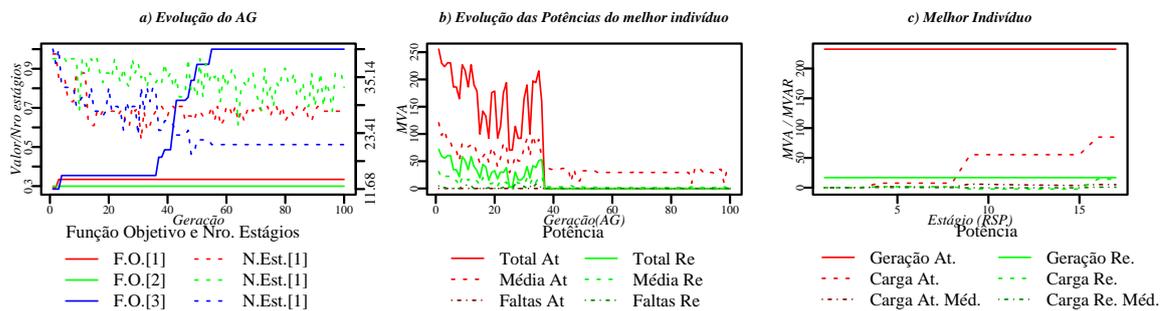


Figura 32: Sistema IEEE14, população=50, gerações=100, função=0, A=1 e B=1.

6.1.1.2 Sistema IEEE 30 barras

O sistema IEEE 30 representa uma parte do sistema da AEPS, no Centro Oeste Norteamericano. As potências máximas a serem atendidas neste sistema são de 283,4 MW e 126,2 MVAR. Ele possui 30 barras de potência e 41 linhas de transmissão.

6.1.1.2.1 População: 25 - Gerações: 25

Os experimentos realizados com o IEEE 30 barras mostram melhores resultados com esta função objetivo. Isto pode ser visualizado na figura 33, onde se percebe um comportamento bem mais estável, em especial na subfigura *b*, nas potências finais e médias. Os resultados dos indivíduos, nas subfiguras *c*, *d* e *e*, também foram mais estáveis, onde os três genomas apresentam comportamento semelhante. Os resultados das subfiguras *a* e *c*, o melhor indivíduo, atendem rapidamente uma parcela expressiva da carga total do sistema.

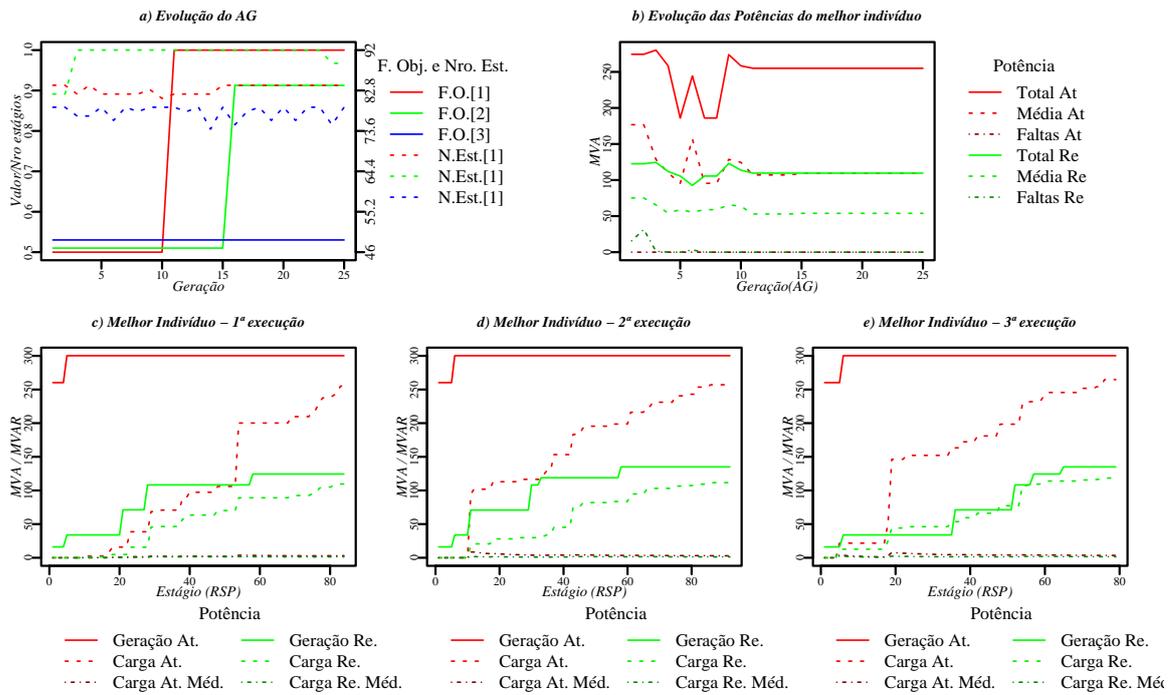


Figura 33: Resultados para o sistema IEEE30, com a população=25, 25 gerações, função objetivo=0, com parâmetros A=1 e B=1.

6.1.1.2.2 Demais populações

Os resultados obtidos com este sistema e com esta função objetivo, para as demais combinações de populações e gerações apresentaram comportamento semelhante, e seus resultados completos podem ser vistos no apêndice E.2.1.

6.1.2 Função Objetivo 1

A função objetivo 1 foi apresentada na seção 4.2.7.1.2, na equação (4.2). Esta equação é repetida abaixo para melhor visualização.

$$F_{O_2} = \sum_{b=1}^M \frac{(P_{AT_b} + P_{RE_b})A}{1 + B * N} \quad (4.2)$$

A inclusão do número total de estágios de cada genoma na função objetivo visa a redução deste valor, e os resultados a seguir demonstram sua eficiência.

6.1.2.1 Sistema IEEE 14 Barras

6.1.2.1.1 População: 25 - Gerações: 25

Os resultados obtidos com esta função objetivo para pequenas populações e poucas gerações pode ser visualizado na figura 34.

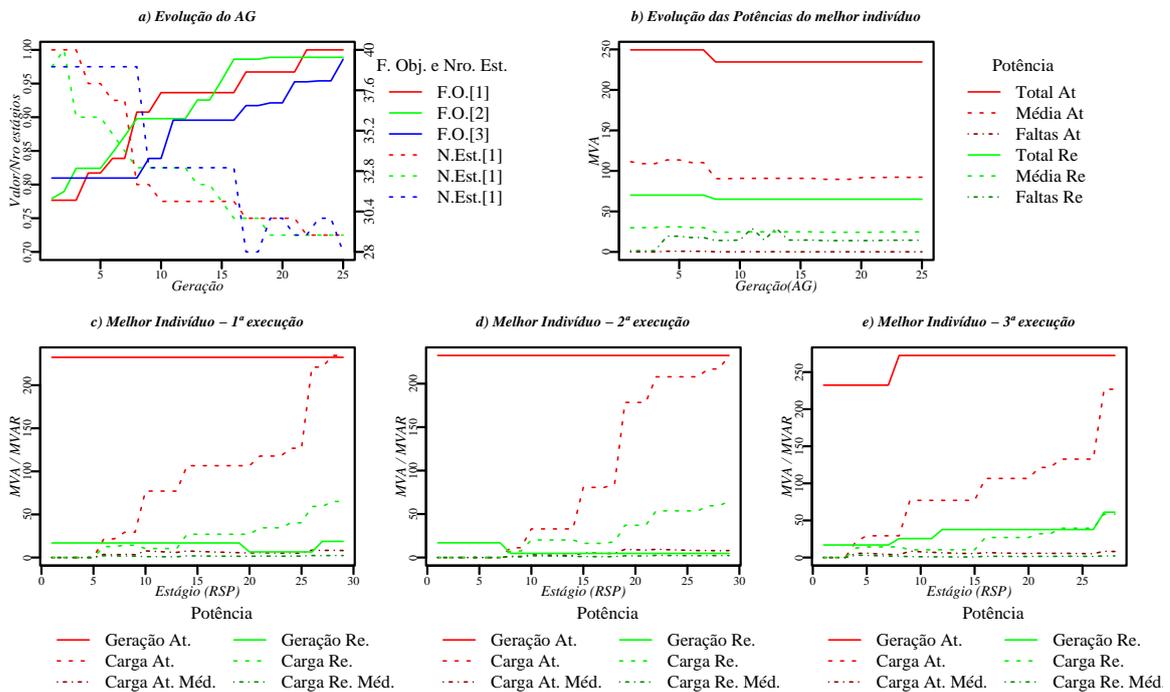


Figura 34: Resultados para o sistema IEEE14, com a população=25, 25 gerações, função objetivo=1, com parâmetros A=1 e B=2.

A subfigura *a* mostra que houve evolução no valor retornado pela função objetivo e que houve uma redução média de cerca de dez estágios entre os três melhores indivíduos do experimento. A subfigura *b* mostra que as potências ativa e reativa mantiveram-se quase constantes durante toda a evolução dos experimentos, o que indica que não houve melhora no atendimento destas, já que a quase totalidade das cargas ativa e reativa encontram-se atendidas já nas primeiras gerações do AG.

A comparação do número de estágios das soluções encontradas neste experimento com os ob-

tidos com a função objetivo 0 mostram que houve uma redução deste número neste experimento, conforme esperado.

6.1.2.1.2 Demais Populações

O comportamento dos resultados desta função objetivo, em relação aos parâmetros populacionais do AG, mostrou-se inalterado. Os resultados completos podem ser visualizados no Anexo E.1.2.

6.1.2.2 Sistema IEEE 30 Barras

6.1.2.2.1 População: 25 - Gerações: 25

A figura 35 apresenta os resultados obtidos com a função objetivo 1 e o sistema IEEE 30 barras, para pequenas populações.

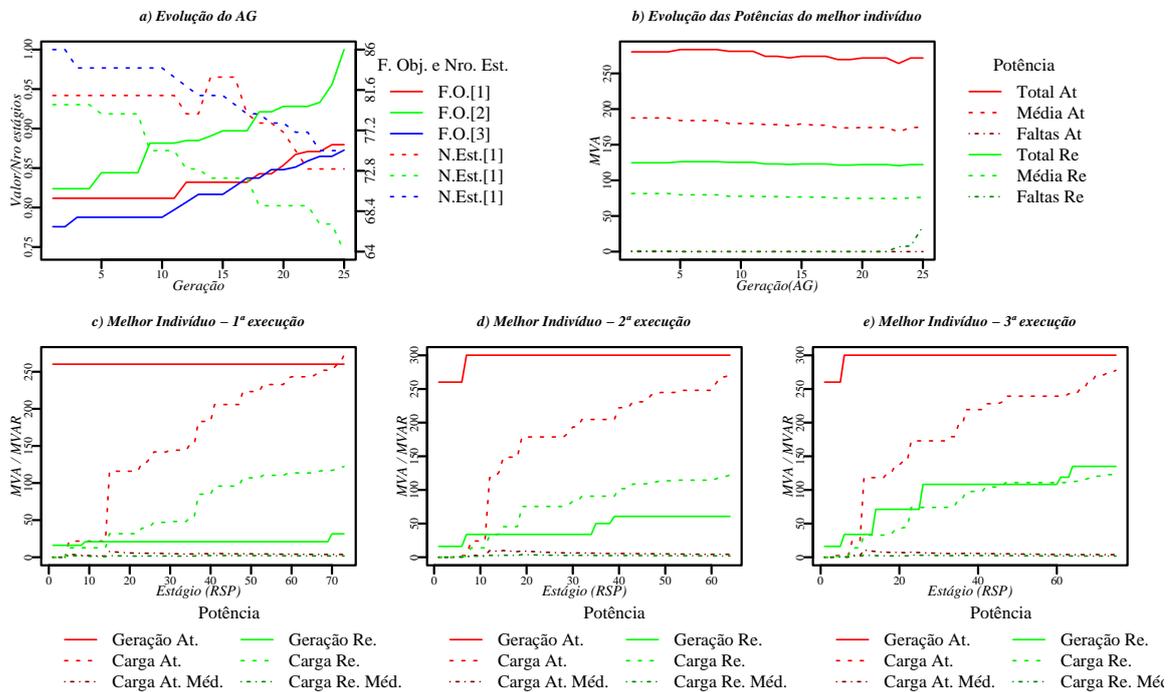


Figura 35: Resultados para o sistema IEEE30, com a população=25, 25 gerações, função objetivo=1, com parâmetros A=1 e B=2.

Os valores lidos na subfigura a mostram que houve redução no número de estágios de solução nos três melhores cromossomos. Esta redução foi inferior a 10 estágios, o que pode ser aferido ao se comparar estes números com os da função objetivo 0 (seção 6.1.1.2).

A alteração do parâmetro B, de 1 para 5, prioriza a otimização do número de estágios na

função objetivo. Os resultados desta alteração podem ser vistos na figura 36. A subfigura *a* mostra que o melhor genoma teve menos de 70 estágios. Neste experimento, houve uma redução de 14 estágios em um dos cromossomos .

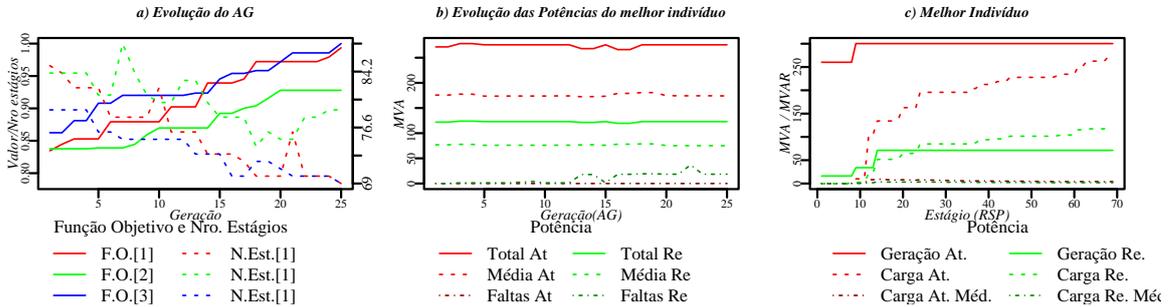


Figura 36: Sistema IEEE30, população=25, gerações=25, função=1, A=1, B=5 e C=0.

6.1.2.2.2 Demais populações

O efeito do aumento do tamanho da população neste experimento pode ser visualizado na figura 37. O número de estágios do melhor indivíduo chegou a cerca de 40 estágios, passando por uma redução de quase 40 estágios.

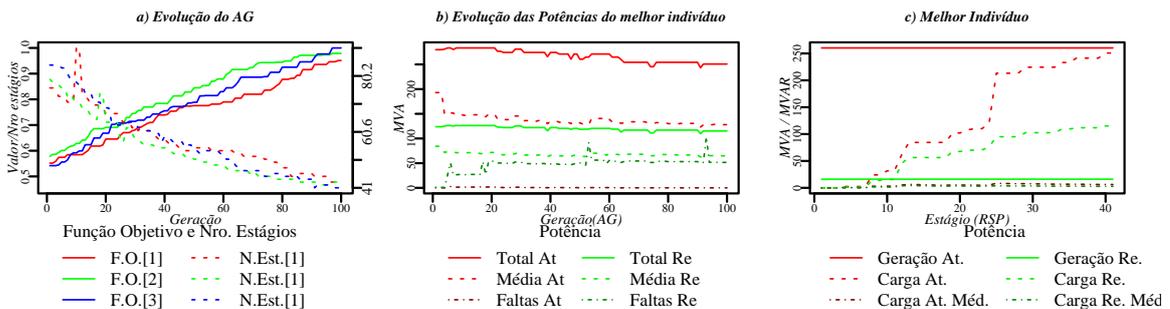


Figura 37: Sistema IEEE30, população=50, gerações=100, função=1, A=1, B=2 e C=0.

6.1.3 Função Objetivo 2

Na seção 4.2.7.1.3 apresenta-se a função objetivo 2 (equação (4.3)). Esta encontra-se novamente transcrita abaixo, para o acompanhamento dos resultados.

$$F_{O_3} = \frac{\sum_{b=1}^M A(P_{AT_b} + P_{RE_b}) - B(\Delta P)}{1 + C * N} \quad (4.3)$$

$$\Delta P = |P_{GE_b} - P_{AT_b}|$$

O termo ΔP procura atuar sobre as diferenças entre a potência atendida em cada estágio e a geração existente no mesmo. A função objetivo procura minimizar estas diferenças, fazendo

com que o sistema apresente sempre o mínimo de geração necessário para atender a potência das cargas acionadas em cada estágio.

A seguir, apresentam-se os resultados dos experimentos com esta função objetivo.

6.1.3.1 Sistema IEEE 14 Barras

6.1.3.1.1 População: 25 - Gerações: 25

Observando-se os resultados das subfiguras *a* e *b*, tem-se a impressão de que este experimento se comportou identicamente ao da função objetivo 1. Entretanto, observando especificamente às subfiguras 34b, 34c, 34e, 35b e 35c, percebe-se que nos resultados dos experimentos anteriores o nível da média das faltas de geração de energia era maior.

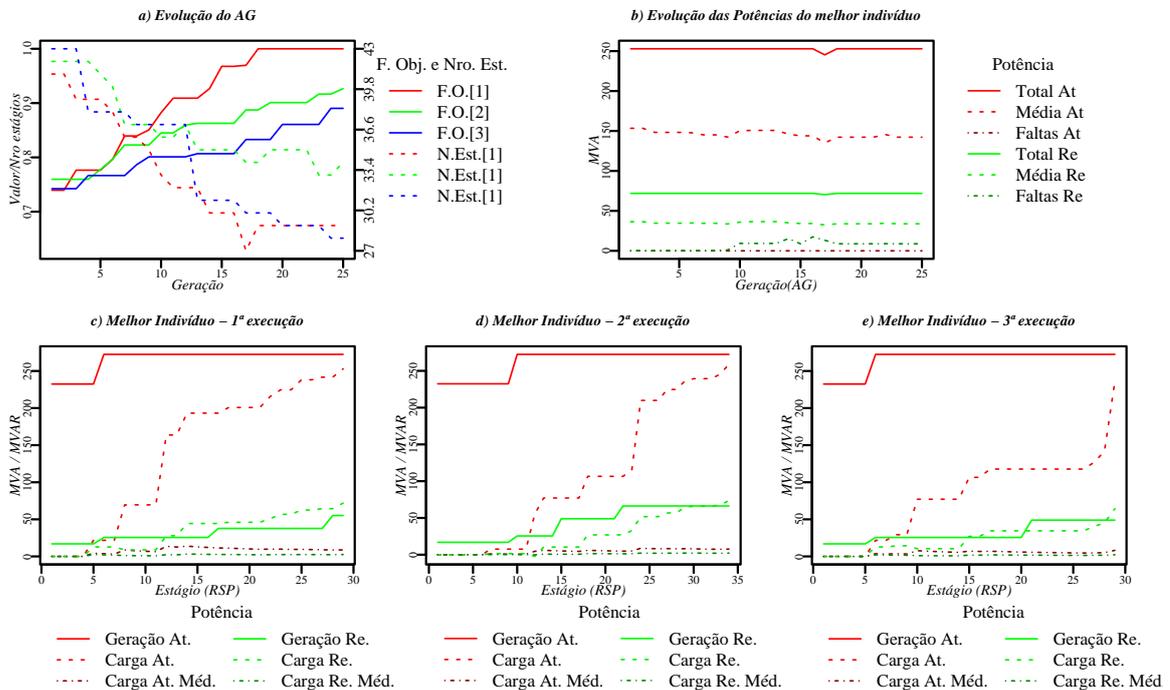


Figura 38: Resultados para o sistema IEEE14, com a população=25, 25 gerações, função objetivo=2, com parâmetros A=1, B=1 e C=1.

Comparando-se as subfiguras 38c, 38d e 38e com as anteriores, pode-se perceber que os momentos onde as potências atendidas excedem as potências geradas são, em geral, menores, tanto em valor quanto em duração. Os estágios onde ocorrem as faltas só estão presentes nas potências reativas. As faltas de geração de potências ativas, por terem maior capacidade de geração, são facilmente controladas por esta função objetivo.

6.1.3.1.2 Demais populações

O aumento do número de geração e do tamanho da população permite ao AG procurar por melhores resultados, em especial, nesta função objetivo.

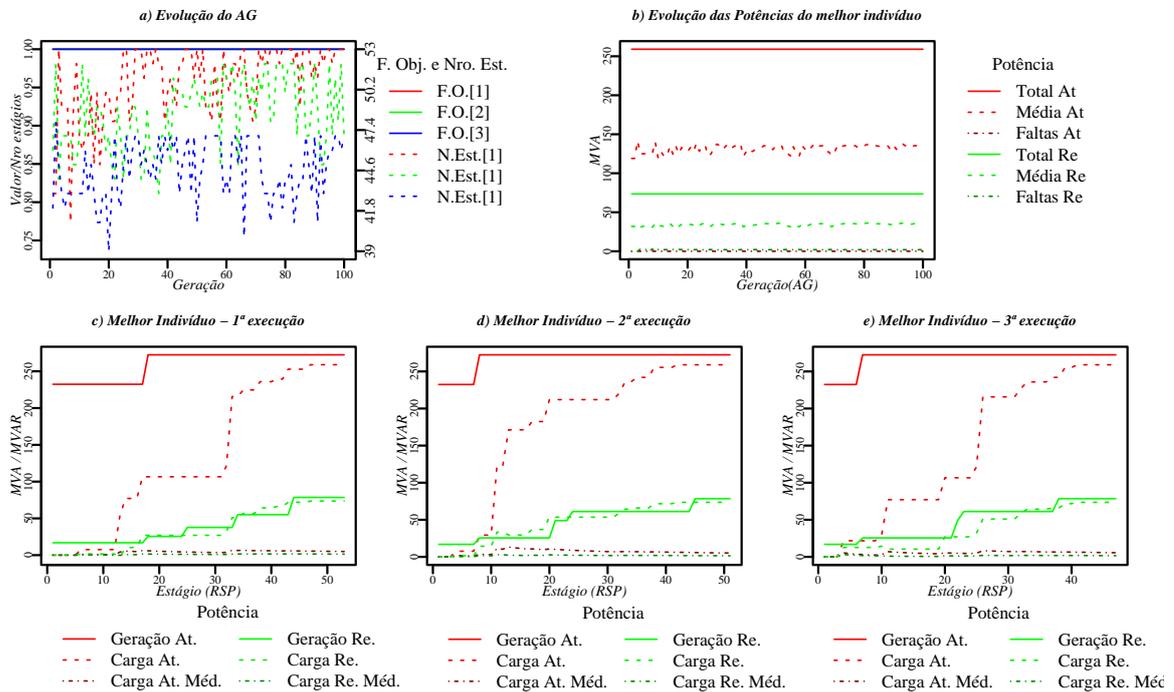


Figura 39: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=1 e C=0.

Na figura 39 vê-se o resultado do experimento com 50 indivíduos na população, e executando por 100 gerações. O parâmetro C deste experimento foi deixado em 0, o que justifica o comportamento que se verifica na subfigura 39a, onde o número de estágios não está mais sendo otimizado. Entretanto, as faltas de geração, na subfigura 39b são praticamente nulas. Observando-se os melhores indivíduos, subfiguras 39c, d e e, pode-se ver que são poucos os estágios onde há excesso de potência atendida, em relação à potência gerada.

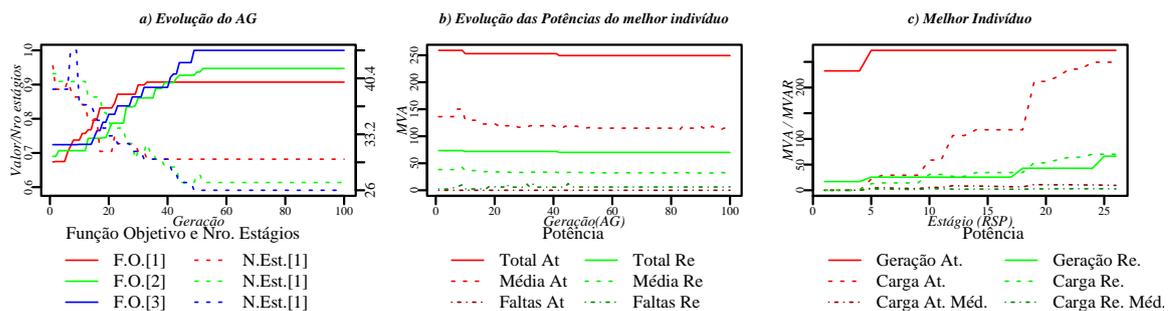


Figura 40: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=1 e C=1.

A figura 40 mostra o resultado para o mesmo experimento, no qual se alterou o valor de

C para 1. Aqui, novamente, o AG tenta otimizar o número de estágios e, ao mesmo tempo, minimizar as faltas de geração. O melhor indivíduo do experimento se comporta como o da figura 39, porém o número de estágios não é reduzido, comparado com o resultado desta mesma figura, ao contrário do que era esperado.

6.1.3.2 Sistema IEEE 30 Barras

A figura 41 apresenta os resultados para a função objetivo 2 aplicada ao sistema IEEE30 barras, com população e gerações iguais a 25. O comportamento do AG sobre este sistema é como esperado. Como indicado na subfigura 41a, há uma redução de, aproximadamente, 12 estágios no melhor genoma. A subfigura 41c mostra um perfil de potências onde não existem grandes faltas de geração. Há uma performance levemente melhor deste sistema, em relação ao IEEE 14 barras equivalente (figura 38), possivelmente relacionada com a maior quantidade de geradores presentes neste sistema, o que permite uma maior quantidade de combinações de equipamentos geradores para atender às cargas ativadas pela solução codificada no genoma.

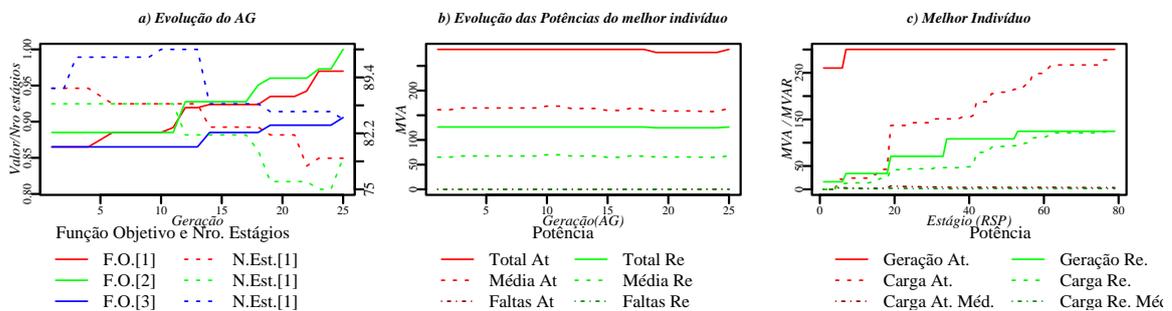


Figura 41: Sistema IEEE30, população=25, gerações=25, função=2, A=1, B=2 e C=1.

