Augusto Cesar dos Santos

Algoritmo Evolutivo Computacionalmente Eficiente para Reconfiguração de Sistemas de Distribuição

Tese de Doutorado apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo como parte dos requisitos para obtenção do título de Doutor em Engenharia Elétrica.

Orientador: Newton G. Bretas
Co-orientador: Alexandre C. B. Delbem

São Carlos

2009

FOLHA DE JULGAMENTO

Candidato: Engenheiro **AUGUSTO CESAR DOS SANTOS**

Tese defendida e julgada em 24/04/2009 perante a Comissão Julgadora:

Prof. Titular **NEWTON GERALDO BRETAS (Orientador)**
(Escola de Engenharia de São Carlos/USP)

APROVADO

Prof. Dr. **ALEXANDRE CLÁUDIO BOTAZZO DELBEM (Co-orientador)**
(Instituto de Ciências Matemáticas e de Computação/USP)

APROVADO

Prof. Titular **ANDRÉ CARLOS PONCE DE LEON FERREIRA DE CARVALHO**
(Instituto de Ciências Matemáticas e de Computação/USP)

Aprovado

Prof. Associado **RUBÉN AUGUSTO ROMERO LÁZARO**
(Universidade Estadual Paulista "Julio de Mesquita Filho"/Campus de Ilha Solteira)

Aprovado

Prof. Associado **RICARDO HIROSHI CALDEIRA TAKAHASHI**
(Universidade Federal de Minas Gerais/UFGM)

aprovado

Prof. Associado **GERALDO ROBERTO MARTINS DA COSTA**
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica e
Presidente da Comissão de Pós-Graduação

“Porque *dele*, e por meio *dele*, e para *ele* são
todas as coisas. A *ele*, pois, a glória eternamente.
Amém!”

Romanos 11:36

Agradecimentos

À Deus, motivo da minha existência, de tudo o que faço e por quem tudo faço. Porque “Todas as coisas foram feitas por ele, e sem ele nada do que foi feito se fez”. (João 1:3).

À minha esposa, Kelly Martins Salvatierra dos Santos, pelo apoio, companheirismo, incentivo e por toda ajuda que me concedeu nesses longos anos.

Aos meus pais, Geraldo Clemente dos Santos e Teresa Ortega dos Santos, que mesmo à distância, sempre apoiaram o trabalho e não mediram esforços para conceder-me uma educação de qualidade.

Ao professor Alexandre Cláudio Botazzo Delbem, pela competência com que me orientou no desenvolvimento desta tese e pelo tempo generosamente dedicado, transmitindo-me os melhores e mais úteis ensinamentos com paciência, lucidez e confiança, além das críticas construtivas. Agradeço também pelo incentivo e motivação à pesquisa e pela revisão de inúmeros textos científicos.

Ao meu orientador, professor Newton Geraldo Bretas, pela visão acadêmica em geral e orientações. Pelas críticas construtivas e revisão de artigos científicos, sempre sugerindo uma nova maneira de apresentá-los. E por acreditar em mim no desenvolvimento desta tese.

Ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), pelo auxílio financeiro concedido em forma de bolsa de estudos e taxa de bancada, a qual foi muito útil na aquisição de equipamentos utilizados no desenvolvimento do trabalho e no financiamento de publicações de artigos, além de possibilitar participações em eventos científicos.

Aos colegas da área Indústria, do Instituto Federal de Educação, Ciência e Tecnologia do Tocantins (IF-TO), que supriram minha ausência em sala de aula durante o período necessário para o desenvolvimento desta tese.

Aos amigos do LACO (Laboratório de Análise Computacional em Sistemas Elétricos de Potência) pelas constantes discussões construtivas e parceria em publicações originadas desta pesquisa.

Aos amigos conquistados durante esse pequeno tempo de trabalho em São Carlos-SP.

À Escola de Engenharia de São Carlos (EESC/USP) pela possibilidade de utilização de todo o espaço físico e apoio geral.