O aumento do número de indivíduos na população e no número de gerações do experimento não obteve melhores resultados do que os obtidos com populações e gerações inferiores. Isto pode ser verificado na figura 42, onde percebe-se que as soluções são muito parecidas.

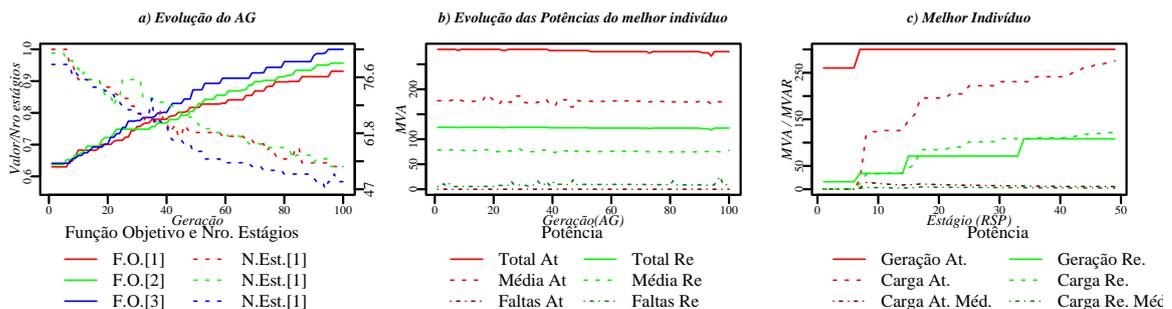


Figura 42: Sistema IEEE30, população=50, gerações=100, função=2, A=1, B=1 e C=1.

6.1.4 Função Objetivo 3

A função objetivo 3, descrita na seção 4.2.7.1.4, equação (4.5), representa uma variação da função anterior. Ela pode ser acompanhada a seguir, onde foi repetida para facilitar o acompanhamento dos resultados.

$$FO_4 = \frac{\sum_{b=1}^M P_{AT_b} + P_{RE_b} - A \cdot P_{Ex_b} - B \cdot P_{Fl_b}}{1 + C * N} \quad (4.5)$$

Nesta função, avaliam-se separadamente os excessos e as faltas de geração de potência.

6.1.4.1 Sistema IEEE 14 Barras

As figuras 43 e 44 representam os resultados de dois experimentos com parâmetros parecidos. No primeiro experimento, o valor do parâmetro A é anulado, fazendo com que o AG reduza ao máximo as faltas de geração neste experimento. Quando o valor de A é alterado para 1 (figura 44), não se obtém observam grandes alterações, indicando que o valor destes parâmetros precise ser adequadamente mais elevado.

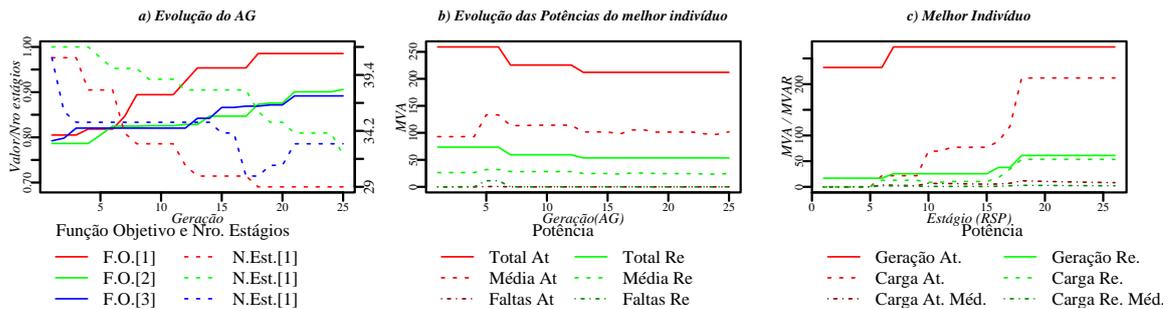


Figura 43: Sistema IEEE14, população=25, gerações=25, função=3, A=0, B=2 e C=1.

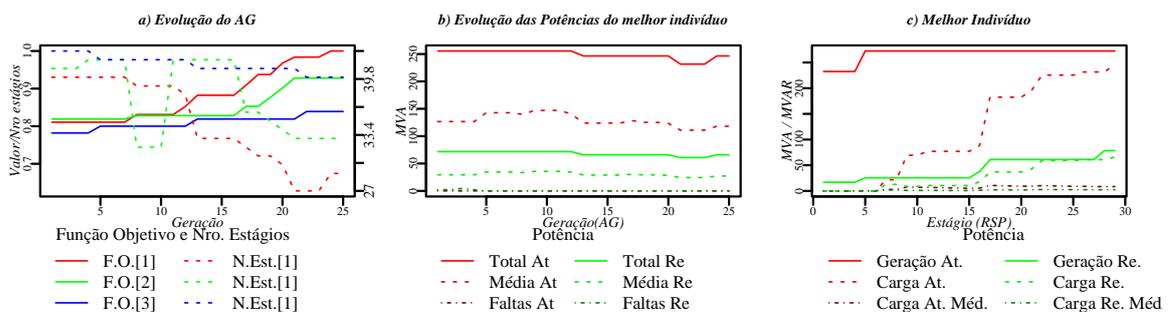


Figura 44: Sistema IEEE14, população=25, gerações=25, função=3, A=1, B=2 e C=1.

6.1.4.2 Sistema IEEE 30 Barras

As figuras 45 e 46 mostram os mesmos resultados obtidos para o sistema IEEE 30 barras. A alteração do parâmetro A para 1 causa uma pequena redução no valor dos excessos de geração, através do atendimento mais imediato das cargas do sistema.

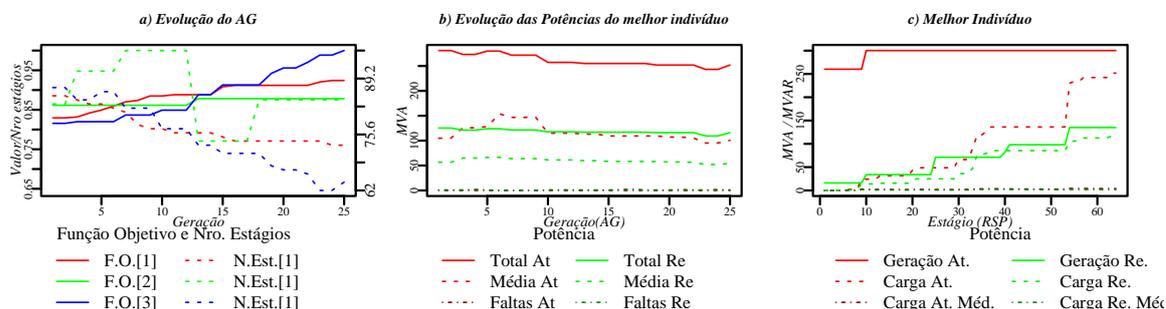


Figura 45: Sistema IEEE30, população=25, gerações=25, função=3, A=0, B=2 e C=1.

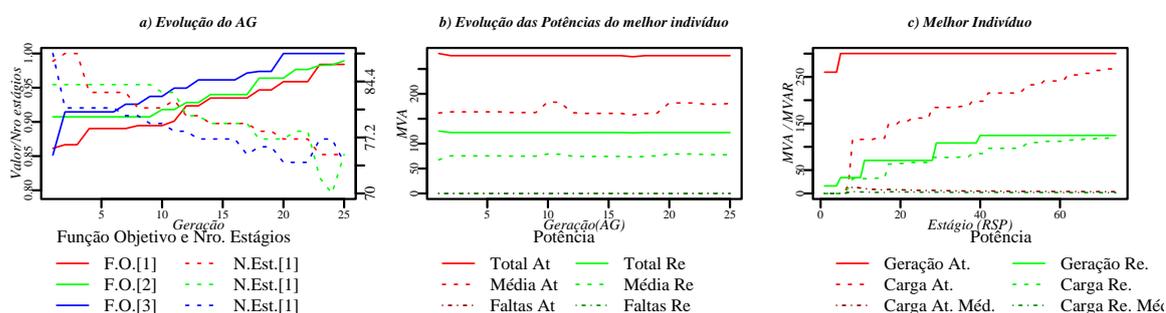


Figura 46: Sistema IEEE30, população=25, gerações=25, função=3, A=1, B=2 e C=1.

6.1.5 Função Objetivo 4

A última função objetivo avaliada é a função objetivo 4 (seção 4.2.7.1.5, equação (4.6)). A seguir, esta equação encontra-se repetida, para facilitar o acompanhamento dos resultados.

$$F_{O_5} = \frac{\sum_{b=1}^M [\overline{P_{AT}} \times (P_{AT_b} - A * P_{Fl_{At}}) + \overline{P_{Re}} \times (P_{Re_b} - B * P_{Fl_{Re}})]}{1 + C * N}$$

$$\overline{P_{AT}} = \frac{\sum_{n=1}^N P_{AT_n}}{N}$$

$$\overline{P_{Re}} = \frac{\sum_{n=1}^N P_{Re_n}}{N}$$

$$P_{Fl_{At}} = \begin{cases} 0, & \text{se } P_{Ge_{AT}} \geq P_{AT} \\ P_{AT} - P_{Ge_{AT}}, & \text{caso contrario} \end{cases}$$

$$P_{Fl_{Re}} = \begin{cases} 0, & \text{se } P_{Ge_{Re}} \geq P_{Re} \\ P_{Re} - P_{Ge_{Re}}, & \text{caso contrario} \end{cases}$$

Conforme descrito na seção 4.2.7.1.5, esta função vai agregar as demais já avaliadas a valorização da potência média atendida nos estágios, numa tentativa de obter soluções que atendam ao máximo de consumidores, representados por potências atendidas, no menor número de estágios.

6.1.5.1 Sistema IEEE 14 Barras

6.1.5.1.1 População: 25 - Gerações: 25

A figura 47 apresenta o resultado obtido com um dos experimentos deste sistema com esta função objetivo. A subfigura 47c mostra que há uma antecipação no atendimento às cargas, que passam a ser atendidas ainda antes do 10^o estágio.

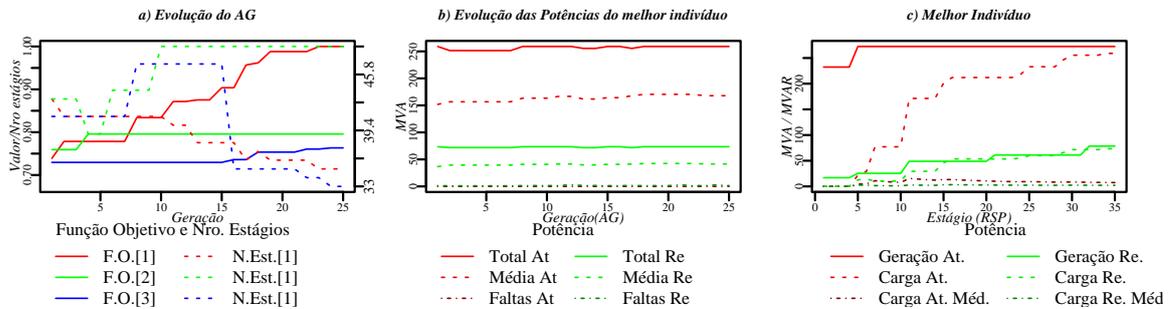


Figura 47: Sistema IEEE14, população=25, gerações=25, função=4, A=1, B=5 e C=1.

6.1.5.1.2 Demais populações

O aumento no número de gerações permitiu ao AG obter solução nas quais as cargas começam a ser atendidas em estágios ainda anteriores. Comparando, por exemplo, o resultado da figura 47 com o da figura 48, há uma alteração do estágio no qual se atendem os primeiros 100MW do 10^o para o 7^o estágio.

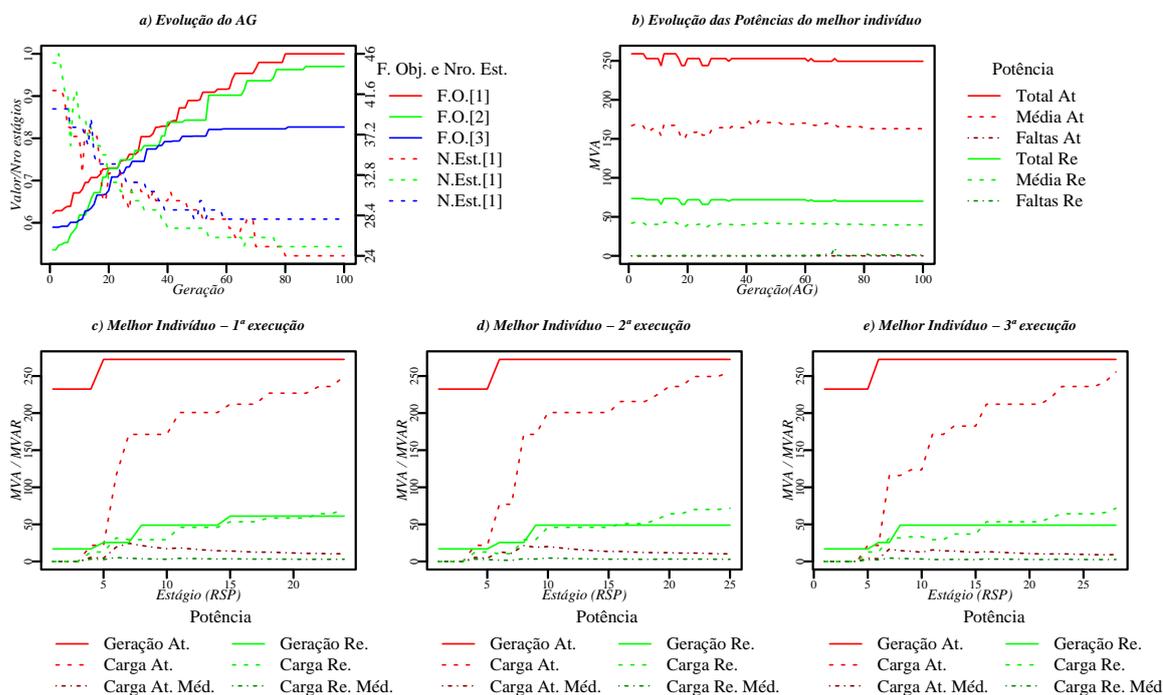


Figura 48: Resultados para o sistema IEEE14, com a população=50, 100 gerações, função objetivo=4, com parâmetros A=1, B=5 e C=1.

6.1.5.2 Sistema IEEE 30 Barras

Para o sistema IEEE 30, os resultados foram semelhantes aos do IEEE 14. As figuras 49 e 50 apresentam alguns dos resultados obtidos. Em ambos os experimentos, o AG manteve os valores de faltas de geração baixos e antecipou o atendimento às carga, prioridade da função objetivo.

O aumento do número de gerações e do tamanho da população trouxe uma redução no número total de estágios da solução vista na subfigura 50c, onde os estágios caíram de 81 (subfigura 49a) para 53.

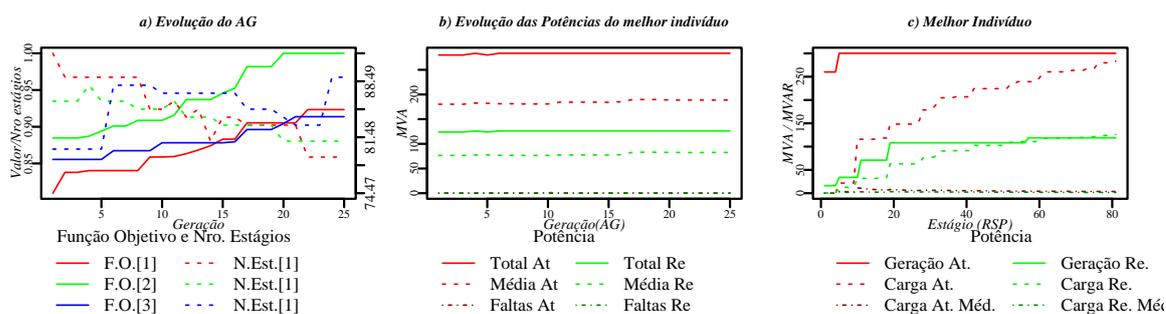


Figura 49: Sistema IEEE30, população=25, gerações=25, função=4, A=1, B=5 e C=1.

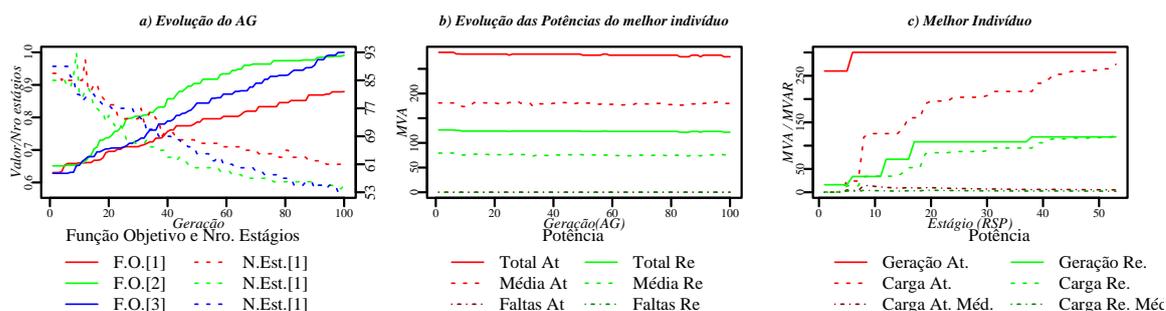


Figura 50: Sistema IEEE30, população=50, gerações=100, função=4, A=1, B=5 e C=1.

6.1.6 Sistema CEEE

O modelo do sistema CEEE é baseado em dados exportados a partir de um dos módulos do sistema de controle da Empresa. Estes dados podem sofrer alterações para modelar situações especiais no sistema. Após uma seqüência de experimentos com resultados insatisfatórios, o arquivo com a descrição do sistema foi avaliado, e verificou-se que algumas barras não apresentavam os dados considerados normais para seus tipos, como por exemplo, cargas negativas em barras PQ. Após a inclusão de lógica adequada para o tratamento destes casos, novos conjuntos de experimentos foram realizados e são descritos a seguir.

Os experimentos com o sistema CEEE não seguem a mesma metodologia empregada nos sistemas IEEE 14 e IEEE 30. Esta diferenciação existe devido ao tempo necessário para o AG resolver este sistema. Este tempo impediu a realização do mesmo conjunto de experimentos empregado nos demais sistemas.

Assim, todos os resultados dos experimentos realizados com o sistema CEEE são descritos a seguir.

6.1.6.1 Função Objetivo 1

A função objetivo 1 foi utilizada com uma população de 25 indivíduos, durante 25 gerações do AG. Os parâmetros empregados foram A=1, B=1 e C=0.

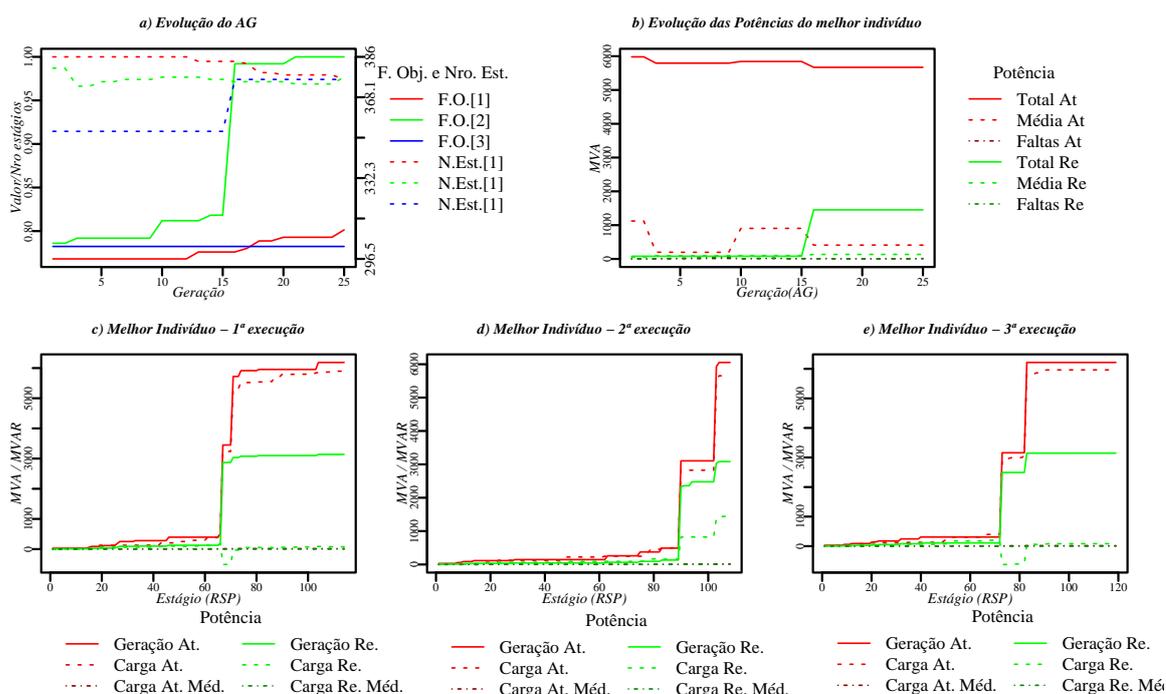


Figura 51: Resultados para o sistema CEEE, com a população=25, 25 gerações, função objetivo=1, com parâmetros A=1, B=1 e C=0.

O primeiro resultado que chama a atenção é a pequena variação do valor da função objetivo. Os valores normalizados variaram entre 0,75 e 1 (subfigura 6.1.6.1a). Da mesma forma, o número de estágios das soluções apresentou pouca variação.

Em torno da 15^a geração, observa-se o aumento do valor da função objetivo de um dos três indivíduos do experimento. Acompanhando na subfigura 6.1.6.1b, associa-se este aumento a um aumento na potência reativa atendida pela seqüência codificada neste genoma. Como esta função objetivo só observa o estado do sistema no último estágio de restauração, e como a potência reativa atendida pelo segundo genoma (subfigura 6.1.6.1d) foi bastante superior às dos demais genomas do experimento, este é o vencedor.

Comparando-se os valores das potências do melhor genoma (subfigura 6.1.6.1d) com o perfil das potências durante a evolução dos experimentos (subfigura b), percebe-se que os valores finais destes dois gráficos são diferentes. Como referem-se às mesmas variáveis, lidas nos mesmos estágios, seus valores deveriam ser iguais.

A diferença existente entre estes valores é um efeito da compensação introduzida para penalizar as não convergências (seção 4.3.1).

6.1.6.2 Função Objeto 2

A figura 52 representa os dados obtidos com o experimento realizado com a função objetivo 2. Os parâmetros utilizados foram $A=1$, $B=1$ e $C=1$. Utiliza-se uma população de 25 indivíduos, executando por 25 gerações.

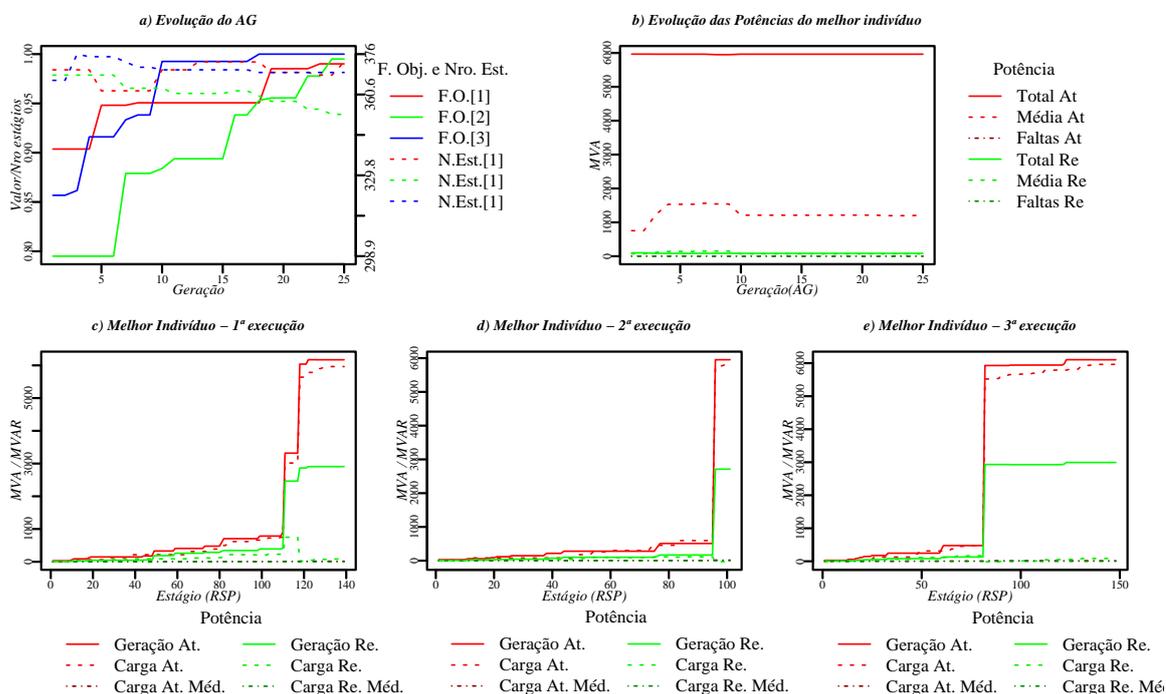


Figura 52: Resultados para o sistema CEEE, com a população=25, 25 gerações, função objetivo=2, com parâmetros $A=1$, $B=1$ e $C=1$.

Neste experimento, o melhor genoma apresenta um aumento da potência ativa atendida. Não há redução expressiva do número de estágios.

6.1.6.3 Função objetivo 3

A função objetivo 3 tem um experimento realizado, com população de 25 indivíduos e com um total de 25 gerações. Os parâmetros utilizados são $A=1$, $B=1$ e $C=1$. A figura 53 apresenta os resultados obtidos com este experimento.

Os resultados do experimento mostram um aumento considerável no valor retornado pela função objetivo. Há uma redução de algo em torno de 20 estágios.

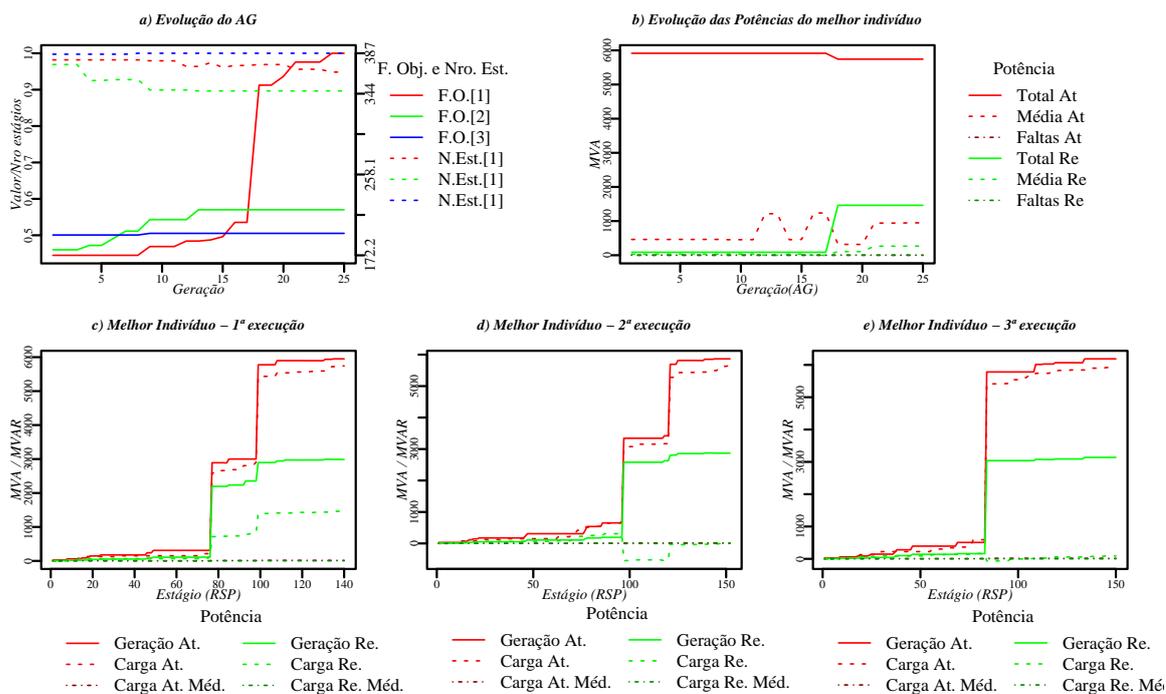


Figura 53: Resultados para o sistema CEEE, com a população=25, 25 gerações, função objetivo=3, com parâmetros A=1, B=1 e C=1.

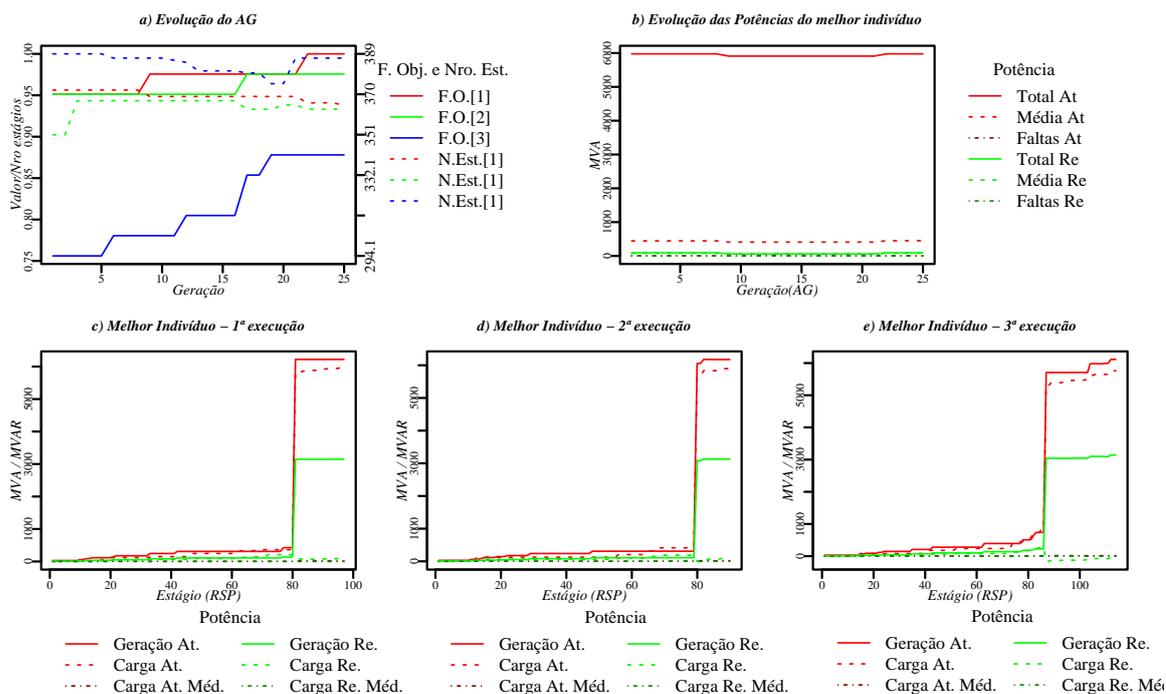


Figura 54: Resultados para o sistema CEEE, com a população=25, 25 gerações, função objetivo=4, com parâmetros A=1, B=5 e C=1.

6.1.6.4 Função Objetivo 4

Os resultados dos experimentos realizados com o sistema CEEE, empregando a função objetivo 4, podem ser visualizados na figura 54. Neste experimento, utiliza-se uma população de 25 indivíduos e com a execução por 25 gerações.

Aumentando-se a população para 50 indivíduos, e o número de gerações para 50, foram obtidos resultados para duas combinações de parâmetros diferentes da função objetivo 4. Os resultados estão ilustrados nas figuras 55 e 56.

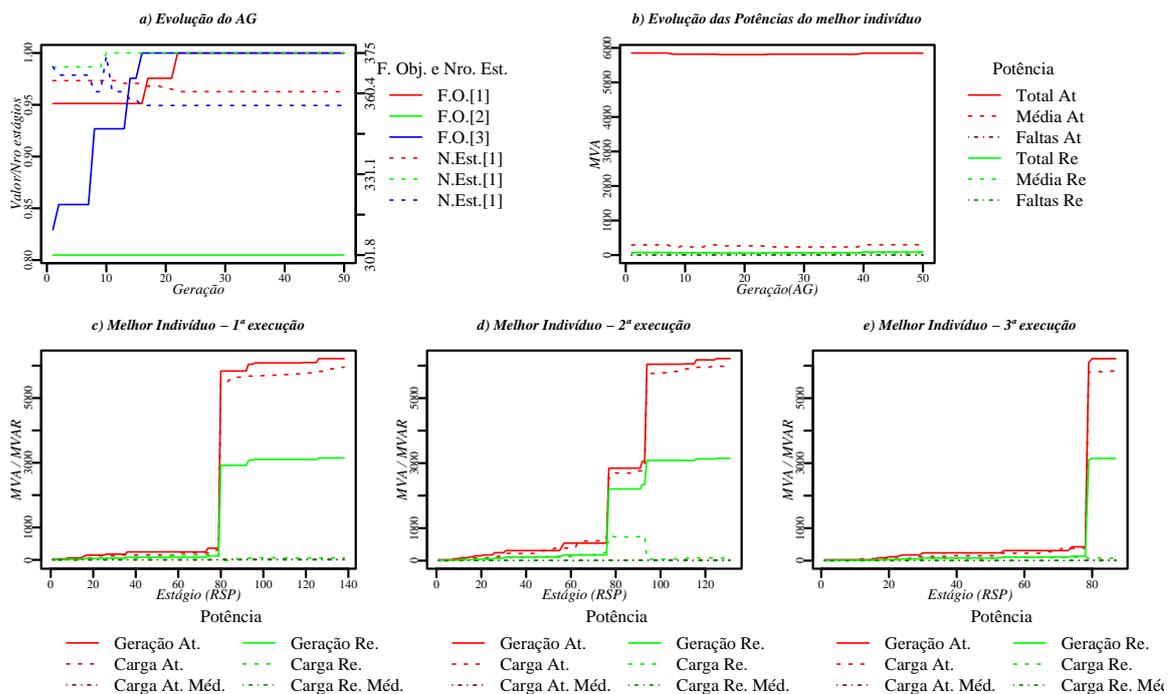


Figura 55: Resultados para o sistema CEEE, com a população=50, 50 gerações, função objetivo=4, com parâmetros A=1, B=2 e C=1.

6.2 Avaliação da Performance

O trabalho desenvolvido, já durante sua proposição, deixou claro que o item performance teria grande impacto. Devido a sua forte característica iterativa, executando um grande número de vezes o PFP, esperava-se que o mesmo não apresentasse uma performance apreciável.

Os tempos obtidos com a execução dos experimentos estão expostos na tabela 3. Os valores estão expressos em segundos, em função do sistema de potência e da Função Objetivo utilizada. Os tempos do sistema CEEE são estimados, baseados nos horários dos respectivos arquivos.

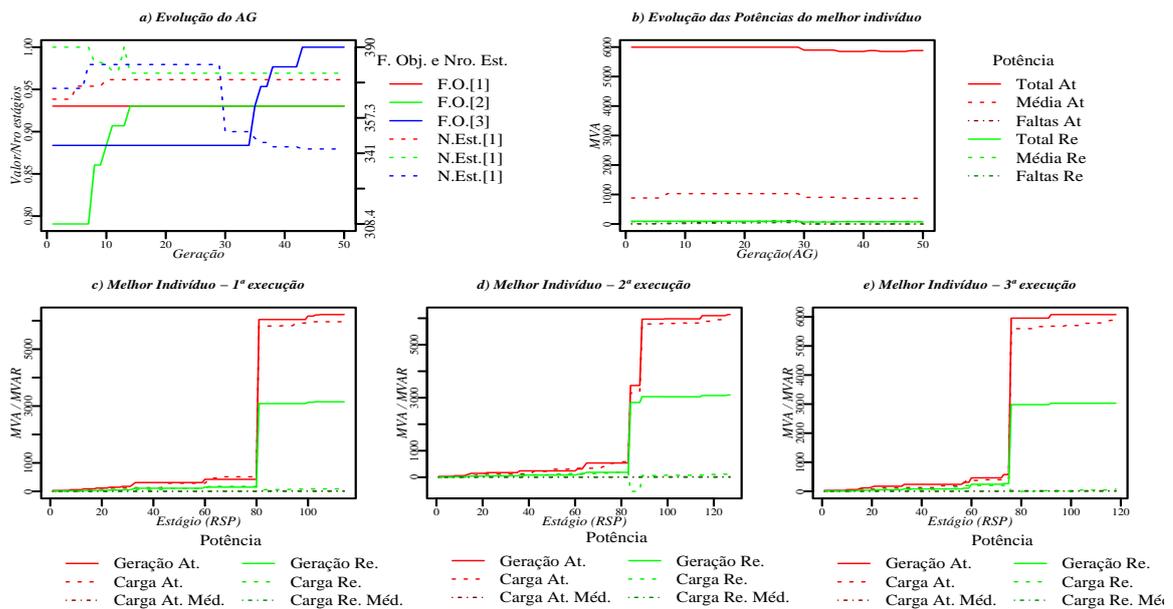


Figura 56: Resultados para o sistema CEEE, com a população=50, 50 gerações, função objetivo=4, com parâmetros A=1, B=5 e C=1.

6.3 Conclusões sobre os resultados dos experimentos

Os experimentos realizados com o módulo evolutivo mostraram que o método proposto é promissor. A utilização de diferentes funções objetivo demonstra que existe flexibilidade, por parte da nova representação desenvolvida, possibilitando que novas funções, mais adequadas, mais abrangentes, ou mesmo mais específicas, sejam empregadas com esta solução.

Tabela 3: Tempos médios de execução do programa, em segundos

Sistema de Potência	Função Objetivo	População x Gerações		
		25x25	50x50	50x100
IEEE14	0	271	408	696
	1	93	324	531
	2	108	445	861
	3	113	359	954
	4	152	378	802
IEEE30	0	889	1389	3083
	1	309	1163	2014
	2	357	1327	2956
	3	371	1151	3128
	4	455	1346	2415
CEEE	1	21000		
	2	29000		
	3	29000		
	4	29000	47000	

7 CONCLUSÃO

O propósito do trabalho, de implementar um sistema que apresente respostas rápidas aos operadores de sistemas de potência, quando da ocorrência de blecautes e outras falhas graves, foi atingido. O sistema não se encontra pronto para uso em sistemas reais, mas a implementação é possível, após incluídos elementos de interface homem-máquina, ainda incompletos. O caráter híbrido do sistema, unindo Algoritmo Genético (AG) e Métodos Analíticos(MA), permitiu a exploração de seqüências de operações *offline*.

O emprego de AGs para a geração de seqüências de operações, ou estratégias, para a restauração mostrou ser viável. A codificação desenvolvida, uma contribuição do trabalho, permitiu a solução do problema de sequenciamento de uma maneira que permitiu a observância às restrições do sistema. Entretanto, a codificação empregada ainda pode ser melhorada em muito para uma aplicação real.

A representação do genoma que foi desenvolvida, bem como dos operadores genéticos necessários para esta nova representação, foram de extrema importância para o desenvolvimento deste AG. O processo de inicialização baseado na pesquisa em um grafo, representando a topologia do sistema elétrico, restringiu o espaço de procura de soluções às soluções viáveis para cada caso. O emprego de aleatoriedade nesta etapa, na determinação das operações a serem realizadas em cada barra de potência e linha de transmissão, permitiu a obtenção de uma população rica em soluções, com seqüências que levavam, quase sempre, ao atendimento máximo das cargas presentes em cada sistema experimentado, à exceção do sistema CEEE.

O motivo mais provável para o comportamento diferenciado com este sistema, aliado a suas dimensões, deve repousar no arquivo que o descreve. Observou-se que o mesmo foi produzido por outro sistema computacional, um *estimador de estados*. Possivelmente este sistema, quando da exportação dos dados para o arquivo de saída, faça transformações baseadas em seu modelo interno, que não invalidam o modelo para seu próprio uso, mas que podem ter influenciado na ferramenta ora apresentada.

Como diretrizes para novos desenvolvimentos, ficam a sugestão da implementação dos pontos relevados na metodologia (alguns dos quais foram citados em 4.5), com a possível utilização de outros módulos analíticos, complementando os resultados do Fluxo de Potências e contemplando

os detalhes que a abstração empregada porventura desconsiderou. Outras possibilidades incluem a atenção maior à performance do programa, através de otimizações, computação distribuída ou outros processos, além de aprimoramentos na interface com o usuário, na representação, manipulação e controle mais apurado do processo.

ANEXO A - FORMATO PSAP

PSAP File Format

May 20, 1993

The PECO PSAP File Format is fully described in the `_PJM Power System Analysis Package Use's Guide_`, available from the Philadelphia Electric Company. The following is a rough description of the most important parts of the format.

A PSAP data file is divided into sections by code cards. The code is in the first three columns. There are something like 60 codes, of which only four are described in this document.

The 1 code indicates that the next card is the case title. Only one title is allowed per case.

The 4 card indicates that line data follows. The line data ends with a 9999 card.

The 5 card indicates that bus data follows. The bus data ends with a 9999 card.

The 15 card indicates that area interchange data follows. The data ends with a 9999 card.

Line Data Card (Code 4 cards)

=====

Cols	Data
1-4	From bus number

6 Change code (blank in 4 section)

7 'C' if second card present for same line. Used for transformers.

9-12 To bus number

14 Circuit number (blank in 4 section)

16 'T' or 'F' - Load flow area of bus at this end of line gets losses.

18-23 Line resistance in percent of base. (NOT per unit.)
(percent = 100 x per unit) Two default decimal places.

24-29 Line reactance, in percent. Two default decimal places.

30-35 Line charging MVAR (total). Three default decimal places.

36-40 Transformer tap (per unit turns ratio). Three default decimal
places, 1000 = 1.000.

41-45 Min tap, for OLTC. Three default decimal places.

46-50 Max tap, for OLTC. Three default decimal places.

51-55 Phase shift angle, for OL phase shifter. Two default decimal places.

56-60 Remote voltage control bus number. Negative if lower tap increases
voltage of this bus.

61-64 Normal MVA rating

65-68 Emergency MVA rating

69-72 MVA Base. Default value 100 MVA if blank.

Second Line Card (follows 'C' in first card)

=====

1-17 Same as first card, except no 'C'. Can be left blank.

35-40 Desired MVAR flow or Min voltage setpoint for OLTC.

41-45 Min phase shifter degrees. Two default decimal places.

46-50 Max phase shifter degrees. Two default decimal places.

51-55 Desired MW flow for phase shifter.

57-60 Controlled line from bus.

62-65 Controlled line to bus.

67-70 Available taps (number of taps)

71-75 Maximum voltage setpoint. Three default decimal places.

Bus Cards (Code 5 cards)

=====

1-4 Bus number

6 Change code (blank in 5 section)

7 Continue code (blank in 5 section)
 8 Regulated bus code:
 Blank - load (PQ) bus
 1 - gen (PV) bus
 2 - swing (V-Theta) bus
 10-21 Name
 23-26 Bus voltage (control setpoint or solved value).
 Three default decimal places.
 27-30 Bus angle
 31-35 Generation MW
 36-40 Generation MVAR (from solution)
 41-45 Generation MVAR low limit
 46-50 Generation MVAR high limit
 51-55 Bus at which generation controls voltage
 56-60 Load MW
 61-65 Load MVAR
 66-70 Shunt MVAR. Reactors are minus.
 71-72 Load flow area. (Used for area interchange and losses).

Area Interchange Cards (Code 15 cards)

=====

3-4 Load flow area number
 5-8 Swing bus for area interchange. Adjusts generation at this bus
 to meet area interchange requirement.
 9-14 Area exports, MW. (+ = out of area)
 15-19 Area Interchange tolerance, MW
 20-55 Area name
 56-60 Area load (usually left blank)
 61-65 Area losses (usually left blank)

ANEXO B - FORMATO CDF

Partial Description of the IEEE Common Data Format for the Exchange of Solved Load Flow Data

The complete description can be found in the paper "Common Data Format for the Exchange of Solved Load Flow Data", Working Group on a Common Format for the Exchange of Solved Load Flow Data, _IEEE Transactions on Power Apparatus and Systems_, Vol. PAS-92, No. 6, November/December 1973, pp. 1916-1925.

The data file has lines of up to 128 characters. The lines are grouped into sections with section headers. Data items are entered in specific columns. No blank items are allowed, enter zeros instead. Floating point items should have explicit decimal point. No implicit decimal points are used.

Data type codes: A - Alphanumeric (no special characters)
 I - Integer
 F - Floating point
 * - Mandatory item

Title Data
 =====

First card in file.

Columns 2- 9 Date, in format DD/MM/YY with leading zeros. If no date provided, use 0b/0b/0b where b is blank.

Columns 11-30 Originator's name (A)

Columns 32-37 MVA Base (F*)

Columns 39-42 Year (I)

Column 44 Season (S - Summer, W - Winter)

Column 46-73 Case identification (A)

Bus Data *

=====

Section start card *:

Columns 1-16 BUS DATA FOLLOWS (not clear that any more than BUS in
1-3 is significant) *

Columns ?- ? NNNNN ITEMS (column not clear, I would not count on this)

Bus data cards *:

Columns 1- 4 Bus number (I) *

Columns 7-17 Name (A) (left justify) *

Columns 19-20 Load flow area number (I) Don't use zero! *

Columns 21-23 Loss zone number (I)

Columns 25-26 Type (I) *

0 - Unregulated (load, PQ)

1 - Hold MVAR generation within voltage limits, (PQ)

2 - Hold voltage within VAR limits (gen, PV)

3 - Hold voltage and angle (swing, V-Theta) (must always
have one)

Columns 28-33 Final voltage, p.u. (F) *

Columns 34-40 Final angle, degrees (F) *

Columns 41-49 Load MW (F) *

Columns 50-59 Load MVAR (F) *

Columns 60-67 Generation MW (F) *

Columns 68-75 Generation MVAR (F) *

Columns 77-83 Base KV (F)
 Columns 85-90 Desired volts (pu) (F) (This is desired remote voltage if
 this bus is controlling another bus.
 Columns 91-98 Maximum MVAR or voltage limit (F)
 Columns 99-106 Minimum MVAR or voltage limit (F)
 Columns 107-114 Shunt conductance G (per unit) (F) *
 Columns 115-122 Shunt susceptance B (per unit) (F) *
 Columns 124-127 Remote controlled bus number

Section end card:

Columns 1- 4 -999

Branch Data *

=====

Section start card *:

Columns 1-16 BRANCH DATA FOLLOWS (not clear that any more than BRANCH
 is significant) *

Columns 40?- ? NNNNN ITEMS (column not clear, I would not count on this)

Branch data cards *:

Columns 1- 4 Tap bus number (I) *
 For transformers or phase shifters, the side of the model
 the non-unity tap is on
 Columns 6- 9 Z bus number (I) *
 For transformers and phase shifters, the side of the model
 the device impedance is on.
 Columns 11-12 Load flow area (I)
 Columns 13-14 Loss zone (I)
 Column 17 Circuit (I) * (Use 1 for single lines)
 Column 19 Type (I) *

0 - Transmission line
 1 - Fixed tap
 2 - Variable tap for voltage control (TCUL, LTC)
 3 - Variable tap (turns ratio) for MVAR control
 4 - Variable phase angle for MW control (phase shifter)

Columns 20-29 Branch resistance R, per unit (F) *
 Columns 30-40 Branch reactance X, per unit (F) * No zero impedance lines
 Columns 41-50 Line charging B, per unit (F) * (total line charging, +B)
 Columns 51-55 Line MVA rating No 1 (I) Left justify!
 Columns 57-61 Line MVA rating No 2 (I) Left justify!
 Columns 63-67 Line MVA rating No 3 (I) Left justify!
 Columns 69-72 Control bus number
 Column 74 Side (I)

0 - Controlled bus is one of the terminals
 1 - Controlled bus is near the tap side
 2 - Controlled bus is near the impedance side (Z bus)

Columns 77-82 Transformer final turns ratio (F)
 Columns 84-90 Transformer (phase shifter) final angle (F)
 Columns 91-97 Minimum tap or phase shift (F)
 Columns 98-104 Maximum tap or phase shift (F)
 Columns 106-111 Step size (F)
 Columns 113-119 Minimum voltage, MVAR or MW limit (F)
 Columns 120-126 Maximum voltage, MVAR or MW limit (F)

Section end card:

Columns 1- 4 -999

Loss Zone Data

=====

Section start card

Columns 1-16 LOSS ZONES FOLLOWS (not clear that any more than LOSS
 is significant)

Columns 40?- ? NNNNN ITEMS (column not clear, I would not count on this)

Loss Zone Cards:

Columns 1- 3 Loss zone number (I)

Columns 5-16 Loss zone name (A)

Section end card:

Columns 1- 3 -99

Interchange Data *

=====

Section start card

Columns 1-16 INTERCHANGE DATA FOLLOWS (not clear that any more than
first word is significant).

Columns 40?- ? NNNNN ITEMS (column not clear, I would not count on this)

Interchange Data Cards *:

Columns 1- 2 Area number (I) no zeros! *

Columns 4- 7 Interchange slack bus number (I) *

Columns 9-20 Alternate swing bus name (A)

Columns 21-28 Area interchange export, MW (F) (+ = out) *

Columns 30-35 Area interchange tolerance, MW (F) *

Columns 38-43 Area code (abbreviated name) (A) *

Columns 46-75 Area name (A)

Section end card:

Columns 1- 2 -9

Tie Line Data

=====

Section start card

Columns 1-16 TIE LINES FOLLOW (not clear that any more than TIE
is significant)

Columns 40?- ? NNNNN ITEMS (column not clear, I would not count on this)

Tie Line Cards:

Columns 1- 4 Metered bus number (I)
Columns 7-8 Metered area number (I)
Columns 11-14 Non-metered bus number (I)
Columns 17-18 Non-metered area number (I)
Column 21 Circuit number

Section end card:

Columns 1- 3 -999

ANEXO C – FORMATO ANAREDE

Dados de Barras - Código de execução DBAR

Campo	Colunas	Descrição	Default
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.	A
Tipo	08-08	0 - barra de carga (PQ - Injeções de potências ativa e reativa fixas). 1 - barra de tensão regulada (PV - Injeção de potência ativa e Magnitude de tensão fixas). 2 - barra de referência (V., Magnitude da tensão e Ângulo de fase fixo). 3 - barra de carga com limite de tensão (PQ - Injeções de potências ativa e reativa fixas enquanto a magnitude de tensão permanecer entre os valores limites).	0
Grupo de Base de Tensão	09-09	Dígito (0 a 9) ou caracter (A a Z). Os valores associados aos Grupos Base de Tensão são definidos no código de execução DGBT. Os grupos que não forem definidos terão valor igual a 1 kV.	0
Nome	10-21	Identificação alfanumérica da barra.	
Grupo de Limite de Tensão	22-22	Dígito (0 a 9) ou caracter (A a Z). Os valores associados aos Grupos de Limite de Tensão são definidos no Código de Execução DGLT. Os grupos que não forem definidos terão valores limites de tensão, mínimo e máximo, iguais a 0.8 e 1.2 pu, respectivamente.	0

Tensão	23-26	Valor inicial da magnitude da 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 da tensão a ser mantido constante. Ponto decimal implícito entre as colunas 23 e 24.	1.0
Ângulo	27-30	Ângulo de fase inicial da tensão da barra, em graus.	0.0
Geração Ativa	31-35	Valor de geração de potência ativa na barra, em MW. Este campo define o ponto base de operação sobre o qual as ações de controle são executadas de modo a manter o intercâmbio de potência ativa programado entre áreas. Os erros de intercâmbio de potência ativa entre áreas são distribuídos entre os geradores das áreas, com base neste valor e de acordo com a participação de cada gerador.	0.0
Geração Reativa	36-40	Valor de geração de potência reativa na barra, em Mvar. Para barra de carga este valor é fixo. Para barra de carga com limite de tensão este valor é mantido constante, enquanto a magnitude da tensão permanecer entre os limites especificados. Para barras de tensão regulada e de referência com limites de geração de potência reativa especificados, este campo pode ser deixado em branco.	0.0
Geração Reativa Mínima	41-45	Valor do limite mínimo de geração de potência reativa na barra, em Mvar.	¹
Geração Reativa Máxima	46-50	Valor do limite máximo de geração de potência reativa na barra, em Mvar.	¹

¹Se a barra for do tipo referência e ambos os campos forem deixados em branco os limites mínimo e máximo de geração de potência reativa serão abertos, isto é, iguais a -9999.0 e 99999.0 Mvar respectivamente. Em qualquer outro caso valor assumido será 0.0 Mvar.