SUMÁRIO

Resumo	ix
Abstract.....	xi
Lista de Siglas e Abreviaturas	xiii
Lista de Figuras.....	xv
Lista de Tabelas	xix
1 Introdução.....	1
1.1 Apresentação do Problema.....	4
1.2 Objetivos	9
1.3 Organização do Documento.....	10
2 Revisão da Literatura.....	13
2.1 Restabelecimento em Contingência.....	14
2.2 Redução de Perdas.....	32
2.3 Outras Aplicações de Reconfiguração de Redes.....	39
3 O Problema de Reconfiguração	45
4 Representação Nó-Profundidade.....	55
4.1 Fundamentos da RNP	55
4.2 Operadores da RNP.....	57
4.2.1 Operador 1	58
4.2.2 Operador 2	61
4.3 Localizando um nó na RNP de uma floresta	63
4.4 Determinação dos nós p, r e a	64
5 Avaliação Eficiente das Soluções.....	67
5.1 Fluxo de Carga.....	67
5.1.1 Métodos de soma de potência.....	69
5.1.2 Extensão da RNP para fluxo de carga	71
5.1.3 Varredura direta/inversa com RNP.....	75
5.1.4 Exemplo de aplicação do fluxo de carga proposto	78
5.2 Cálculo do número de manobras	80
6 Algoritmos Evolutivos	89
6.1 Conceitos Básicos.....	92
6.2 Elitist Non-Dominated Sorting Genetic Algorithm.....	94
6.2.1 Método de ordenação rápida por não dominância	95
6.2.2 Preservação da diversidade	96
6.2.3 Operador comparação.....	97
6.2.4 Algoritmo principal e complexidade computacional.....	97
6.3 Strength Pareto Evolutionary Algorithm	99
6.3.1 Método de atribuição de fitness.....	100

6.3.2	<i>Seleção e truncamento</i>	101
6.3.3	<i>Algoritmo principal e complexidade computacional</i>	102
6.4	Algoritmo Evolutivo Multi-objetivo Proposto	104
6.4.1	<i>Algoritmo Evolutivo Multi-objetivo em Tabela</i>	105
6.4.2	<i>Algoritmo principal do AEMT</i>	107
7	Resultados Experimentais	109
7.1	Diversas formas de utilização dos Operadores da RNP	111
7.2	Modelagem mono-objetivo para falta única	112
7.3	Modelagem mono-objetivo para múltiplas faltas	116
7.4	Modelagem multi-objetivo para falta única	118
7.5	Modelagem multi-objetivo para múltiplas faltas	122
7.6	Resultados com sistemas construídos a partir do SDR-SC	126
7.7	Testes com o SDR da Taiwan Power Company	129
7.8	AEMT com modelos de corrente constante e potência constante para o fluxo de carga	132
7.8.1	<i>Desempenho do AEMT com FMCC e FMPC</i>	133
7.9	Comparação do AEMT com outros AEs Multiobjetivos	135
8	Conclusões	141
9	Publicações Originadas desta Pesquisa	145
10	Referências Bibliográficas	147
	Apêndice A	165

Resumo

SANTOS, A. C., Algoritmo Evolutivo Computacionalmente Eficiente para Reconfiguração de Sistemas de Distribuição. São Carlos, 2009, 186p. Tese de doutorado – Escola de Engenharia de São Carlos, Universidade de São Paulo.

O restabelecimento de energia em sistemas de distribuição de energia elétrica radiais geralmente envolve a reconfiguração de redes para restaurar eletricidade à(s) área(s) fora de serviço. As principais técnicas para restabelecimento de energia em sistemas de distribuição de grande porte têm sido os Algoritmos Evolutivos (AEs). Após a falta ter sido identificada e a zona em falta ter sido isolada do sistema, o algoritmo deve encontrar soluções em que: 1) supra com energia o maior número de consumidores possível, 2) minimize o número de operações de chaveamentos, 3) não viole restrições operacionais do sistema, 4) reduza o total de perdas resistivas, 5) a configuração da rede seja radial e, 6) obtenha tal solução em tempo real. Este projeto emprega uma nova estrutura de dados para manipular grafos produzindo exclusivamente configurações radiais e conexas, chamada Representação Nó-Profundidade (RNP), garantindo que todas as soluções potenciais geradas pelo algoritmo satisfaçam os itens (1) e (5). Além disso, propõe-se um AE utilizando a RNP capaz de encontrar planos de restabelecimento adequados para sistemas de distribuição de larga-escala, com milhares de chaves e barras, em tempo real.

Palavras Chave: Algoritmos Evolutivos, Representação Nó-profundidade, Restabelecimento de Energia, Sistema de Distribuição de grande porte.

Abstract

SANTOS, A. C., Evolutionary Algorithm Computationally Efficient for Distribution System Reconfiguration. Sao Carlos, 2009, 186p. Thesis of doctorate – Engineering School of Sao Carlos, University of Sao Paulo.