Barra Controlada	51-55	Para barras de tensão regulada e de referência, com limites de potência reativa especificados, este campo destina-se ao número da barra cuja magnitude da tensão será controlada. O valor da magnitude da tensão a ser mantido é obtido no campo Tensão do registro relativo à barra.	A mesma barra
Carga Ativa	56-60	Valor da carga ativa da barra, em MW. No caso da carga variar com a magnitude da tensão da barra, entre neste campo o valor da carga para a tensão especificada no campo Tensão Para Definição de Carga.	0.0
Carga Reativa	61-65	Valor da carga reativa da barra, em Mvar. No caso da carga variar com a magnitude da tensão da barra, entre neste campo o valor da carga para a tensão especificada no campo Tensão Para Definição de Carga.	0.0
Capacitor Reator	66-70	Valor total da potência reativa injetada na barra, em Mvar, por bancos de capacitores/reatores. O valor a ser preenchido neste campo refere-se a potência reativa injetada na tensão nominal (1.0 p.u.). Este valor deve ser positivo para capacitores e negativo para reatores.	0.0
Área	71-72	Número da área à qual pertence a barra.	0
Tensão Para Definição de Carga	73-76	Entre neste campo com o valor em p.u. da tensão para a qual foi medido o valor das parcelas ativa e reativa da carga definidos nos campos Carga Ativa e Carga Reativa, respectivamente. Ponto decimal implícito entre as colunas 73 e 74.	Tensão

Dados de Linhas - Código de execução DLIN

Campo	Colunas	Descrição	Default
Da Barra	01-04	Número da barra de uma das extremidades do circuito como definido no campo Número do Código de Execução DBAR.	
Operação	06-06	A ou 0 - adição de dados de circuito. E ou 1 - eliminação de dados de circuito. M ou 2 - modificação de dados de circuito.	A
Para Barra	09-12	Número da barra da outra extremidade do circuito como definido no campo Número do Código de Execução DBAR.	
Circuito	13-14	Número de identificação da circuito CA em paralelo.	²
Estado	15-15	L se o circuito estiver em operação (ligado). D se o circuito estiver fora de operação (desligado).	L
Proprietário ³	16-16	F se o circuito pertencer a área da barra definida no campo Da Barra. T se o circuito pertencer a área da barra definida no campo Para Barra.	F
Resistência	18-23	Valor da resistência do circuito, em %. Para transformadores este valor corresponde ao valor da resistência para o tap nominal. Ponto decimal implícito entre as colunas 21 e 22.	0.0
Reatância	24-29	Valor da reatância do circuito, em %. Para transformadores este valor corresponde ao valor da reatância para o tap nominal. Ponto decimal implícito entre as colunas 27 e 28.	
Susceptância	30-35	Valor total da susceptância shunt do circuito, em Mvar. Ponto decimal implícito entre as colunas 32 e 33.	0.0

²No caso de adição de dado de circuito o valor default para o número do circuito em paralelo consiste do primeiro número disponível a partir do maior número do circuito em paralelo cujo dado já existe. No caso de alteração ou eliminação o valor default é igual ao menor número do circuito em paralelo.

³As perdas de potência ativa nos circuitos são contabilizadas para a área a qual pertence o circuito (definido pelo campo proprietário) e, para efeito de intercâmbio, os fluxos são calculados na extremidade conectada à barra da área não proprietária do circuito.

Tap	36-40	Valor do tap referido à barra definida no campo Da Barra, em p.u., para os transformadores de tap fixo ou, uma estimativa deste valor para os transformadores com variação automática de tap (LTC) ⁴ . Ponto decimal implícito entre as colunas 37 e 38.
-----	-------	---

⁴Os transformadores tipo LTC são identificados pelo preenchimento dos campos Tap Mínimo e Tap Máximo. Nesse caso, se o valor inicial do tap não for especificado, o valor 1.0 p.u. é considerado. Se o valor inicial do tap estiver fora dos limites especificados, este valor é considerado igual ao valor do limite violado.

ANEXO D – LICENÇA GALIB

Licensing Agreement for GALib

10dec99 mbw

GALib License Agreement

As of 1 Jan 2000, GALib is free for both commercial and non-commercial uses under the terms of a BSD-style open source license as follows:

Copyright (c) 1995-1996 Massachusetts Institute of Technology (MIT)

Copyright (c) 1996-2000 Matthew Wall (the Author)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Massachusetts Institute of Technology (MIT) nor the names of contributors may be used to endorse or

promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ‘‘AS IS’’ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The GNU portions of the GALib distribution

The portions of GALib (see below) that contain code from the GNU g++ library are covered under the terms of the GNU Public License. As such they are freely available and do not fall under the terms of the GALib licensing conditions above.

The portions of GALib that are based upon GNU code are all in the ‘gnu’ directory in the examples directory (in GALib release 2.3.2 and later).

ANEXO E – GRÁFICOS DOS RESULTADOS - COMPLETOS

E.1 Sistema IEEE 14

E.1.1 Função 0

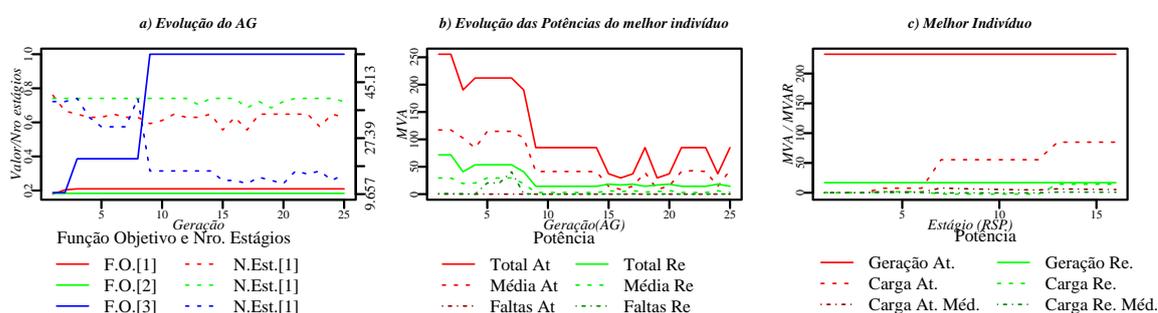


Figura 57: Sistema IEEE14, população=25, gerações=25, função=0, A=1 e B=1.

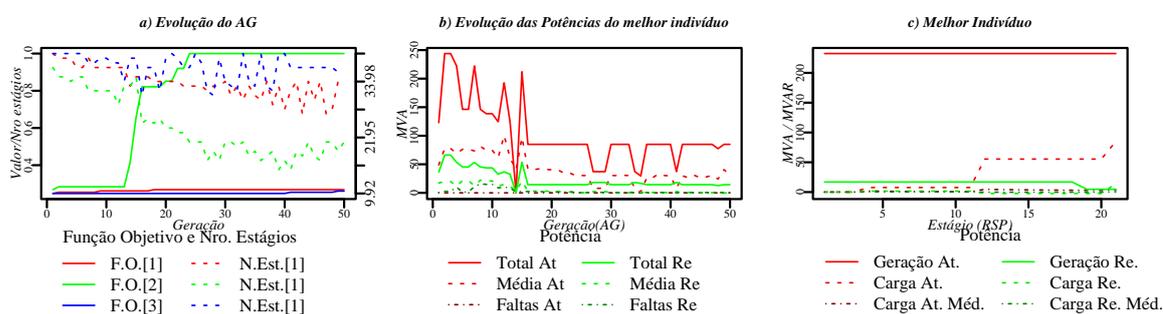


Figura 58: Sistema IEEE14, população=50, gerações=50, função=0, A=1 e B=1.

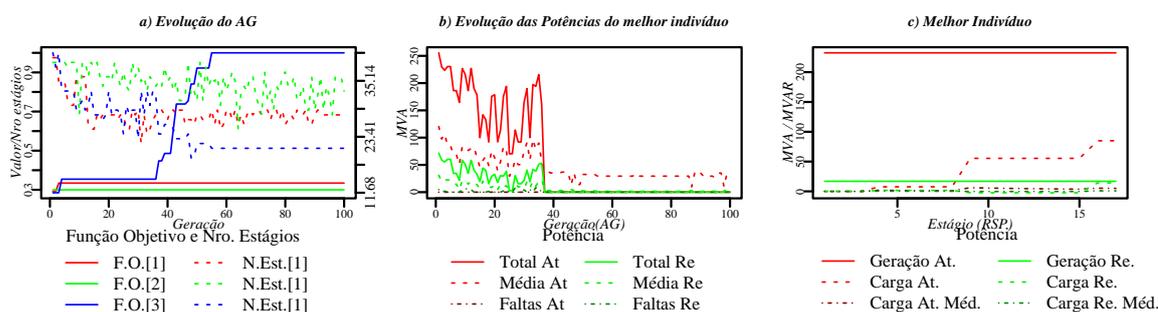


Figura 59: Sistema IEEE14, população=50, gerações=100, função=0, A=1 e B=1.

E.1.2 Função 1

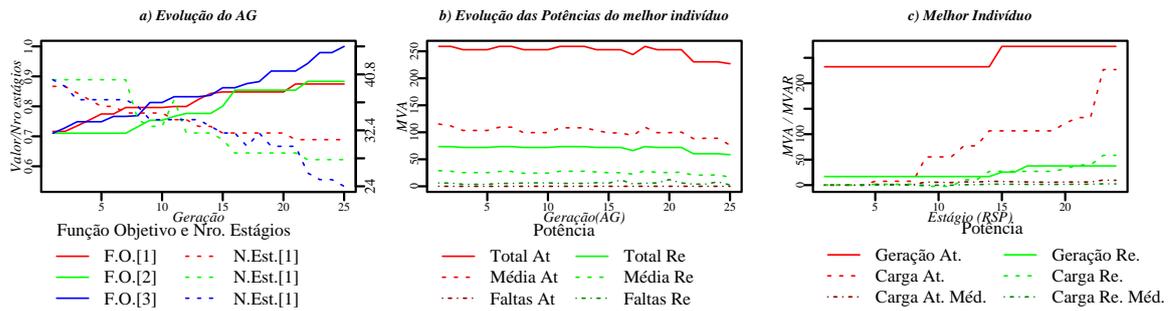


Figura 60: Sistema IEEE14, população=25, gerações=25, função=1, A=1, B=1 e C=0.

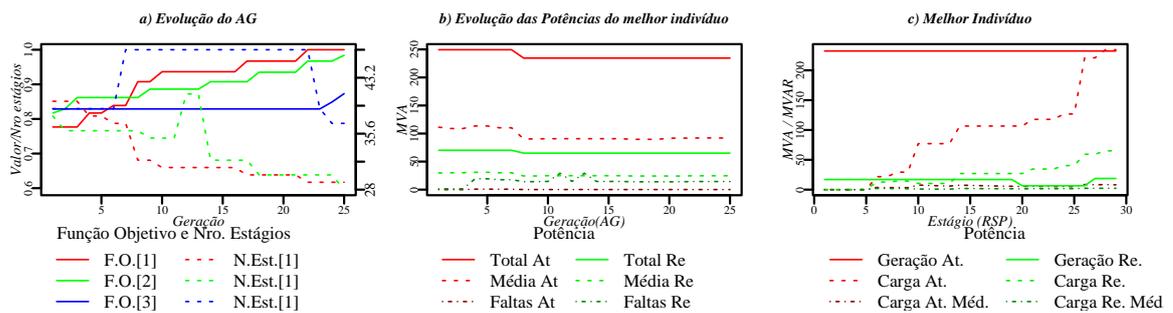


Figura 61: Sistema IEEE14, população=25, gerações=25, função=1, A=1, B=2 e C=0.

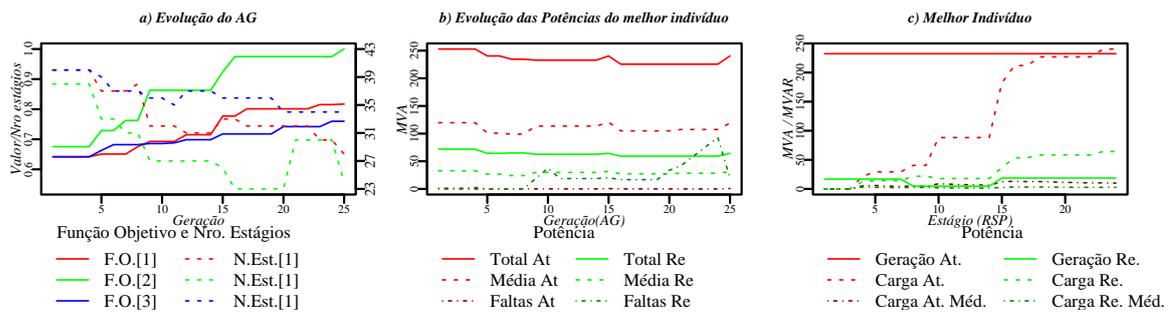


Figura 62: Sistema IEEE14, população=25, gerações=25, função=1, A=1, B=5 e C=0.

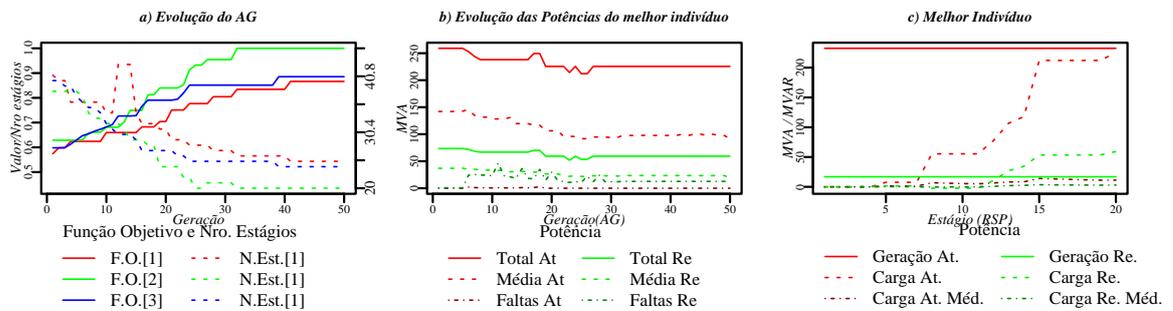


Figura 63: Sistema IEEE14, população=50, gerações=50, função=1, A=1, B=1 e C=0.

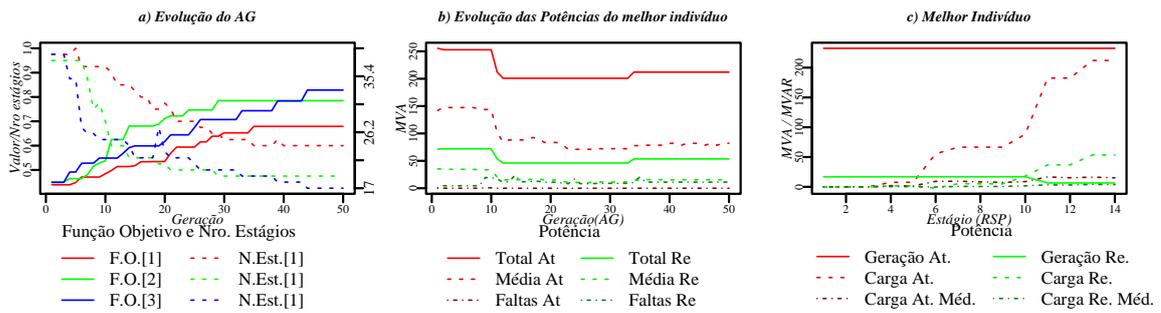


Figura 64: Sistema IEEE14, população=50, gerações=50, função=1, A=1, B=2 e C=0.

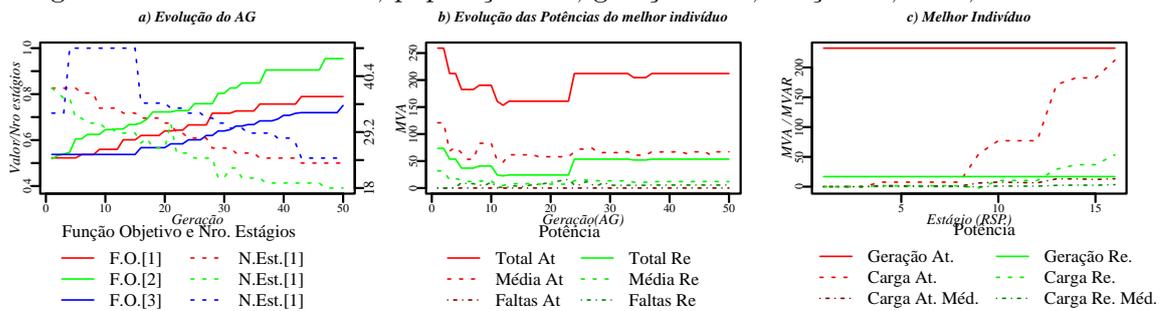


Figura 65: Sistema IEEE14, população=50, gerações=50, função=1, A=1, B=5 e C=0.

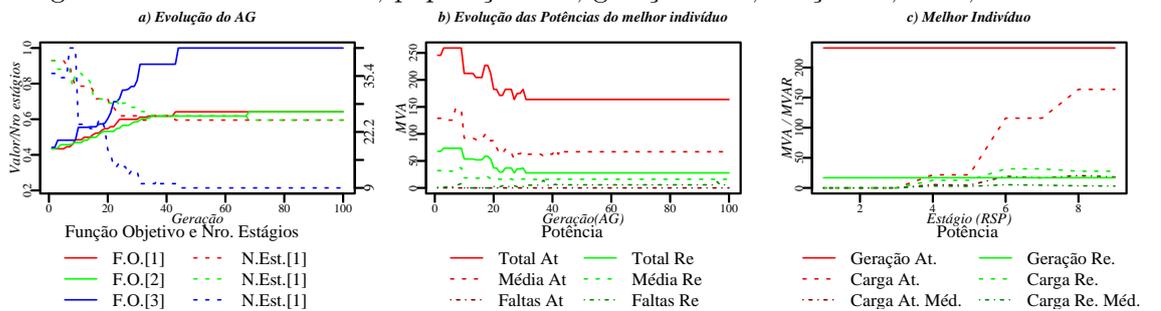


Figura 66: Sistema IEEE14, população=50, gerações=100, função=1, A=1, B=1 e C=0.

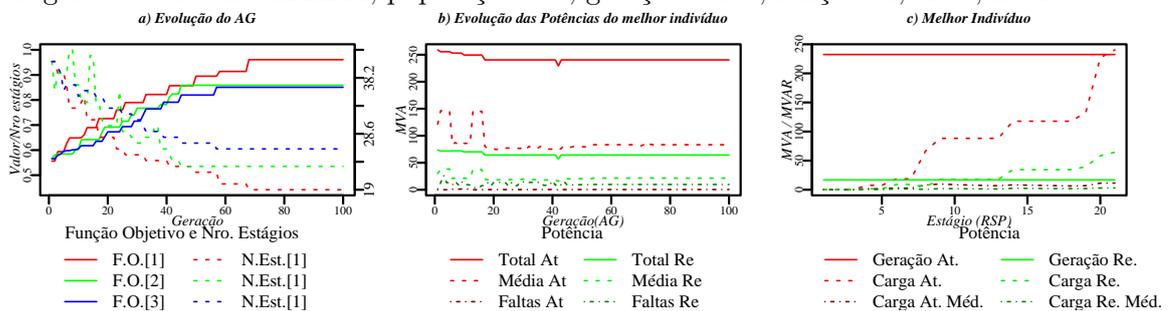


Figura 67: Sistema IEEE14, população=50, gerações=100, função=1, A=1, B=2 e C=0.

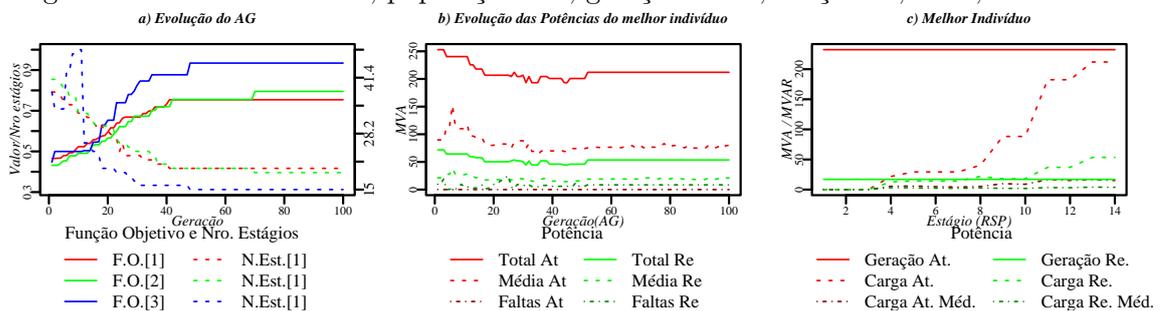


Figura 68: Sistema IEEE14, população=50, gerações=100, função=1, A=1, B=5 e C=0.

E.1.3 Função 2

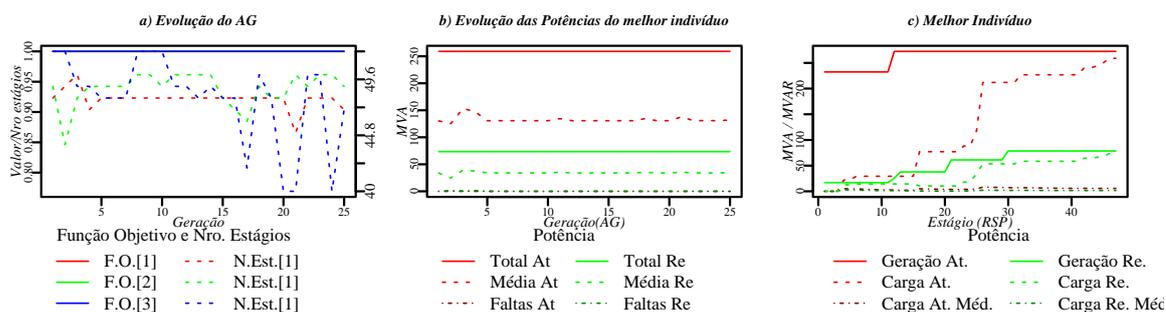


Figura 69: Sistema IEEE14, população=25, gerações=25, função=2, A=1, B=1 e C=0.

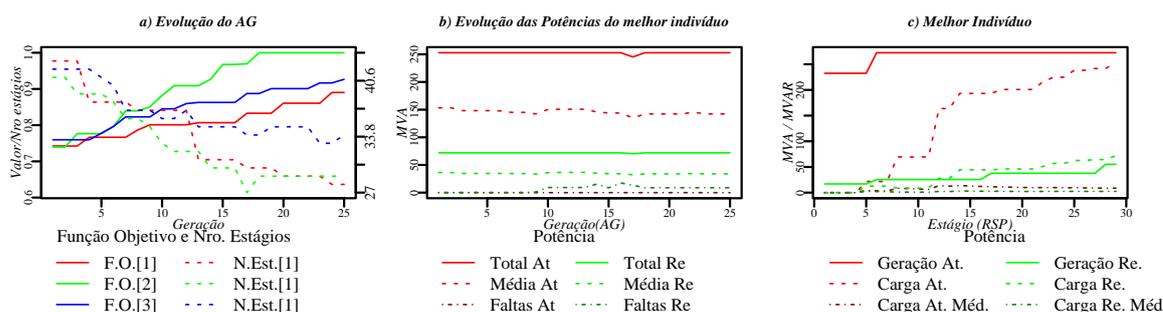


Figura 70: Sistema IEEE14, população=25, gerações=25, função=2, A=1, B=1 e C=1.

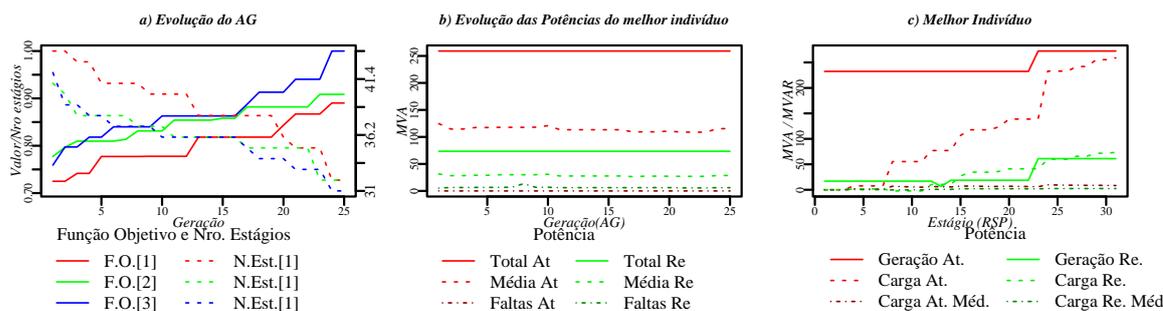


Figura 71: Sistema IEEE14, população=25, gerações=25, função=2, A=1, B=1 e C=2.

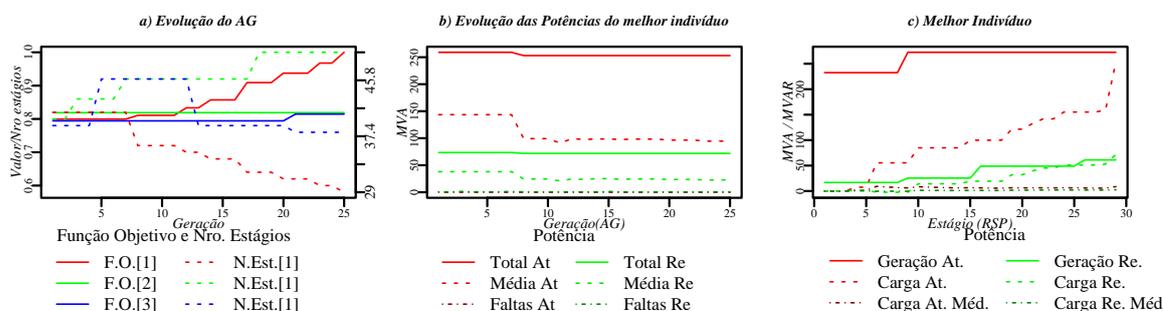


Figura 72: Sistema IEEE14, população=25, gerações=25, função=2, A=1, B=2 e C=1.

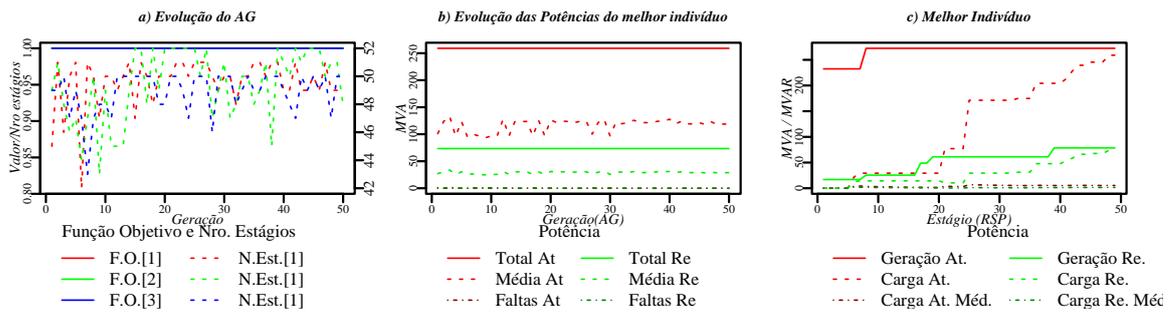


Figura 73: Sistema IEEE14, população=50, gerações=50, função=2, A=1, B=1 e C=0.

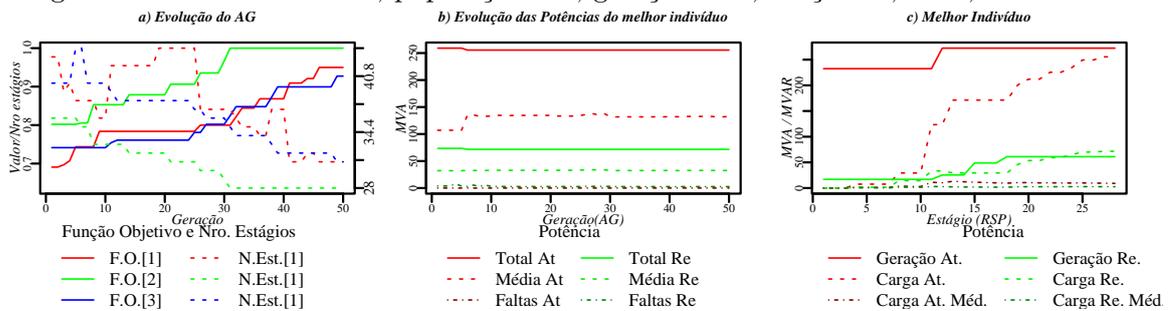


Figura 74: Sistema IEEE14, população=50, gerações=50, função=2, A=1, B=1 e C=1.

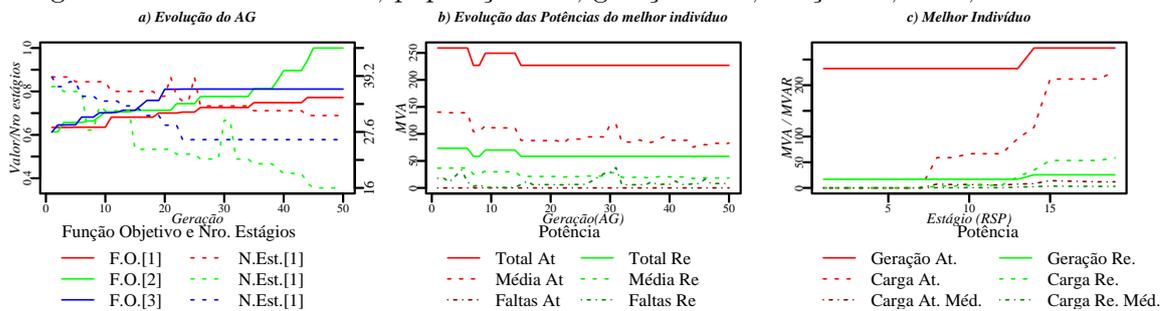


Figura 75: Sistema IEEE14, população=50, gerações=50, função=2, A=1, B=1 e C=2.

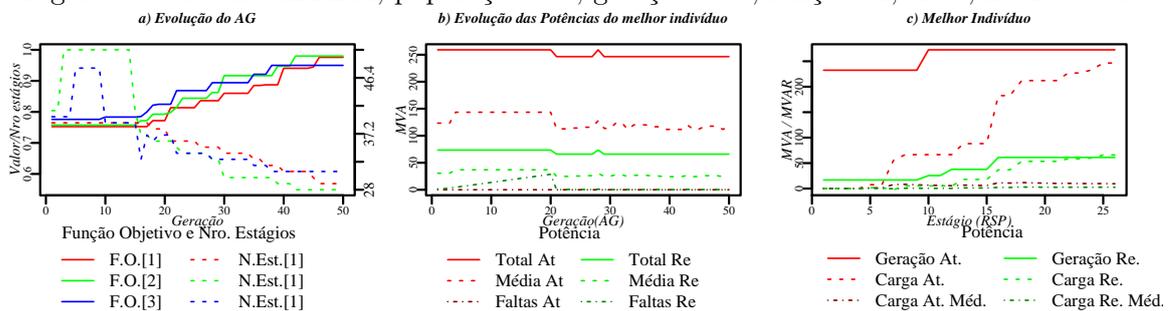


Figura 76: Sistema IEEE14, população=50, gerações=50, função=2, A=1, B=2 e C=1.

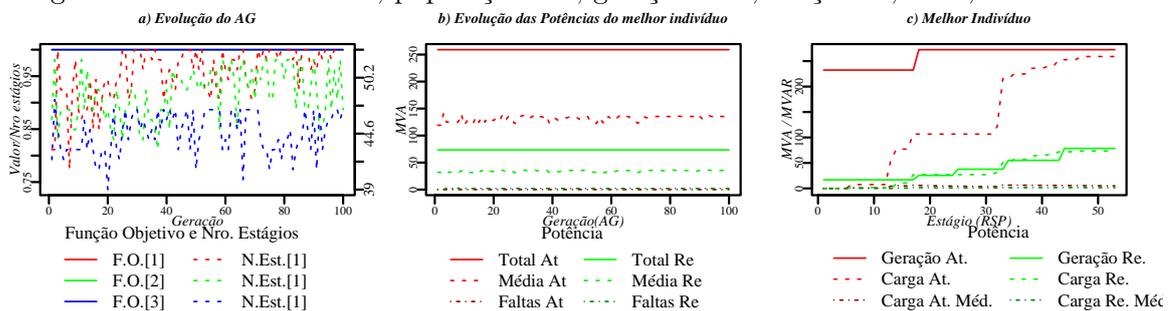


Figura 77: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=1 e C=0.

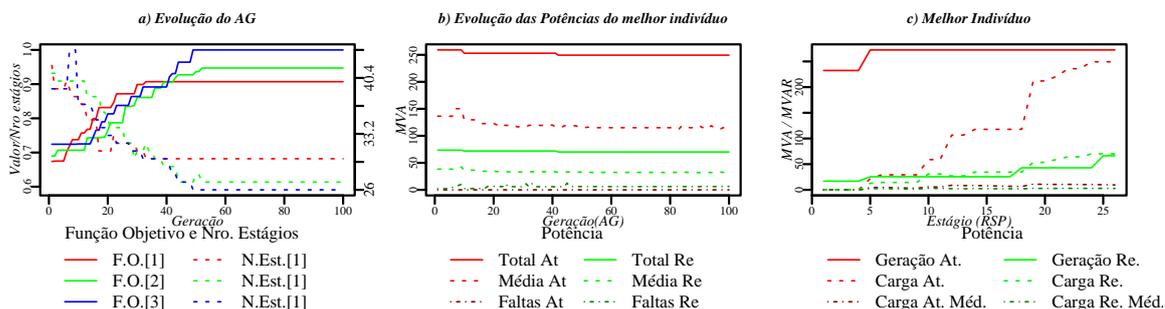


Figura 78: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=1 e C=1.

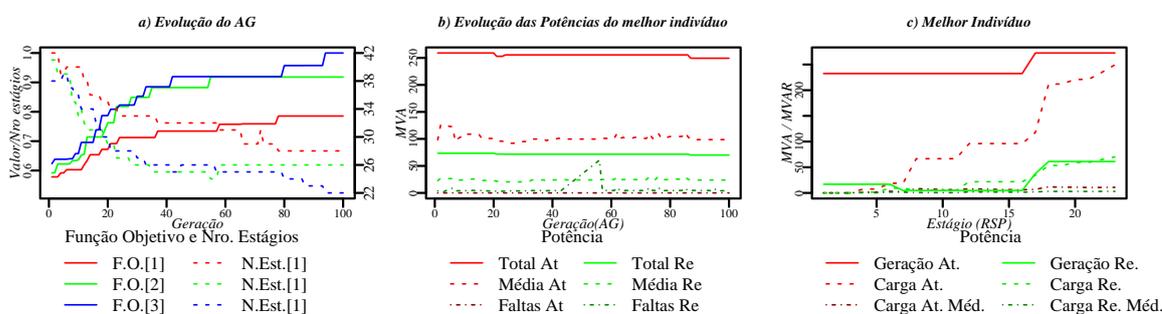


Figura 79: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=1 e C=2.

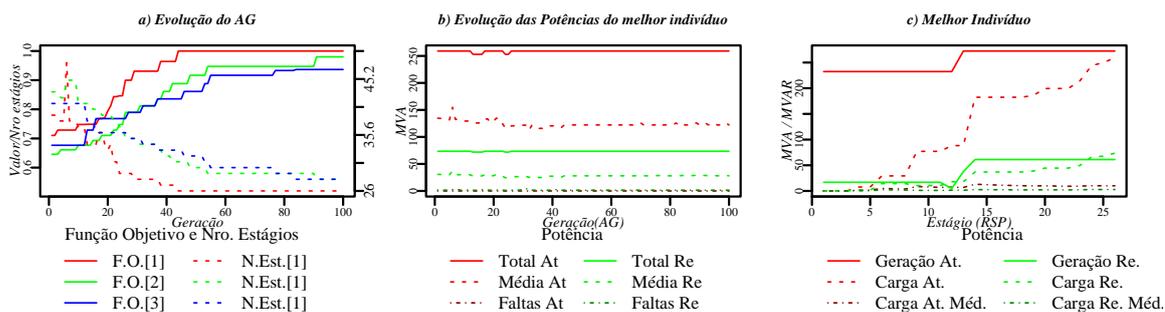


Figura 80: Sistema IEEE14, população=50, gerações=100, função=2, A=1, B=2 e C=1.

E.1.4 Função 3

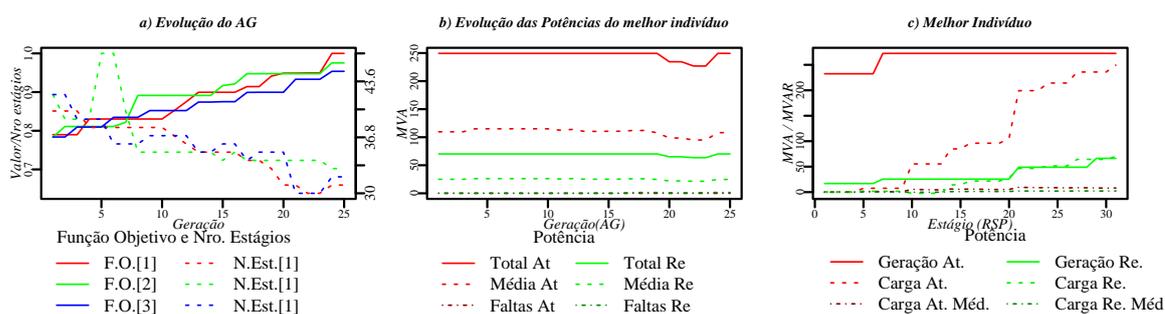


Figura 81: Sistema IEEE14, população=25, gerações=25, função=3, A=0, B=1 C=1.

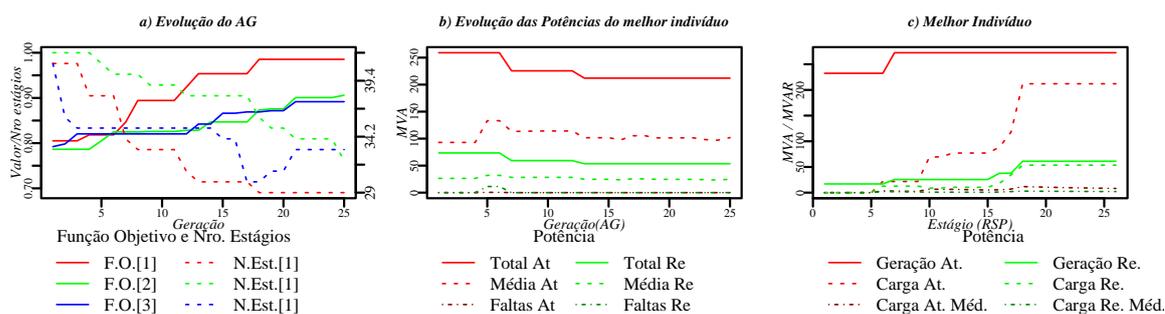


Figura 82: Sistema IEEE14, população=25, gerações=25, função=3, A=0, B=2 e C=1.

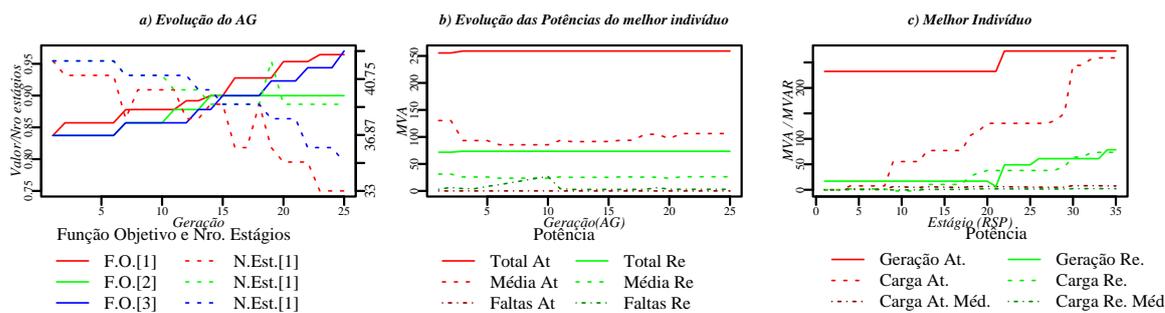


Figura 83: Sistema IEEE14, população=25, gerações=25, função=3, A=1, B=1 e C=1.

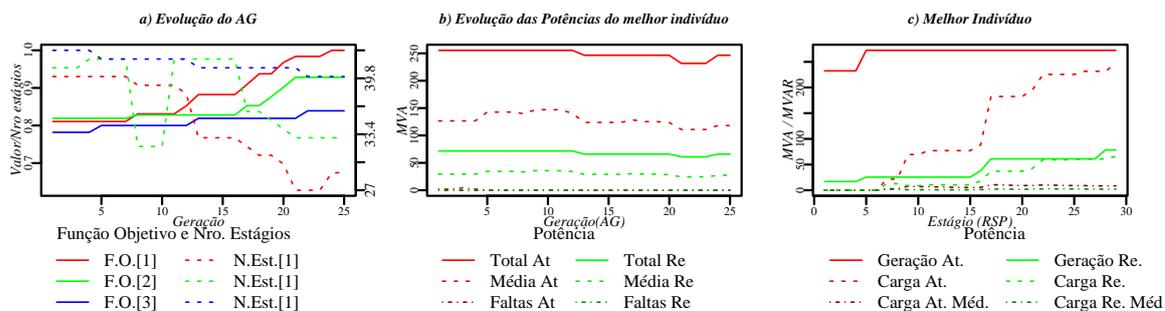


Figura 84: Sistema IEEE14, população=25, gerações=25, função=3, A=1, B=2 e C=1.

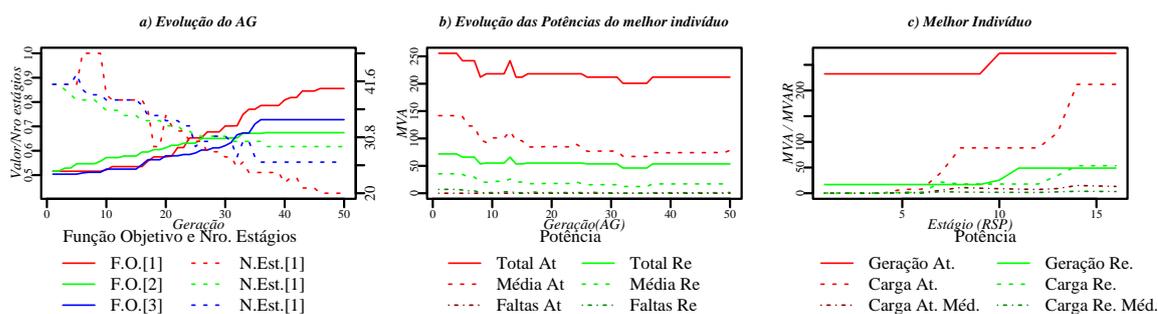


Figura 85: Sistema IEEE14, população=50, gerações=50, função=3, A=0, B=1 e C=1.

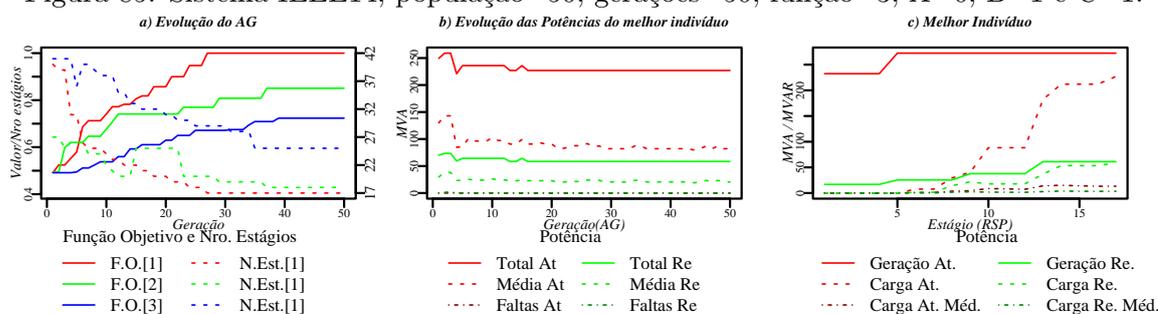


Figura 86: Sistema IEEE14, população=50, gerações=50, função=3, A=0, B=2 e C=1.

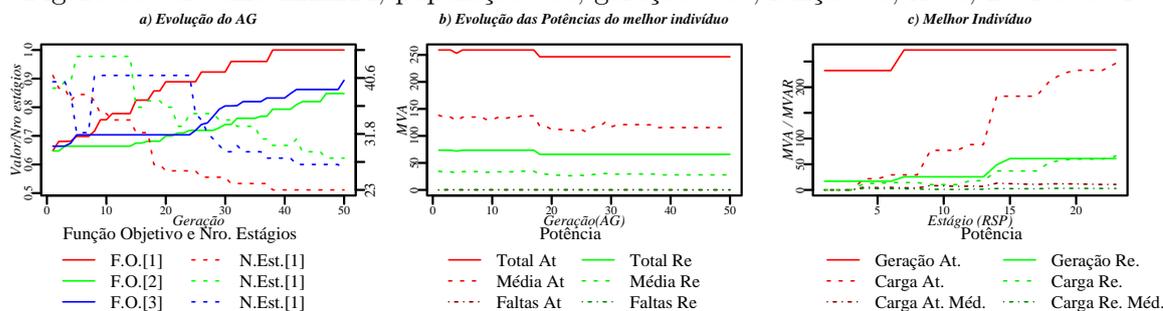


Figura 87: Sistema IEEE14, população=50, gerações=50, função=3, A=1, B=1 e C=1.

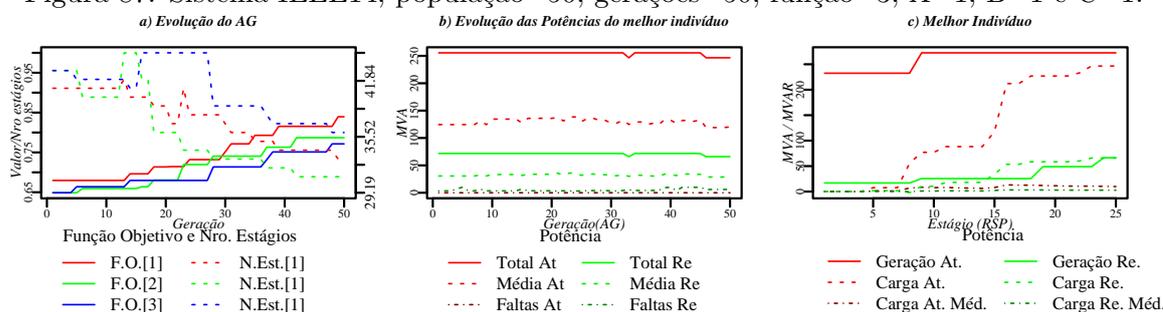


Figura 88: Sistema IEEE14, população=50, gerações=50, função=3, A=1, B=2 e C=1.

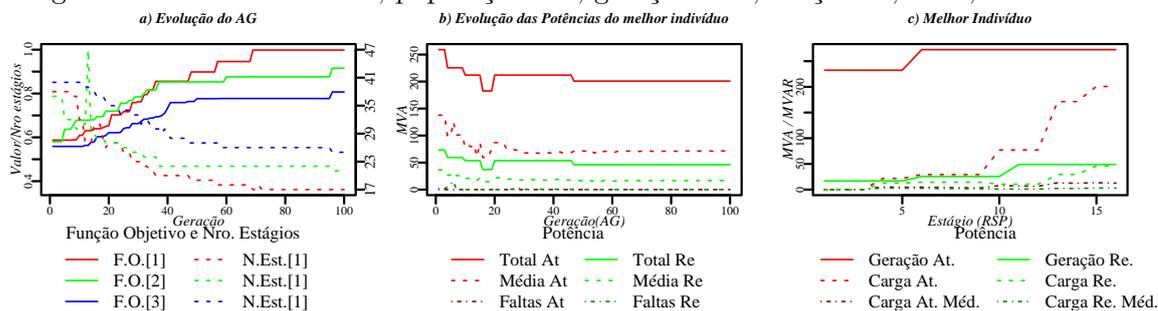


Figura 89: Sistema IEEE14, população=50, gerações=100, função=3, A=0, B=1 e C=1.

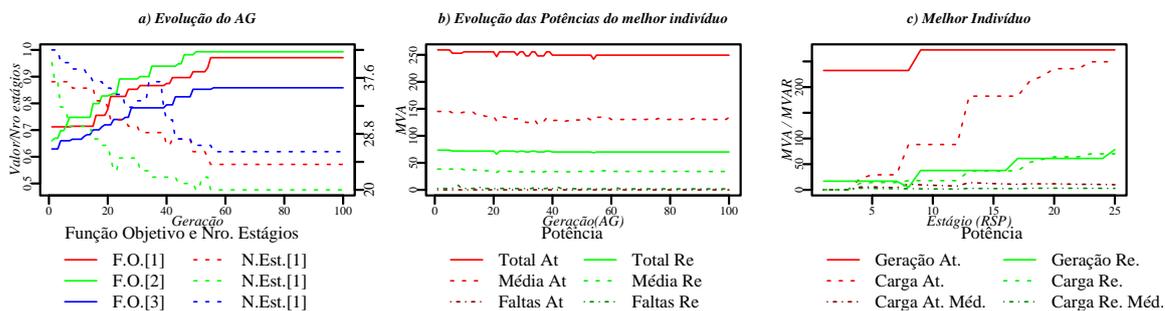


Figura 90: Sistema IEEE14, população=50, gerações=100, função=3, A=0, B=2 e C=1.

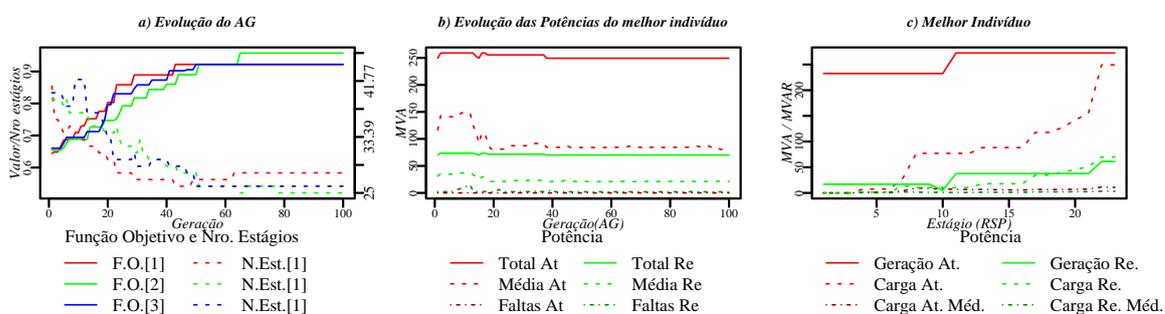


Figura 91: Sistema IEEE14, população=50, gerações=100, função=3, A=1, B=1 e C=1.

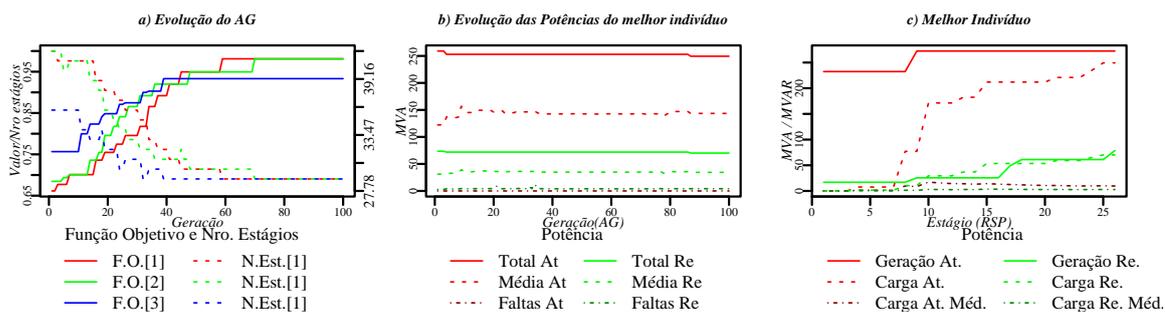


Figura 92: Sistema IEEE14, população=50, gerações=100, função=3, A=1, B=2 e C=1.

E.1.5 Função 4

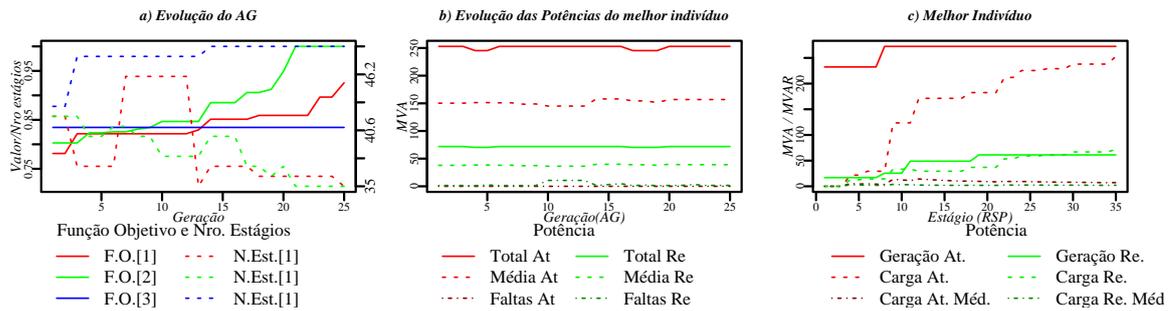


Figura 93: Sistema IEEE14, população=25, gerações=25, função=4, A=1, B=1 e C=1.