Energy Restoration in radial distribution systems usually involves the network reconfiguration to restore the electricity to the out-of-service areas. The main approaches for energy restoration in large-scale distribution systems have been the Evolutionary Algorithms (EAs). After a fault has been identified and isolated, the algorithm must find solutions that: 1) supply energy to the larger number of consumers, 2) reduce the number of switching operations, 3) respect operational constraints of the system, 4) reduce the amount of power losses, 5) generate exclusively radial configurations and 6) find solutions in real time. This work uses a new data structure, called Node-Depth Encoding (NDE), to manipulate graphs producing exclusively radial and connected configurations, and guaranteeing that all potential solutions generated by the algorithm satisfy items (1) and (5). Moreover, we propose an EA using the NDE that is capable of finding adequate restoration plans in real time for large-scale distribution systems, with thousands of switches and buses.

Keywords: Energy Restoration, Evolutionary Algorithms, Large-scale distribution systems, Node-depth Encoding.

Lista de Siglas e Abreviaturas

ACF	Algoritmo de busca baseado em Colônias de Formigas
AE	Algoritmo Evolutivo
AEMT	Algoritmo Evolutivo Multi-objetivo em Tabela
AG	Algoritmo Genético
BT	Busca Tabu
BTR	Busca Tabu Reativa
EDA	<i>Estimations of Distributions Algorithms</i>
EDHI	Evolução Diferencial Híbrida Inteiro-mista
FMCC	Fluxo de carga Modelo Corrente Constante para soma de corrente
FMPC	Fluxo de carga Modelo Potência Constante para soma de potência
GRASP	<i>Greedy Randomized Adaptive Search Procedure</i>
IA	Inteligência Artificial
MPF	Modelo Pai-Filho
NA	Normalmente Aberta
NF	Normalmente Fechada
NSGA	<i>Nondominated Sorting Genetic Algorithm</i>
PGSA	<i>Plant Growth Simulation Algorithm</i>
PM	Programação matemática
PSO	Otimização por enxames de partículas
RNP	Representação Nó-Profundidade
SA	<i>Simulated Annealing</i>
SDR	Sistema de Distribuição Radial
SDR-SC	Sistema de Distribuição da cidade de São Carlos

SE	Subestação
SEP	Sistema Elétrico de Potência
SPEA	<i>Strength Pareto Evolutionary Algorithm</i>
TPC	<i>Taiwan Power Company</i>

Lista de Figuras

Figura 1.1 – Sistema elétrico de potência, onde G é um gerador, SD um dispositivo de manobra e TR é transformador.	3
Figura 1.2 – Representação de um sistema de distribuição, onde SB é um barramento na subestação, TR um transformador e S é uma chave seccionadora.....	4
Figura 3.1 – Exemplo de um grafo.	45
Figura 3.2 – Sistema de Distribuição com 4 alimentadores.	47
Figura 3.3 – Sistema de Distribuição em falta.	48
Figura 3.4 – Setores a jusante do setor em falta desconectados do SDR.	48
Figura 3.5 – Nova configuração.	49
Figura 4.1 – SDR com três alimentadores. Grafo com três árvores.	56
Figura 4.2 – RNP para as 3 árvores da Figura 4.1.	56
Figura 4.3 – T_{de} , T_{para} e suas respectivas RNPs.	60
Figura 4.4 – T_{tmp} e sua RNP.	60
Figura 4.5 – T'_{de} e sua RNP.	60
Figura 4.6 – T'_{para} e sua RNP.	61
Figura 4.7 – T_{de} e sua RNP.	62
Figura 4.8 – Subárvores enraizadas nos nós do caminho de r a p.	62
Figura 4.9 – RNP da subárvore podada.	62
Figura 4.10 – Árvore T'_{para} e sua RNP.	63
Figura 5.1 – SDR com dois alimentadores.	71
Figura 5.2 – Agrupamento das linhas e barras em setores.	72
Figura 5.3 – Grafo representando setores do SDR da Figura 5.2.	72
Figura 5.4 – Operações necessárias para isolar o setor em falta.	81
Figura 5.5 – Cálculo das manobras: configuração inicial.	84