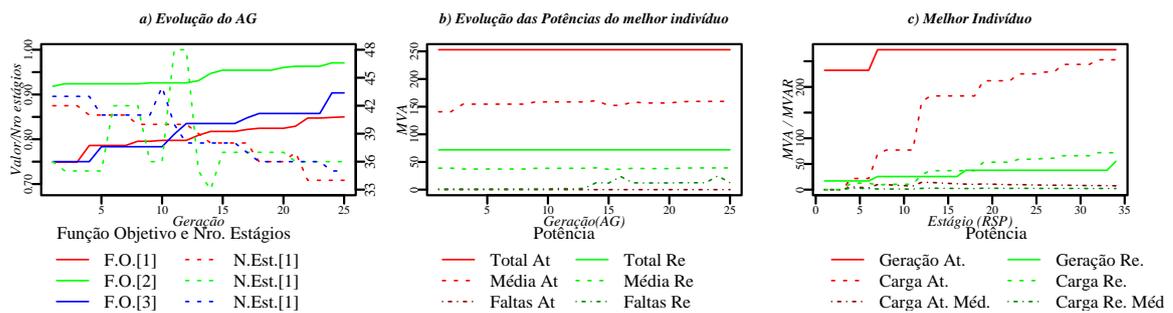


Figura 94: Sistema IEEE14, população=25, gerações=25, função=4, A=1, B=2 e C=1.

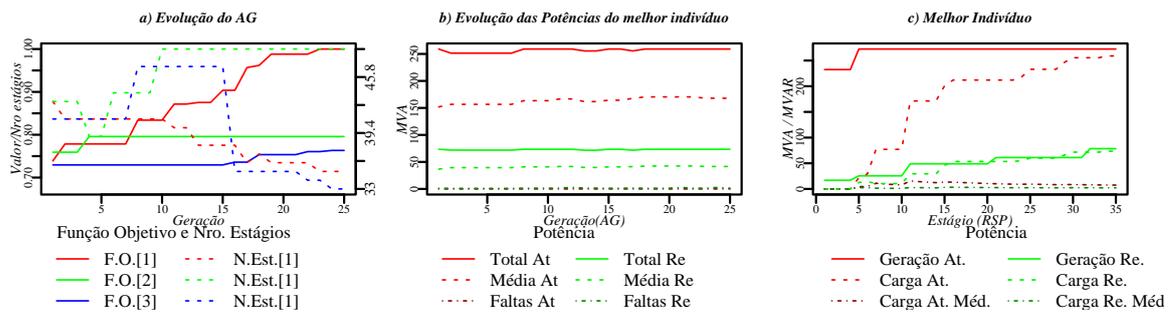


Figura 95: Sistema IEEE14, população=25, gerações=25, função=4, A=1, B=5 e C=1.

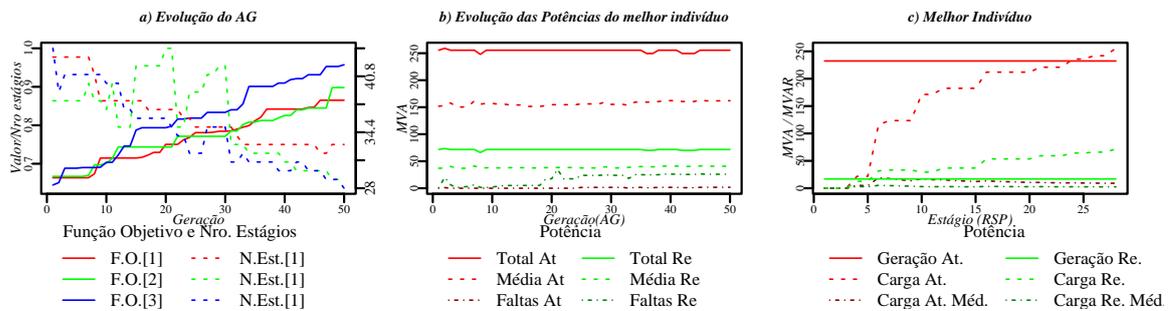


Figura 96: Sistema IEEE14, população=50, gerações=50, função=4, A=1, B=1 e C=1.

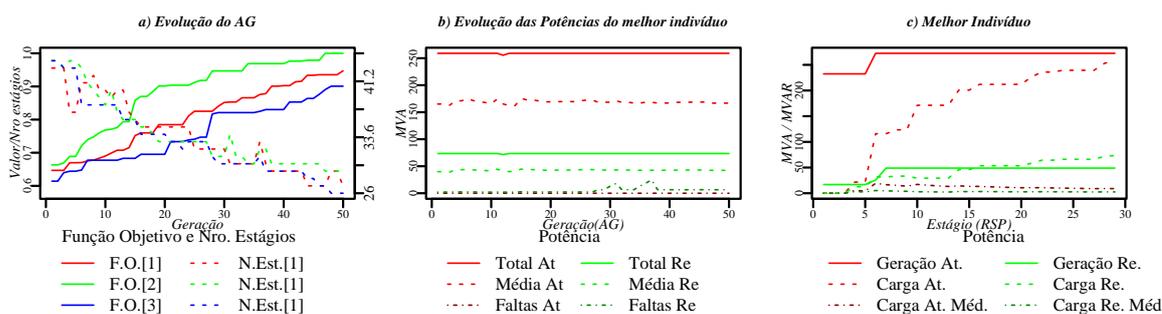


Figura 97: Sistema IEEE14, população=50, gerações=50, função=4, A=1, B=2 e C=1.

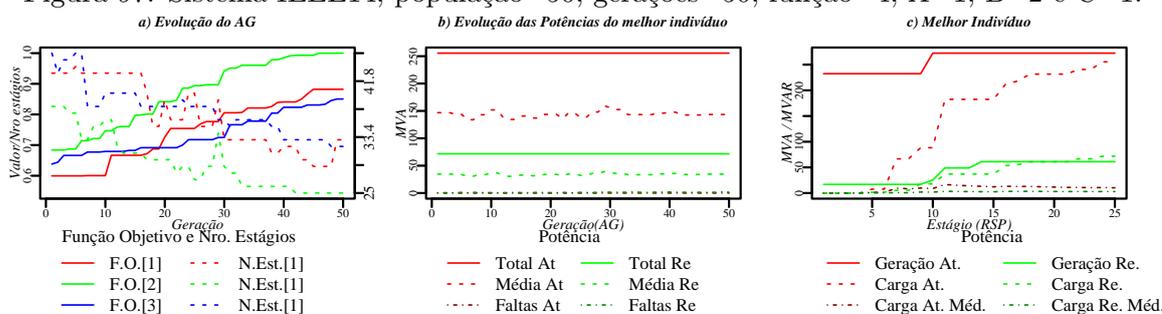


Figura 98: Sistema IEEE14, população=50, gerações=50, função=4, A=1, B=5 e C=1.

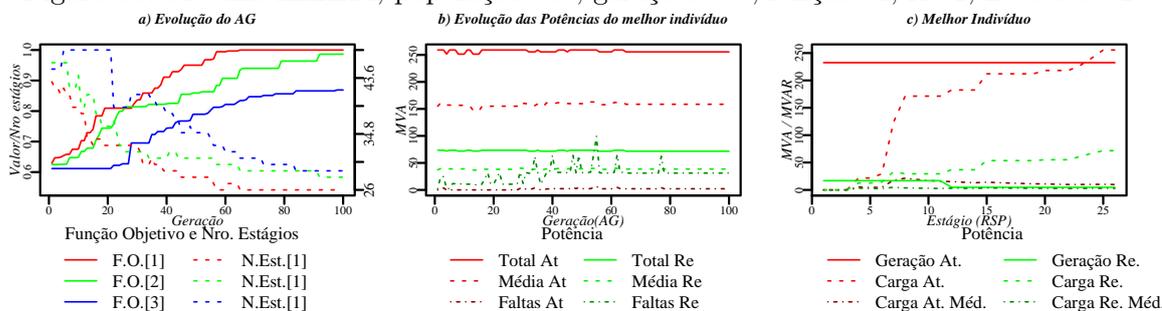


Figura 99: Sistema IEEE14, população=50, gerações=100, função=4, A=1, B=1 e C=1.

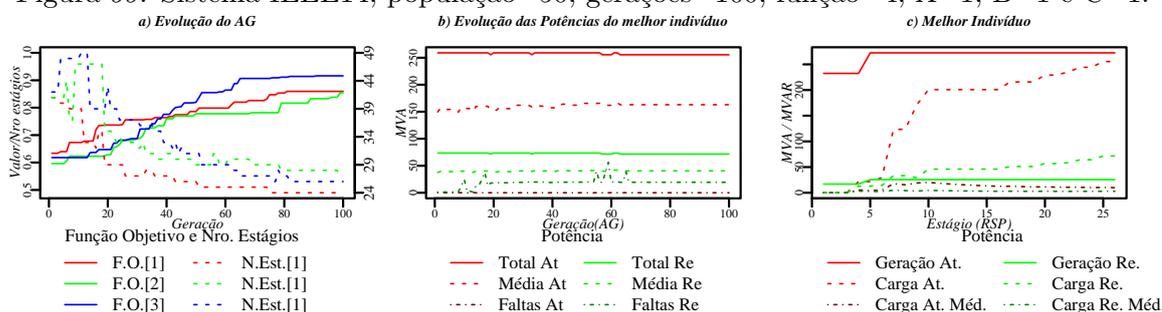


Figura 100: Sistema IEEE14, população=50, gerações=100, função=4, A=1, B=2 e C=1.

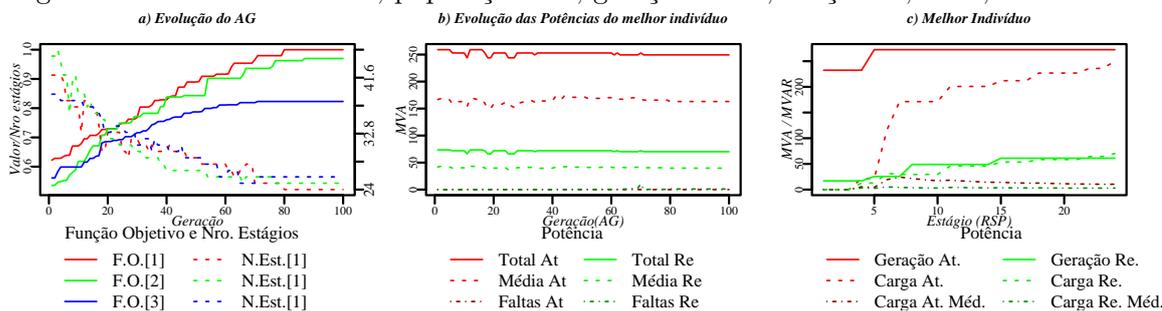


Figura 101: Sistema IEEE14, população=50, gerações=100, função=4, A=1, B=5 e C=1.

E.2 Sistema IEEE 30

E.2.1 Função 0

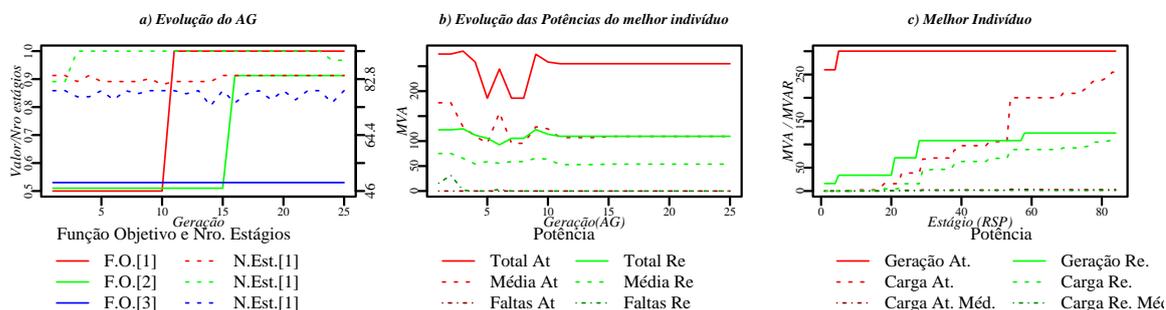


Figura 102: Sistema IEEE30, população=25, gerações=25, função=0, A=1 e B=1.

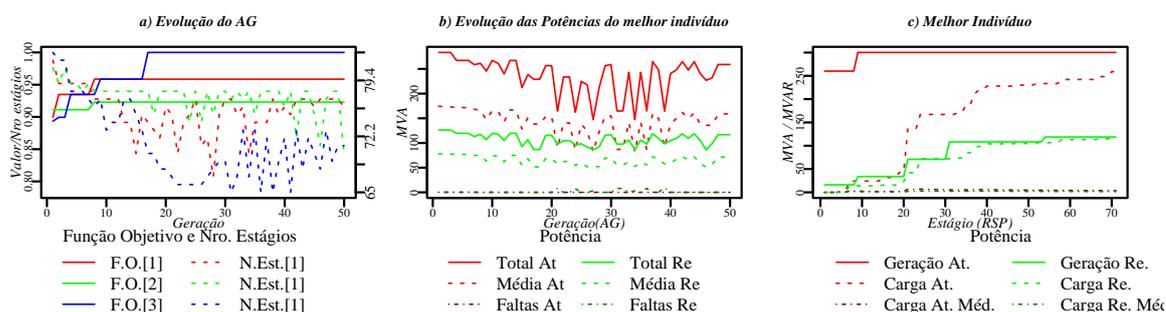


Figura 103: Sistema IEEE30, população=50, gerações=50, função=0, A=1 e B=1.

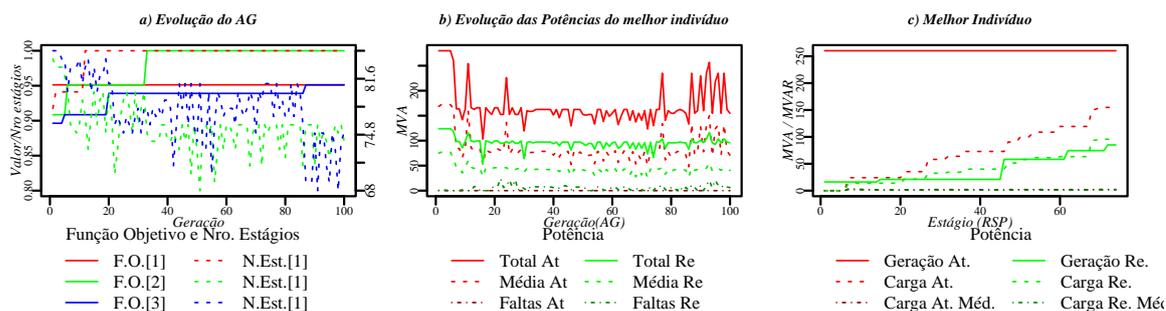


Figura 104: Sistema IEEE30, população=50, gerações=100, função=0, A=1 e B=1.

E.2.2 Função 1

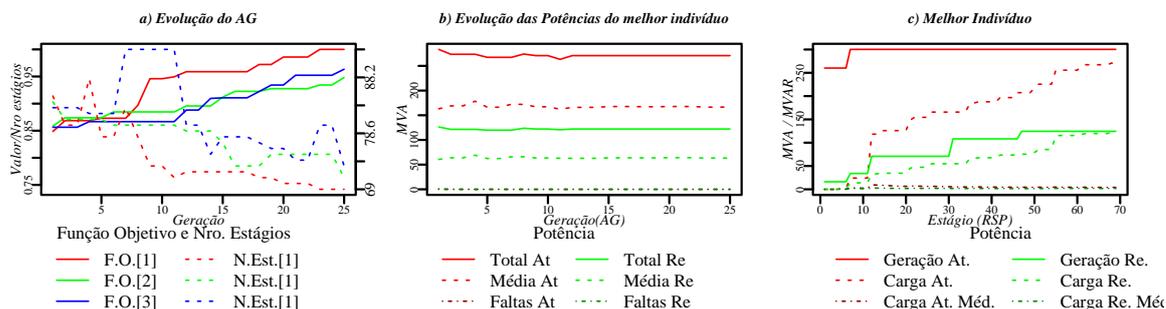


Figura 105: Sistema IEEE30, população=25, gerações=25, função=1, A=1, B=1 e C=0.

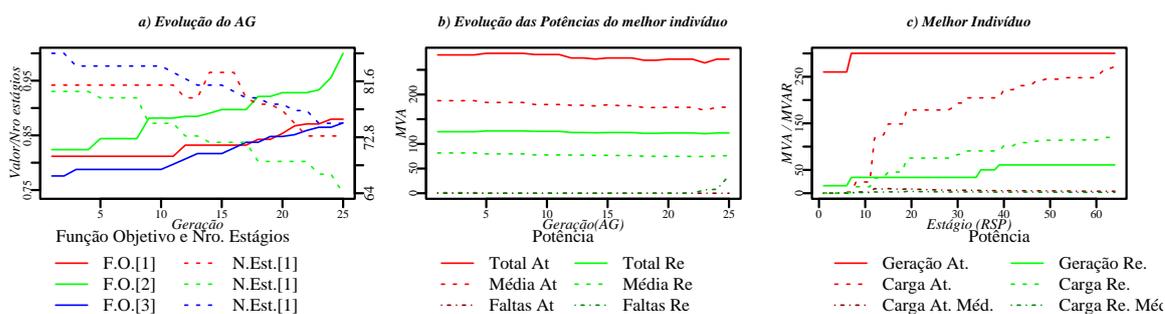


Figura 106: Sistema IEEE30, população=25, gerações=25, função=1, A=1, B=2 e C=0.

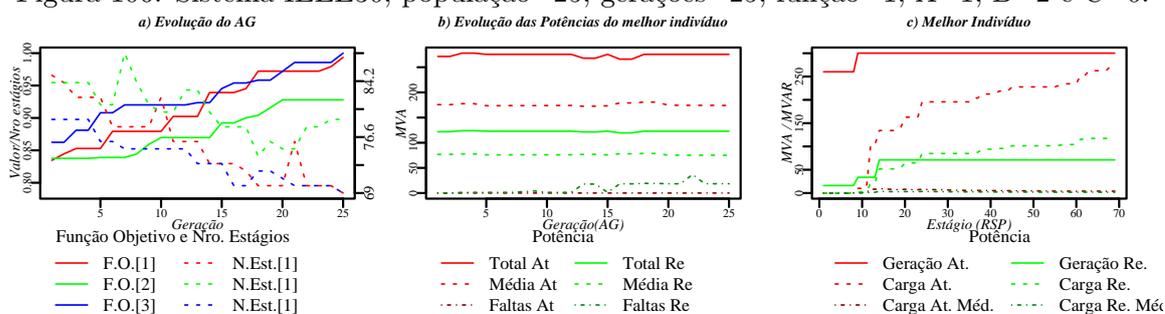


Figura 107: Sistema IEEE30, população=25, gerações=25, função=1, A=1, B=5 e C=0.

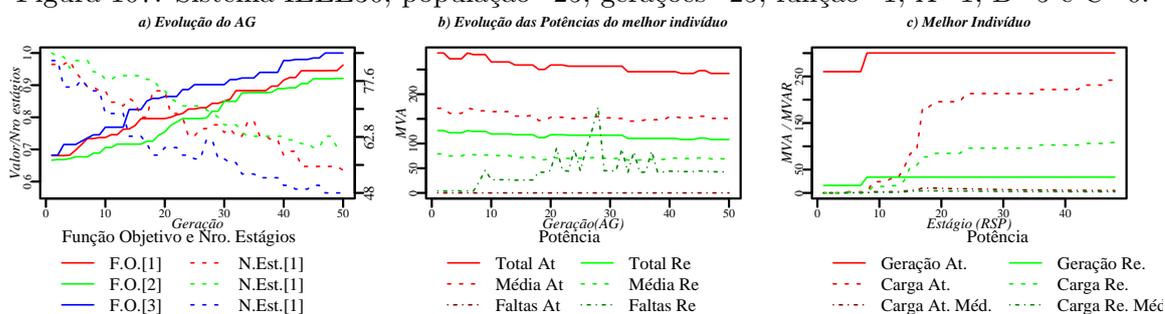


Figura 108: Sistema IEEE30, população=50, gerações=50, função=1, A=1, B=1 e C=0.

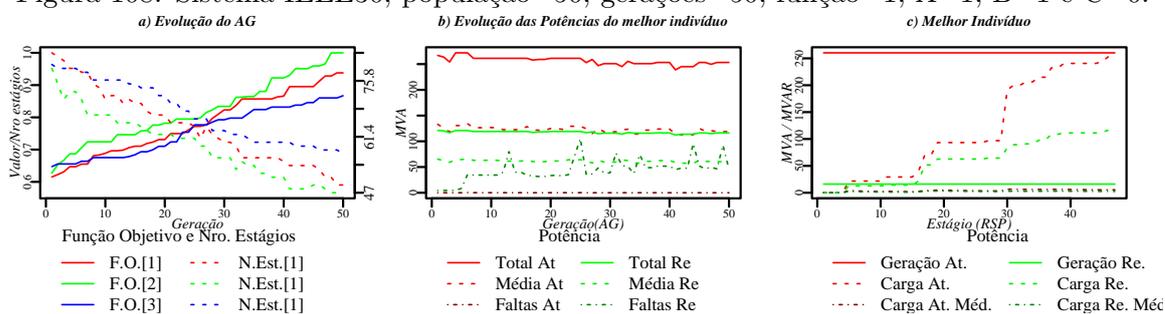


Figura 109: Sistema IEEE30, população=50, gerações=50, função=1, A=1, B=2 e C=0.

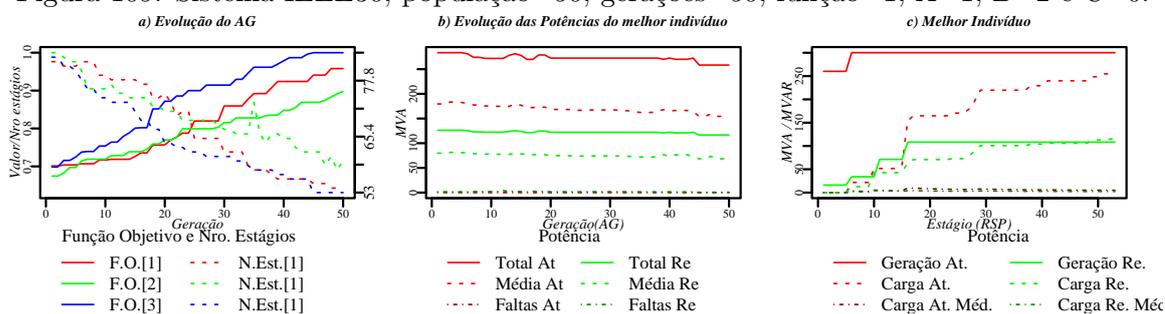


Figura 110: Sistema IEEE30, população=50, gerações=50, função=1, A=1, B=5 e C=0.

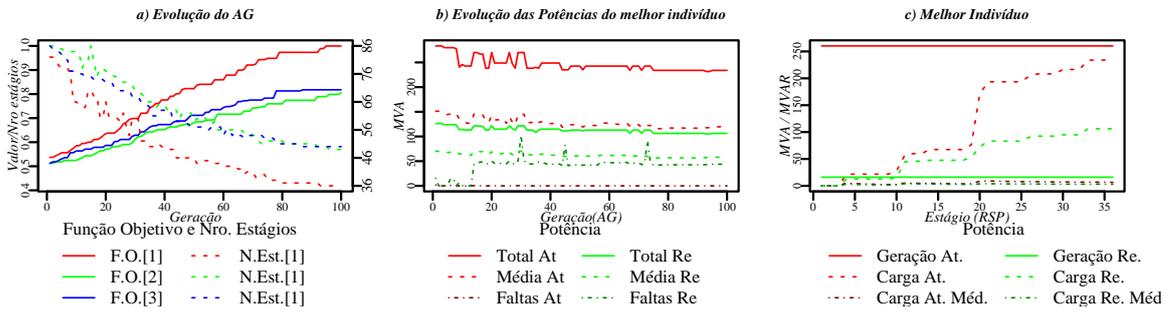


Figura 111: Sistema IEEE30, população=50, gerações=100, função=1, A=1, B=1 e C=0.

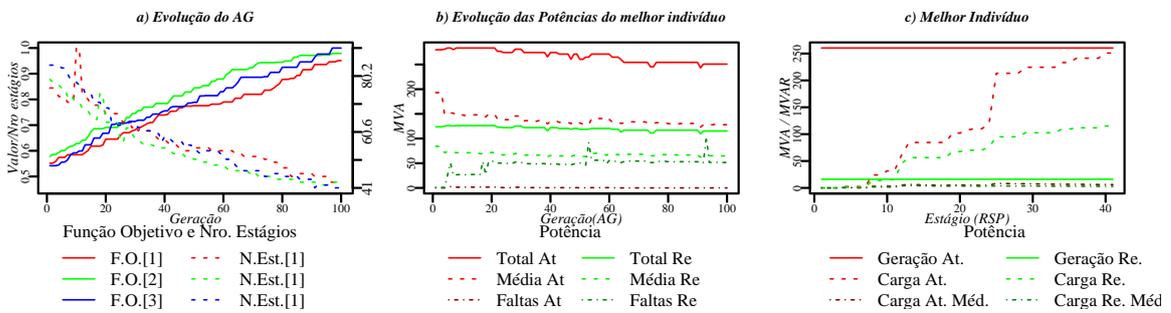


Figura 112: Sistema IEEE30, população=50, gerações=100, função=1, A=1, B=2 e C=0.

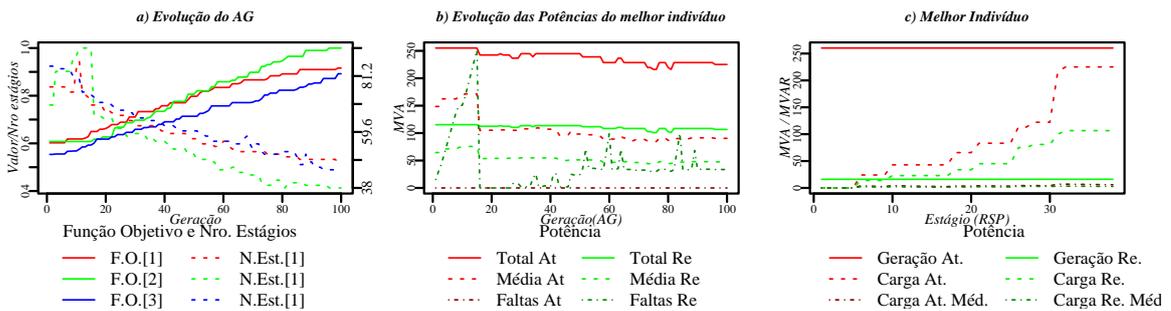


Figura 113: Sistema IEEE30, população=50, gerações=100, função=1, A=1, B=5 e C=0.