Figura 5.6 – Cálculo das manobras: configuração 1.....	85
Figura 5.7 – Cálculo das manobras: configuração 2.....	85
Figura 5.8 – Cálculo das manobras: configuração 3.....	86
Figura 6.1 – Opções de compra de um automóvel envolvendo custo e conforto.	93
Figura 6.2 – Cálculo da crowding distance.	96
Figura 6.3 – Esquema do modelo NSGA-II apresentado em [48]......	99
Figura 7.1 – Perdas de potencia obtidas usando os Operadores 1 e 2 separadamente, seleção aleatória e ajuste dinâmico de probabilidade de seleção de cada Operador.	112
Figura 7.2 – Destaque do alimentador 23 do SDR-SC.	115
Figura 7.3 – Alimentador 6 do SDR-SC.....	117
Figura 7.4 – Alimentador 22 do SDR-SC.	117
Figura 7.5 – Falta única, perdas totais.....	120
Figura 7.6 – Falta única, menor tensão.	120
Figura 7.7 – Falta única, carregamento da rede.	120
Figura 7.8 – Falta única, carregamento das subestações.	120
Figura 7.9 – Falta única, número de chaveamentos.....	120
Figura 7.10 – Falta única, função agregação.....	120
Figura 7.11 – Múltiplas faltas, perdas totais.....	123
Figura 7.12 – Múltiplas faltas, menor tensão.	123
Figura 7.13 – Múltiplas faltas, carregamento da rede.....	123
Figura 7.14 – Múltiplas faltas, carregamento das subestações.	123
Figura 7.15 – Múltiplas faltas, número de chaveamentos.....	123
Figura 7.16 – Múltiplas faltas, função agregação.	123
Figura 7.17 – Subpopulação, perdas totais.	124
Figura 7.18 – Subpopulação, menor tensão.....	124
Figura 7.19. – Subpopulação, carregamento da rede.....	124
Figura 7.20 – Subpopulação, carregamento das subestações.....	124

Figura 7.21 – Resultados indicando tempo de processamento sublinear para o AEMT.....	127
Figura 7.22 – Topologia Inicial do SDR da Taiwan Power Company.....	130
Figura 7.23 – Fronteiras de Pareto encontradas pelo NSGA-II nas iterações 1 e 150.....	137
Figura 7.24 – Fronteiras de Pareto encontradas pelo SPEA2 nas iterações 1 e 150.....	137
Figura 7.25 – Soluções encontradas pelo AEMT.	138
Figura 7.26 – Comparação das melhores fronteiras obtidas pelo AEMT, NSGA-II e SPEA2.	138

Lista de Tabelas

Tabela 4.1 – Lista de adjacência para os nós do grafo da Figura 4.1.....	58
Tabela 5.1 – Árvores dos setores com o nó adicional.....	74
Tabela 5.2 – RNPs dos setores mostrados na Tabela 5.1.....	75
Tabela 5.3 – Manobras de chaves: Caso 1.....	82
Tabela 5.4 – Manobras de chaves: Caso 2.....	82
Tabela 5.5 – Manobras de chaves: Caso 3.....	83
Tabela 7.1 – Média de perdas de potência nas soluções obtidas de 15 execuções do AEMT para falta única.	111
Tabela 7.2 – Média de 15 execuções do AE com RNP com modelagem mono-objetivo para falta única.	115
Tabela 7.3 – Média de 15 execuções do AE com RNP com modelagem mono-objetivo para múltiplas faltas.....	118
Tabela 7.4 – Média de 15 execuções do AEMT para falta única.	122
Tabela 7.5 – Média de 15 execuções do AEMT para múltiplas faltas.....	125
Tabela 7.6 – Dados dos Sistemas construídos a partir do SDR-SC (Sistema 1). ...	126
Tabela 7.7 – Média de Resultados de 30 execuções do AEMT.....	128
Tabela 7.8 – Tempo de processamento dos métodos que utilizam o SDR da TPC.....	132
Tabela 7.9 – Resultados médios de 30 execuções utilizando o FMCC e o FMPC.....	134
Tabela 7.10 – Menor tensão no alimentador de acordo com o modelo de carga....	134

1 Introdução

Nas últimas décadas, Sistemas Elétricos de Potência (SEP) tornaram-se mais complexos. Em um SEP, existem centenas de componentes: geradores, transformadores, linhas de transmissão, equipamentos de proteção e controle, etc. Além do aumento em escala dos SEP, a utilização crescente de cargas envolvendo eletrônica de potência e controladores baseados nessa tecnologia, bem como a ampliação dos sistemas de co-geração são características que tornam a operação dos SEP mais complexa. Nesse contexto, o desenvolvimento de ferramentas computacionais de auxílio à operação tem crescido, tornando-se cada vez mais importante.

Esta pesquisa foca na elaboração de uma ferramenta de reconfiguração para auxiliar a operação de sistemas de distribuição de energia. O desenvolvimento de ferramentas computacionais para SEP de larga-escala é relativamente complexo, uma vez que esses sistemas possuem um número relativamente alto de componentes interligados de forma esparsa, gerando o que tem sido chamado na literatura de redes complexas [1-3].