E.2.3 Função 2

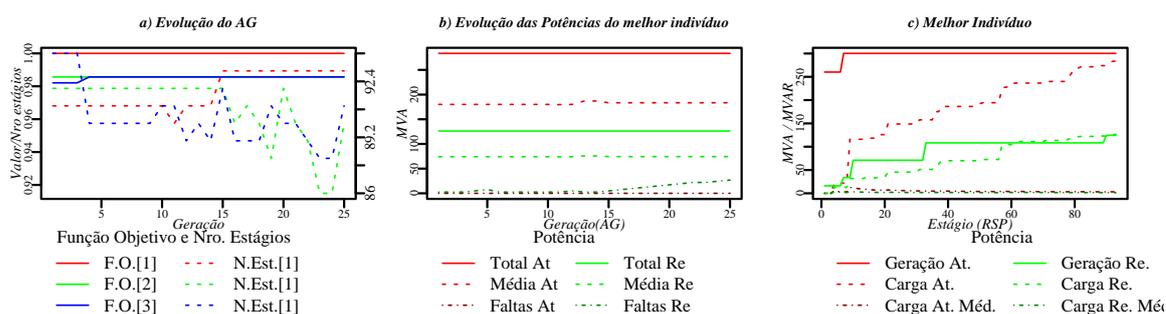


Figura 114: Sistema IEEE30, população=25, gerações=25, função=2, A=1, B=1 e C=0.

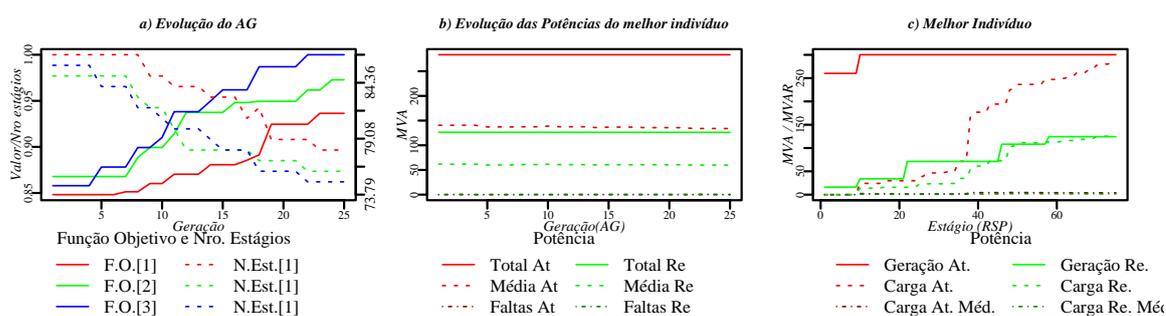


Figura 115: Sistema IEEE30, população=25, gerações=25, função=2, A=1, B=1 e C=1.

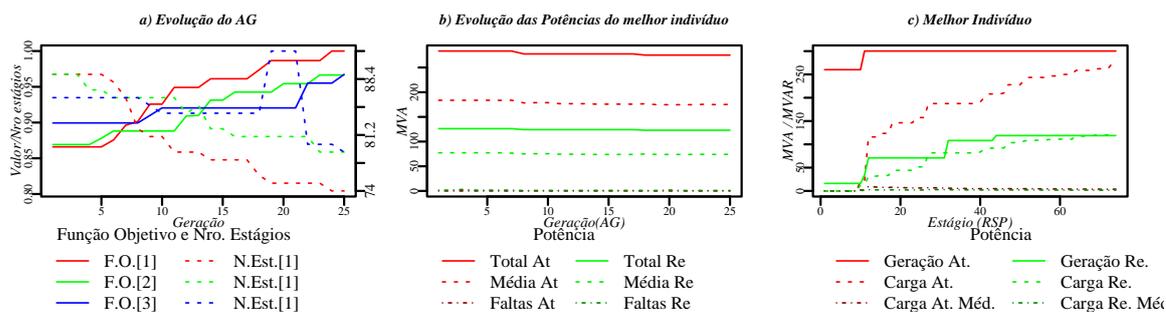


Figura 116: Sistema IEEE30, população=25, gerações=25, função=2, A=1, B=1 e C=2.

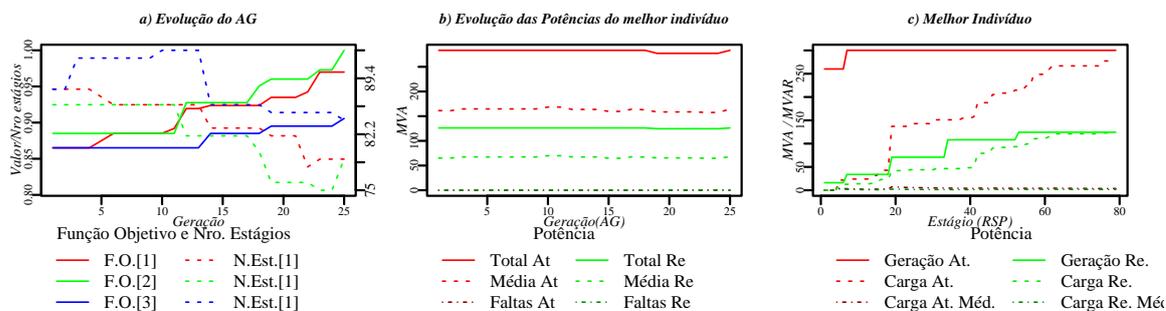


Figura 117: Sistema IEEE30, população=25, gerações=25, função=2, A=1, B=2 e C=1.

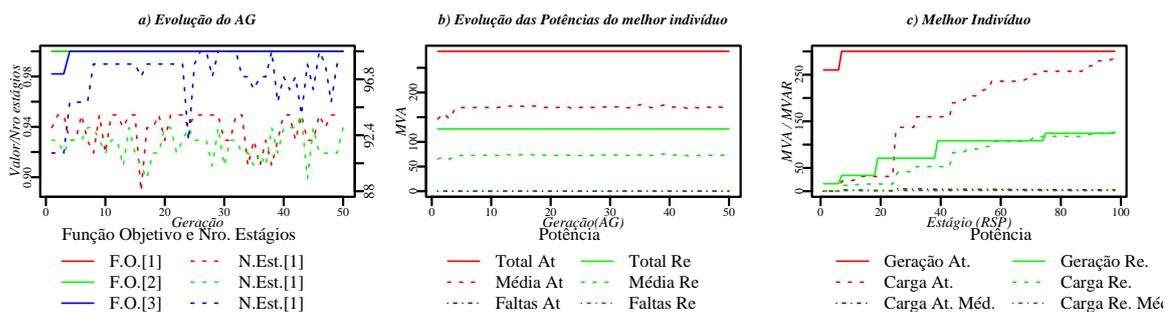


Figura 118: Sistema IEEE30, população=50, gerações=50, função=2, A=1, B=1 e C=0.

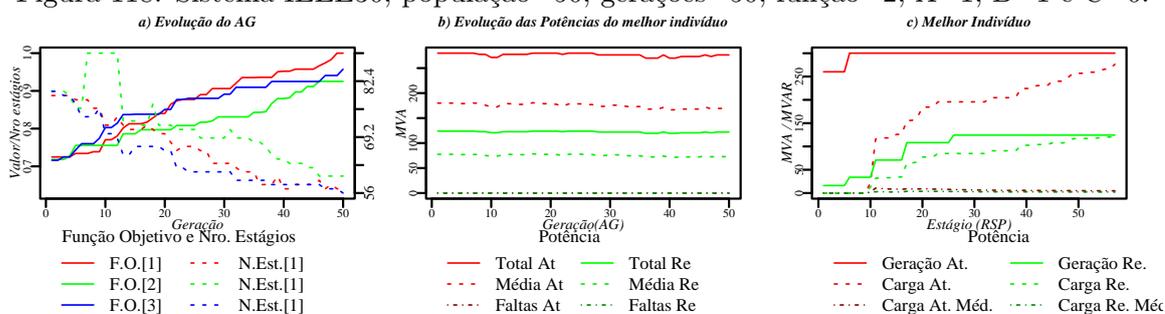


Figura 119: Sistema IEEE30, população=50, gerações=50, função=2, A=1, B=1 e C=1.

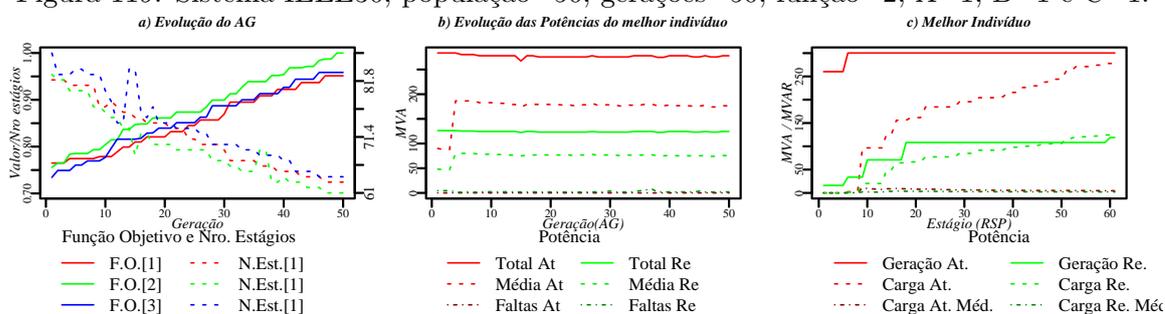


Figura 120: Sistema IEEE30, população=50, gerações=50, função=2, A=1, B=1 e C=2.

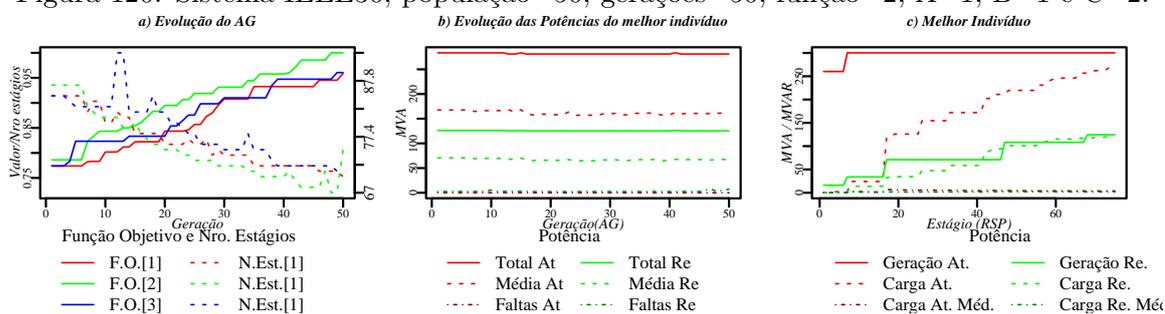


Figura 121: Sistema IEEE30, população=50, gerações=50, função=2, A=1, B=2 e C=1.

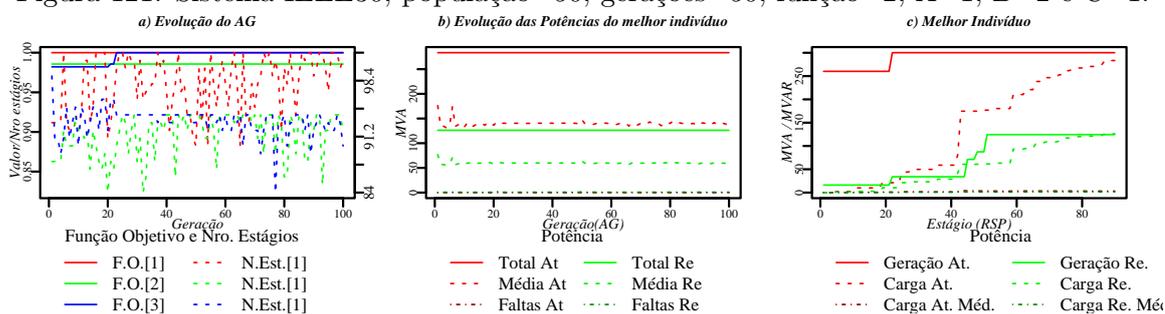


Figura 122: Sistema IEEE30, população=50, gerações=100, função=2, A=1, B=1 e C=0.

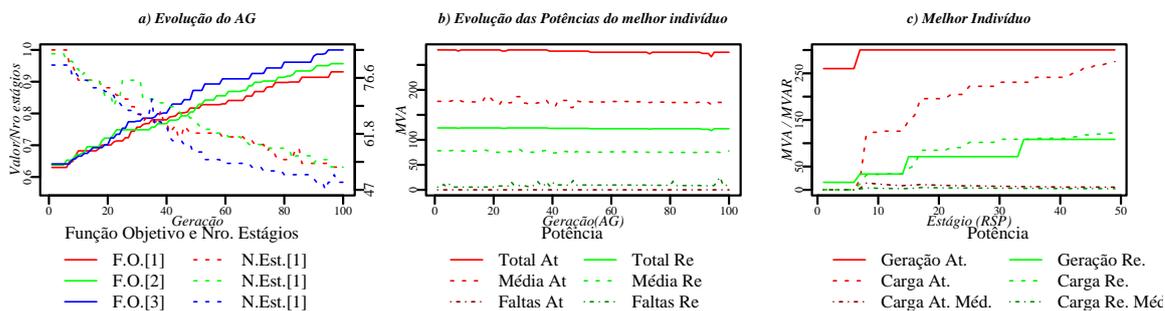


Figura 123: Sistema IEEE30, população=50, gerações=100, função=2, A=1, B=1 e C=1.

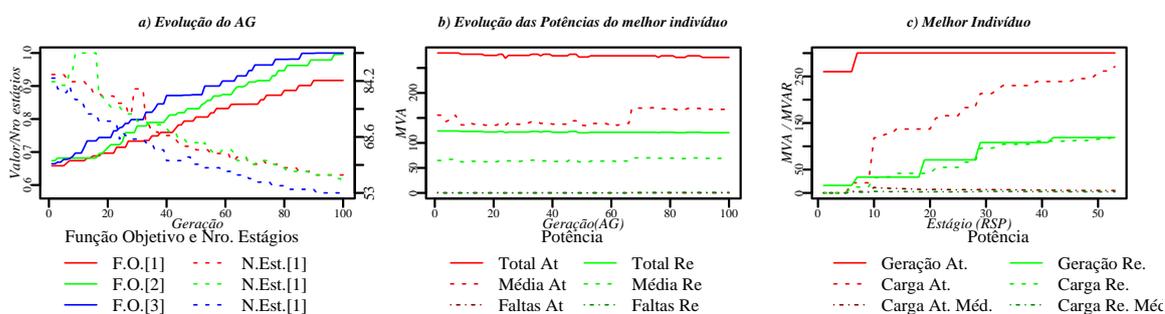


Figura 124: Sistema IEEE30, população=50, gerações=100, função=2, A=1, B=1 e C=2.

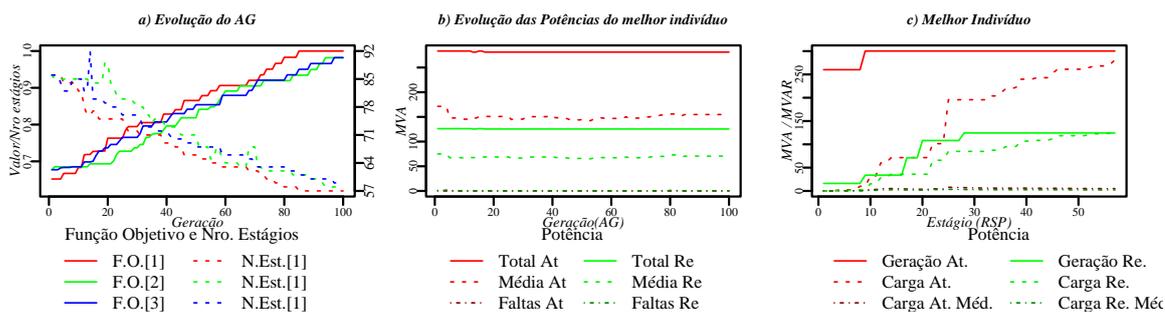


Figura 125: Sistema IEEE30, população=50, gerações=100, função=2, A=1, B=2 e C=1.

E.2.4 Função 3

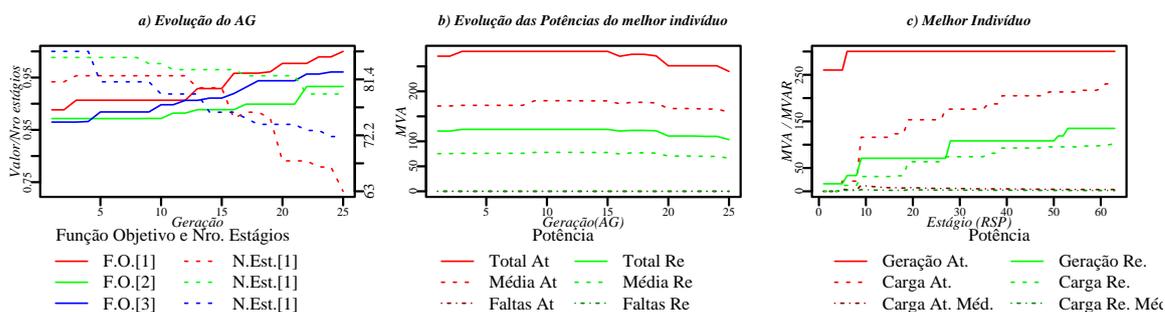


Figura 126: Sistema IEEE30, população=25, gerações=25, função=3, A=0, B=1 e C=1.

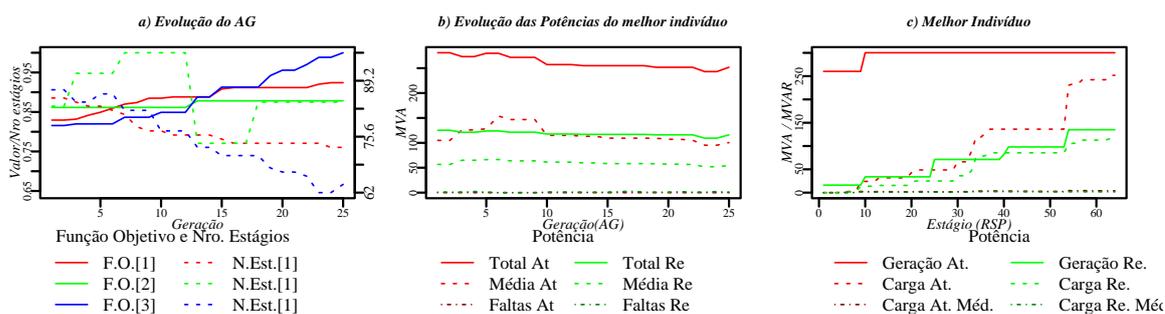


Figura 127: Sistema IEEE30, população=25, gerações=25, função=3, A=0, B=2 e C=1.

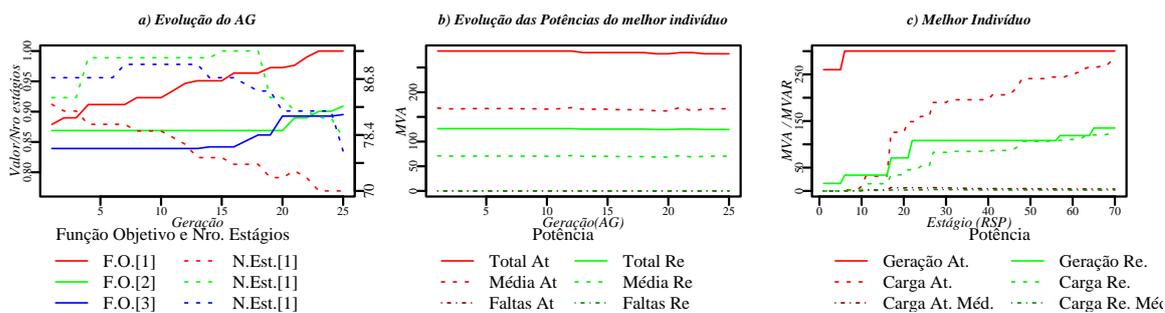


Figura 128: Sistema IEEE30, população=25, gerações=25, função=3, A=1, B=1 e C=1.

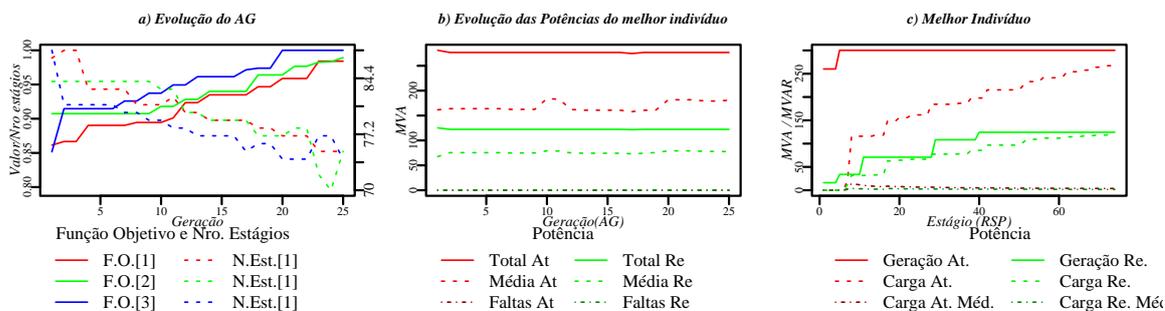


Figura 129: Sistema IEEE30, população=25, gerações=25, função=3, A=1, B=2 e C=1.

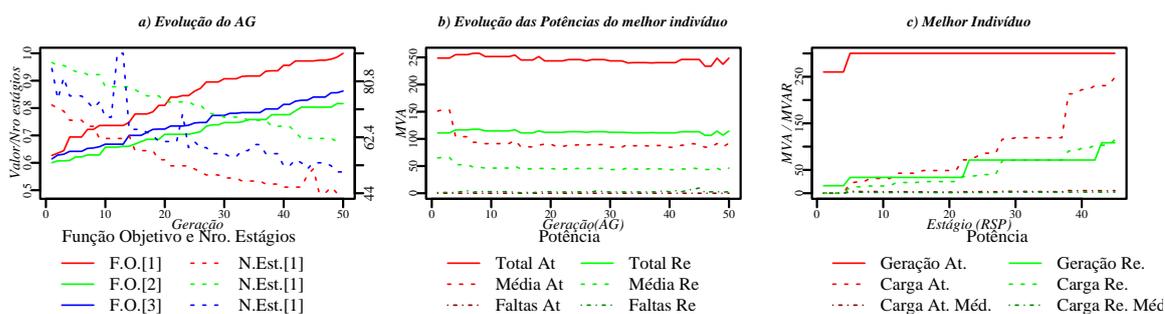


Figura 130: Sistema IEEE30, população=50, gerações=50, função=3, A=0, B=1 e C=1.

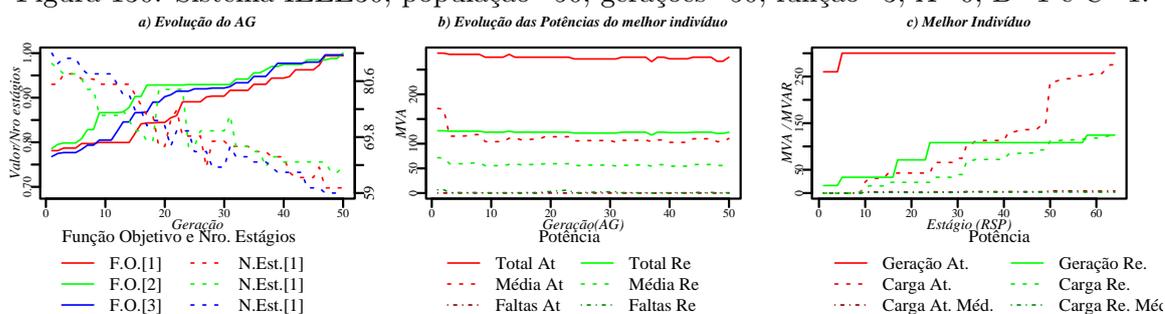


Figura 131: Sistema IEEE30, população=50, gerações=50, função=3, A=0, B=2 e C=1.

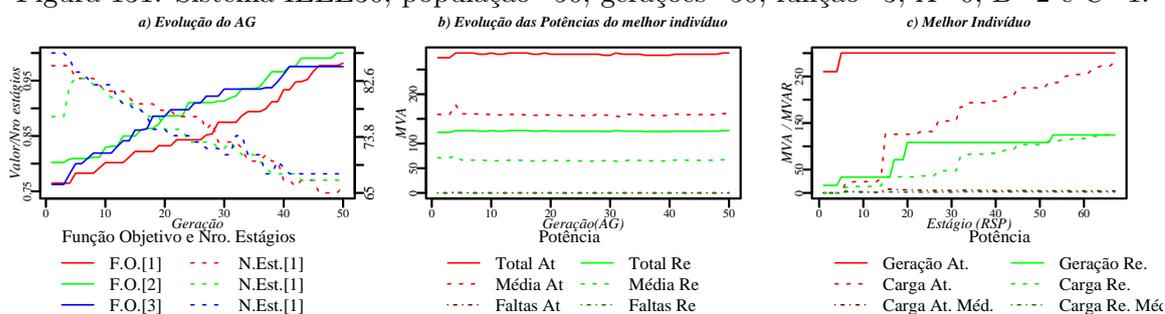


Figura 132: Sistema IEEE30, população=50, gerações=50, função=3, A=1, B=1 e C=1.

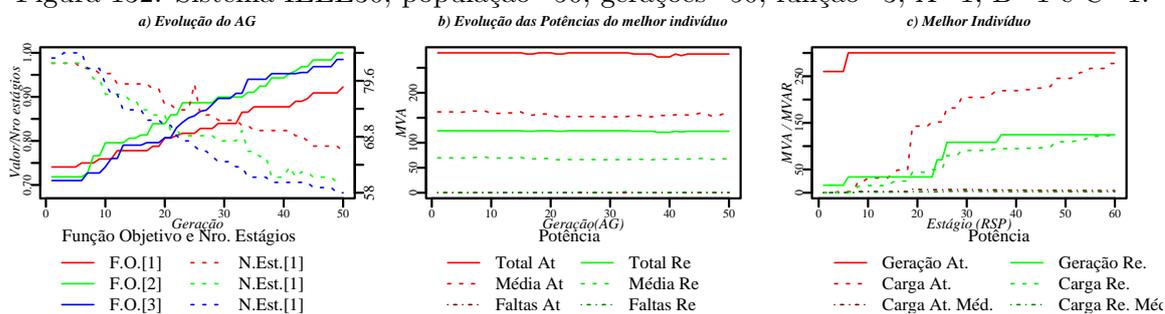


Figura 133: Sistema IEEE30, população=50, gerações=50, função=3, A=1, B=2 e C=1.

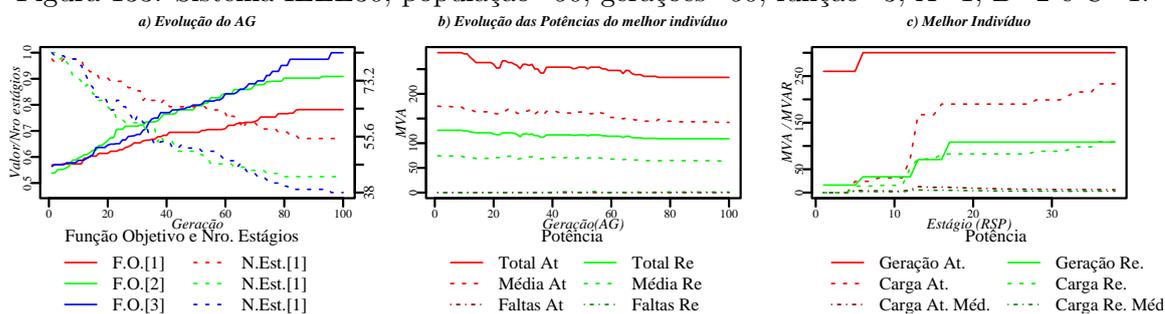


Figura 134: Sistema IEEE30, população=50, gerações=100, função=3, A=0, B=1 e C=1.

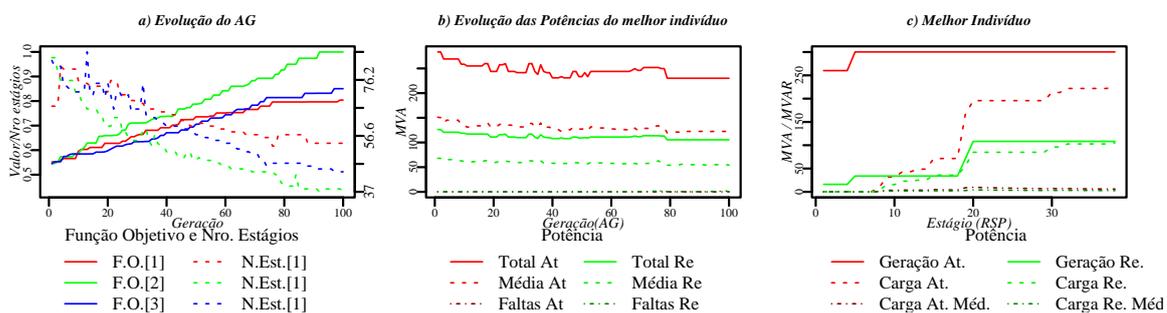


Figura 135: Sistema IEEE30, população=50, gerações=100, função=3, A=0, B=2 e C=1.

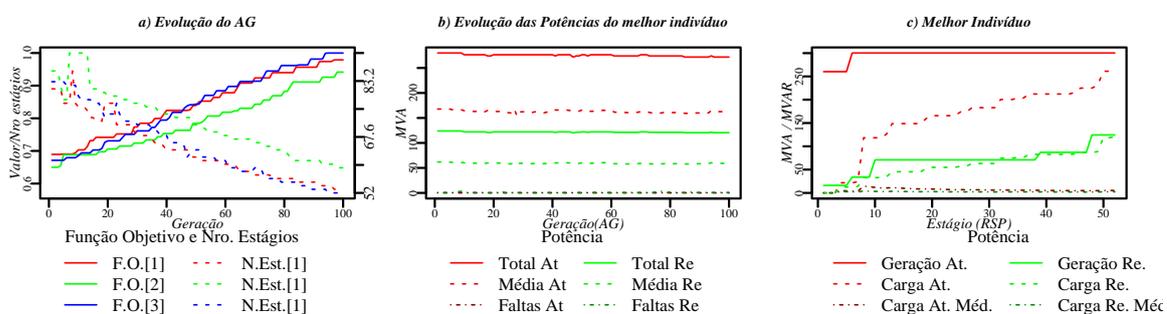


Figura 136: Sistema IEEE30, população=50, gerações=100, função=3, A=1, B=1 e C=1.

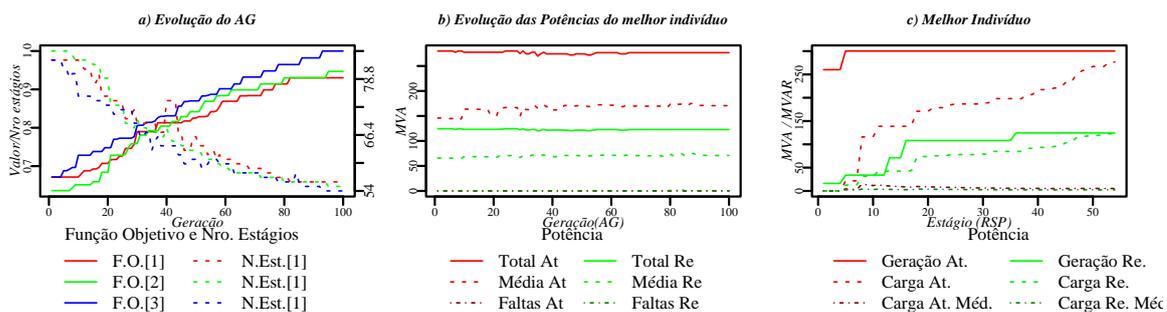


Figura 137: Sistema IEEE30, população=50, gerações=100, função=3, A=1, B=2 e C=1.

E.2.5 Função 4

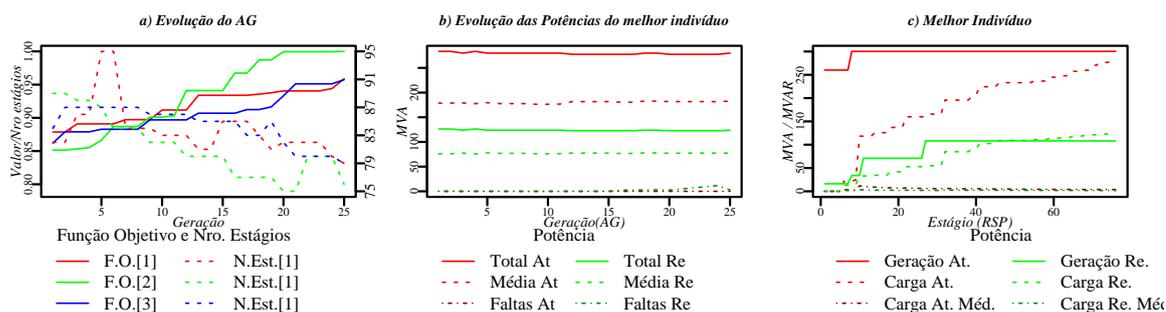


Figura 138: Sistema IEEE30, população=25, gerações=25, função=4, A=1, B=1 e C=1.

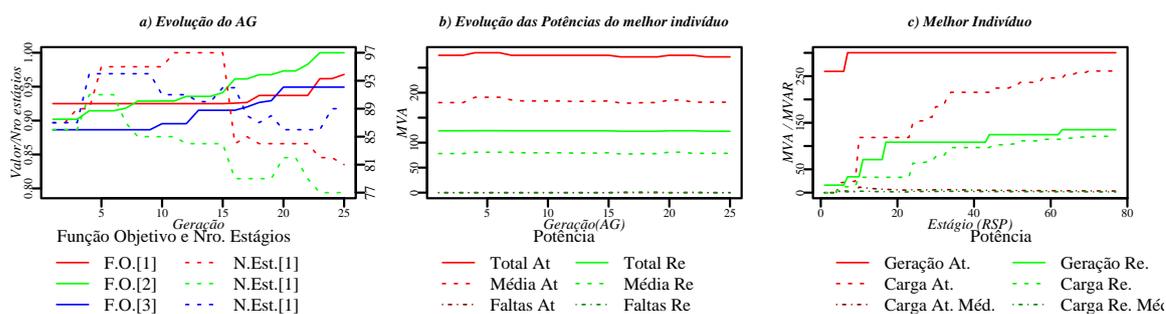


Figura 139: Sistema IEEE30, população=25, gerações=25, função=4, A=1, B=2 e C=1.

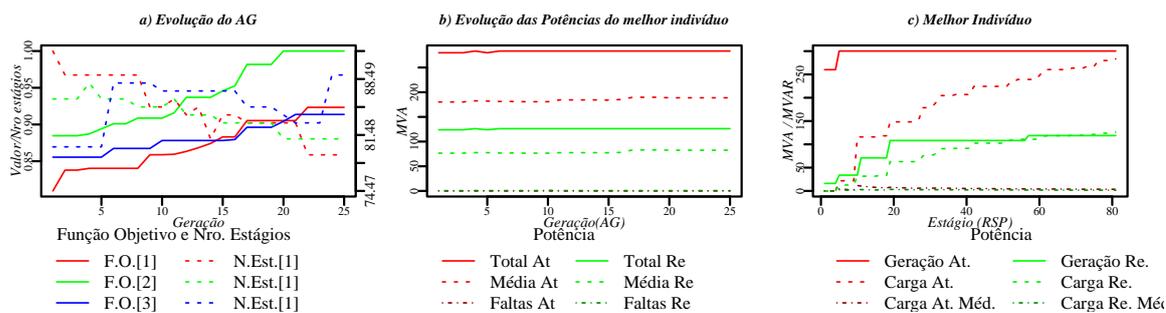


Figura 140: Sistema IEEE30, população=25, gerações=25, função=4, A=1, B=5 e C=1.

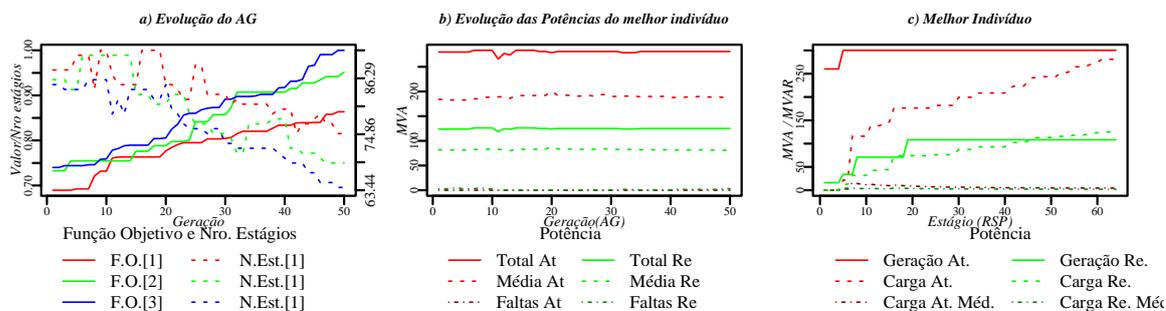


Figura 141: Sistema IEEE30, população=50, gerações=50, função=4, A=1, B=1 e C=1.

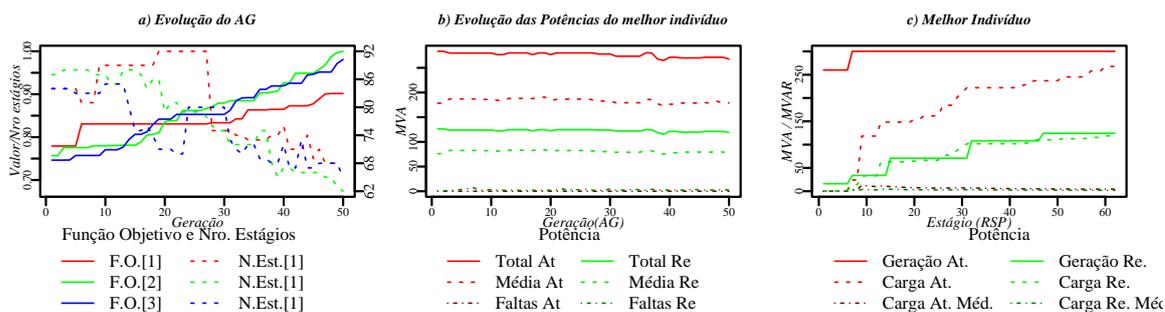


Figura 142: Sistema IEEE30, população=50, gerações=50, função=4, A=1, B=2 e C=1.

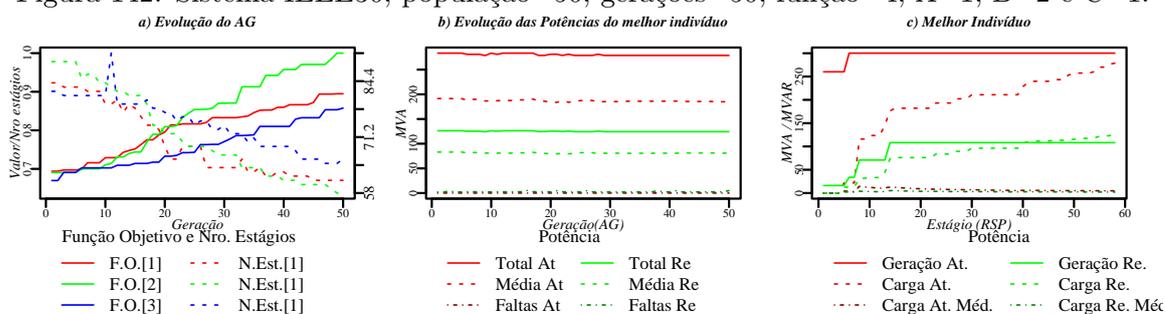


Figura 143: Sistema IEEE30, população=50, gerações=50, função=4, A=1, B=5 e C=1.

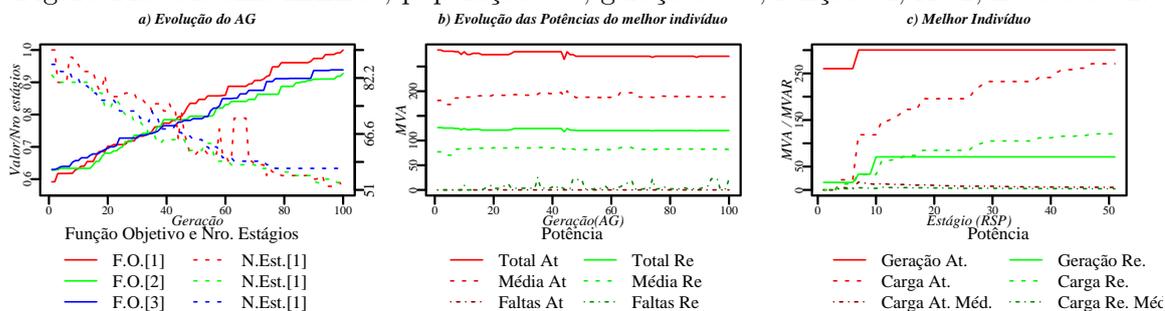


Figura 144: Sistema IEEE30, população=50, gerações=100, função=4, A=1, B=1 e C=1.

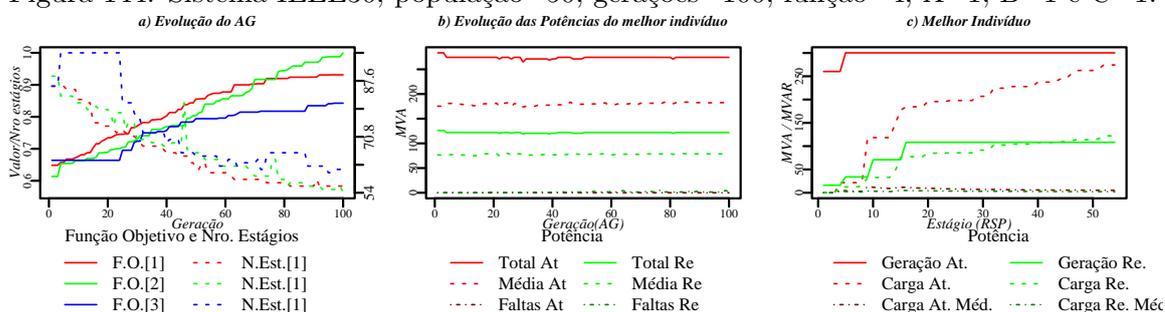


Figura 145: Sistema IEEE30, população=50, gerações=100, função=4, A=1, B=2 e C=1.

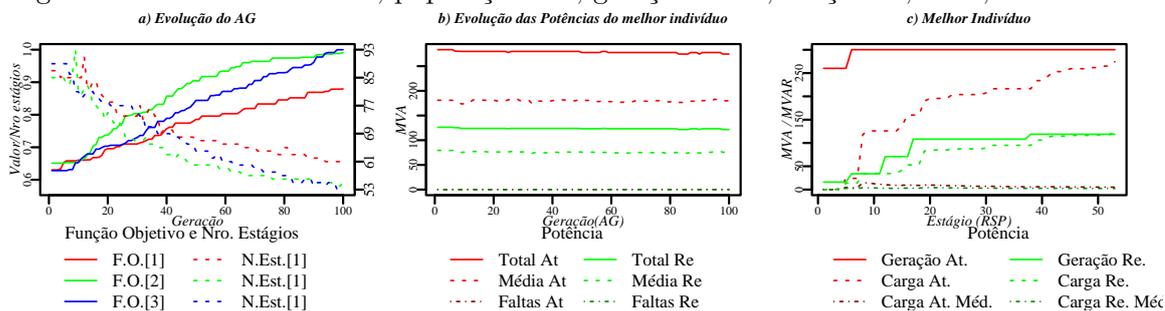


Figura 146: Sistema IEEE30, população=50, gerações=100, função=4, A=1, B=5 e C=1.

REFERÊNCIAS

- ADIBI, M. M.; BORKOSKI, J. N.; KAFKA, R. J. Analytical tool requirements for Power System Restoration. *IEEE Transactions on Power Systems*, v. 9, n. 3, p. 1582–1591, agosto 1994.
- ADIBI, M. M.; FINK, L. H. Power System Restoration Planning. *IEEE Transactions on Power Systems*, v. 9, n. 1, p. 22–28, janeiro 1994.
- ADIBI, M. M.; FINK, L. H. *et al.* New approaches in Power System Restoration: A report by the Power System Restoration working group. *IEEE Transactions on Power Systems*, v. 7, n. 4, p. 1428–1434, novembro 1992.
- ADIBI, M. M.; FINK, L. H. *et al.* Special considerations on Power System Restoration. *IEEE Transactions on Power Systems*, v. 7, n. 4, p. 1419–1427, novembro 1992.
- ADIBI, M. M.; KAFKA, R. J.; MILANICZ, D. P. Expert system requirements for Power System Restoration. *IEEE Transactions on Power Systems*, v. 9, n. 3, p. 1592–1600, agosto 1994.
- ADIBI, M. M.; MILANICZ, D. P. Protective system issues during restoration. *IEEE Transactions on Power Systems*, v. 10, n. 3, p. 1492–1497, agosto 1995.
- ANCONA, J. J. A framework for power system restoration following a major power failure. *IEEE Transactions on Power Systems*, v. 10, n. 3, p. 1480–1485, agosto 1995.
- BANN, J. J.; IRISARRI, G. *et al.* Integrating ai applications in an energy management system. *IEEE Expert*, v. 12, n. 6, p. 53–59, novembro 1991.
- BRETAS, A.; PHADKE, A. G. Artificial neural networks in Power System Restoration. *IEEE Transactions on Power Delivery*, v. 18, n. 4, p. 1181–1186, outubro 2003.
- BRETAS, N. G.; DELBEN, A. C. B.; CARVALHO, A. de. Optimal energy restoration for general distribution systems by genetic algorithm. In: *Power System Technology, 1998 International Conference on*. [S.l.: s.n.], 1998. v. 1, n. 1, p. 43–47.
- EL-HAWARY, M. E. *Electrical Energy Systems*. Boca Raton, FL: CRC Press, 2000.
- FUKUYAMA, Y.; ENDO, H.; NAKANISHI, Y. A hybrid system for service restoration using expert system and genetic algorithm. In: *Intelligent Systems Applications to Power Systems, International Conference on*. [S.l.: s.n.], 1996. v. 1, p. 394–398.
- F.WOLLENBERG, B.; SAKAGUCHI, T. Artificial intelligence in power system operations. *Proceedings of the IEEE*, v. 75, n. 12, p. 1678–1685, dezembro 1987.
- GOMES, P. New strategies to improve bulk power system security: lessons learned from large blackouts. In: *IEEE Power Engineering Society, Proceedings of the*. [S.l.: s.n.], 2004.
- GOMES, P.; LIMA, A. C. S. de; GUARINI, A. de P. Guidelines for power system restoration in the brazilian system. *IEEE Transactions on Power Systems*, v. 19, n. 2, p. 1159–1164, maio 2004.

- GRIGSBY, L. L.; HANSON, A. Power flow analysis. In: GRIGSBY, L. L. (Ed.Chefe). *The Electric Power Engineering Handbook*. 1. ed. Boca Raton, FL: CRC Press, 2001. p. 1–1.
- HOLLAND, J. *Adaptation in Natural and Artificial Systems*. Cambridge, MA: MIT Press, 1999.
- HSU, Y. Y.; HUANG, H. M. Distribution systems service restoration using the artificial neural network approach and pattern recognition method. In: *IEE Proceeding on Generation, Transmission and Distribution*. [S.l.: s.n.], 1995. v. 142, n. 3, p. 251–256.
- HUANG, J. A.; AUDETTE, L.; HARRISON, S. A systematic method for power system restoration planning. *IEEE Transactions on Power Systems*, v. 10, n. 2, p. 869–875, maio 1995.
- HUANG, J. A.; GALIANA, F. D.; VUONG, G. T. Power system restoration incorporating interactive graphics and optimization. In: *Power Industry Computer Application Conference, 1991. Conference Proceedings*. Baltimore: [s.n.], 1991. p. 216–222.
- ISLAM, S.; CHOWDHURY, N. A case-based windows graphic package for the education and training of power system restoration. *IEEE Transactions on Power Systems*, v. 16, n. 2, p. 181–187, maio 2001.
- JOHNSON, D. E.; HILBURN, J. L.; JOHNSON, J. R. *Fundamentos de Análise de Circuitos Elétricos*. 4. ed. Rio de Janeiro, RJ: Prentice/Hall do Brasil, 1994.
- JOHNSON, W. A.; PRABHAKARA, F. S. *et al.* System restoration - deploying the plan. *IEEE Transactions on Power Apparatus and Systems*, PAS-101, n. 11, p. 4263–4271, novembro 1982.
- KIRSCHEN, D. S.; VOLKMANN, T. L. Guiding a Power System Restoration with and expert system. *IEEE Transactions on Power Systems*, v. 6, n. 2, p. 558–566, maio 1991.
- KOJIMA, Y. *et al.* Development of a guidance method for Power System Restoration. *IEEE Transactions on Power Systems*, v. 4, n. 2, p. 1219–1227, agosto 1989.
- KOJIMA, Y. *et al.* The development of power system restoration method for a bulk power system by applying knowledge engineering techniques. *IEEE Transactions on Power Systems*, v. 4, n. 2, p. 1228–1235, agosto 1989.
- KOMAI, K.; MATSUMOTO, K.; SAKAGUCHI, T. Analysis and evaluation of expert's knowledge for Power System Restoration by mathematical programming method. *IEEE International Symposium on Circuits and Systems*, v. 2, n. 1, p. 1895–1898, junho 1988.
- KOSTIE, T. *et al.* Decision aid function for restoration of transmission power systems: Conceptual design and real time consideration. *IEEE Transactions on Power Systems*, v. 13, n. 3, p. 923–929, agosto 1998.
- LIOU, K.-L.; LIU, C.-C.; CHU, R. F. Tie line utilization during Power System Restoration. *IEEE Transactions on Power Systems*, v. 10, n. 1, p. 192–199, fevereiro 1995.
- LIU, C.-C.; LEE, S. J.; VENKATA, S. S. An expert system operational aid for restoration and loss reduction of distribution systems. *IEEE Transactions on Power Systems*, v. 3, n. 2, p. 619–626, maio 1988.
- LUAN, W. P.; IRVING, M. R.; DANIEL, J. S. Genetic algorithm for supply restoration and optimal load shedding in power system distribution networks. In: *IEE Proceeding on Generation, Transmission and Distribution*. [S.l.: s.n.], 2002. v. 149, n. 2, p. 145–151.

MATSUMOTO, K. *et al.* Knowledge-based systems as operational aids in Power System Restoration. *Proceedings of the IEEE*, v. 80, n. 5, p. 689–697, maio 1992.

MELLONI, E. *Cabo não foi única causa do blecaute, dizem especialistas*. Agência Estado, 2002. Disponível em: <http://www.copa.esp.br/agestado/noticias/2002/jan/22/295.htm>. Acesso em: 01 jun. 2005.

MITCHELL, M. *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press, 1999.

NAGATA, T. *et al.* A new Power System Restoration framework by means of multi-agent approach. *Transmission and Distribution Conference and Exhibition*, v. 3, n. 1, p. 1695–1700, outubro 2002.

NAGATA, T.; SASAKI, H. A multi-agent approach to Power System Restoration. *IEEE Transactions on Power Systems*, v. 17, n. 2, p. 457–462, maio 2002.

NAGATA, T.; SASAKI, H.; R.YOKOYAMA. Power System Restoration by joint usage of expert sytem and mathematical programming approach. *IEEE Transactions on Power Systems*, v. 10, n. 3, p. 1473–1479, agosto 1995.

RADIOBRÁS. *Sinopse - Resumo dos Jornais*. Radiobrás, 2002. Disponível em: http://www.radiobras.gov.br/anteriores/2002/sinopses_2201.htm. Acesso em: 01 jun. 2005.

SAKAGUCHI, T.; MATSUMOTO, K. Development of a knowledge based system for Power System Restoration. *IEEE Transactions on Power Apparatus and Systems*, PAS-102, n. 2, p. 320–329, fevereiro 1983.

SANTOS, J. V. C. dos *et al.* Contribuições para a solução do problema de restauração de sistemas de potência: Investigação do uso de algoritmos genéticos e aplicação de processamento distribuído. In: *Anais do XV Congresso Brasileiro de Automática*. Gramado: [s.n.], 2004.

SANTOS, J. V. C. dos; GARCIA, A. V. Um método para auxiliar processos de restauração de sistemas de potência. *Revista Controle e Automação*, v. 15, n. 1, p. 85–92, janeiro 2004.

TEREZA, N. P. e I. *ONS dá sua versão fina para apagão*. Agência Estado, 2002. Disponível em: <http://www.copa.esp.br/agestado/noticias/2002/jan/22/315.htm>. Acesso em: 01 jun. 2005.

WALL, M. *GAlib: A C++ Library of Genetic Algorithm Components*. [S.l.], 1996. Disponível em: <http://lancet.mit.edu/ga/dist/galibdoc.pdf>. Acesso em: 03/01/2006.

WU, F. F.; MONTICELLI, A. Analytical tools for Power System Restoration - conceptual design. *IEEE Transactions on Power Systems*, v. 3, n. 1, p. 10–16, fevereiro 1988.