A Figura 1.1 apresenta um SEP destacando-se suas três partes principais: geração, transmissão e distribuição. A seguir, destacam-se características qualitativas e quantitativas que diferenciam as partes de um SEP:

- **Geração**, composta pelas usinas, tem a função de converter energia de alguma fonte em eletricidade. A tensão é, em geral, em torno de 13,8kV. A segunda maior hidrelétrica do mundo em potência instalada é a Usina de Itaipu com 14.000MW divididas em 20 unidades geradoras de aproximadamente 700MW cada. As 10 unidades de 60Hz que atendem o Brasil, tem tensão nominal de 18kV;
- **Transmissão**, composta basicamente por linhas de transmissão, é responsável por transportar a energia dos pontos de geração aos pontos de consumo. A tensão nesses sistemas variam, geralmente, de 69kV a 750kV. O sistema de transmissão de Itaipu é dividido em 2 grupos: o de corrente alternada que recebe a energia em 60Hz e eleva para 750kV com 3 linhas de transmissão; e o de corrente contínua, que recebe a energia em 50Hz, sua tensão é convertida através de retificadores para aproximadamente 600kV em corrente contínua e transmitida por duas linhas até Ibiúna-SP, onde é convertida para 60Hz, interligando-se ao sistema do Sudeste;
- **Distribuição**, composta basicamente por subestações abaixadoras e linhas de distribuição, é responsável por distribuir a energia elétrica recebida do sistema de transmissão aos centros consumidores, que geralmente utilizam 13,8kV. Como exemplo de um sistema de distribuição típico, pode-se citar o da cidade de Natal-RN, composto por 7 subestações, 48 alimentadores em 13,8kV que atende 220.197 consumidores.

Figura 1.1 – Sistema elétrico de potência, onde G é um gerador, SD um dispositivo de manobra e TR é transformador.

Um Sistema de Distribuição, por sua vez, pode ser subdividido em duas partes principais:

- **Sistema de Distribuição Primária:** também conhecido como distribuição de média tensão, operam geralmente em redes radiais aéreas na tensão de 13,8kV. Tem a possibilidade de transferência de blocos de cargas entre circuitos para o atendimento da operação em condições de contingência ou para manutenção preventiva e/ou corretiva. Essas redes atendem aos consumidores primários (indústrias de médio porte, conjuntos comerciais, grandes hospitais, *shopping centers*, instalações de iluminação pública, etc.) e aos transformadores de distribuição, que por sua vez, suprem as redes secundárias ou de baixa tensão;
- **Sistema de Distribuição Secundária:** também conhecido como distribuição de baixa tensão, geralmente operam com tensões 220/127V ou 380/220V. Suprem consumidores de baixa tensão, pequenos comércios e indústrias e, principalmente, os consumidores residenciais. Essa parte do sistema de distribuição usualmente não conta com recurso para o atendimento de contingências.

A Figura 1.2 ilustra um Sistema de Distribuição subdividido em rede primária e secundária. Neste trabalho o foco principal está voltado para as redes primárias de distribuição.

Figura 1.2 – Representação de um sistema de distribuição, onde SB é um barramento na subestação, TR um transformador e S é uma chave seccionadora.

1.1 Apresentação do Problema

A busca pelo fornecimento de energia elétrica com qualidade aos consumidores tem sido um alvo constante das concessionárias de energia elétrica e um dos focos da Agência Nacional de Energia Elétrica. De acordo com Cipoli em [4], essa qualidade pode ser avaliada pelos seguintes itens:

- Continuidade do fornecimento;
- Nível de tensão;
- Oscilações rápidas de tensão;
- Desequilíbrio da tensão;
- Distorções harmônicas da tensão;
- Nível de interferência em sistemas de comunicações.

Por questões de custos de investimentos e operação, as redes de distribuição primárias operam em geral com configurações radiais [5], ou seja, sem a formação de malhas (ver Figura 1.2). Assim, em redes radiais, os nós de carga recebem energia de um único nó a jusante do mesmo, simplificando a operação e proteção das redes elétricas. Por outro lado, a utilização de redes radiais diminui a confiabilidade do sistema em relação à continuidade do fornecimento de energia.

As interrupções são inevitáveis, seja em consequência de defeitos que fatalmente ocorrem na rede (manutenção corretiva), seja para a execução de serviços de manutenção preventiva. Assim, o agrupamento de vários pontos de cargas em blocos separados por chaves que operam no estado Normalmente Aberto (NA) e Normalmente Fechado (NF) foi uma solução encontrada para melhorar a confiabilidade do sistema sem incorrer em gastos excessivos. Desta forma, a operação de chaves permite a troca de carga entre circuitos em caso de interrupção em algum ponto da rede. Nessas situações, deve-se dispor de um plano para restabelecimento de energia (verificar conceito de restabelecimento no Capítulo 3), que consiste em determinar um conjunto de manobras de chaves, para que essas interrupções restrinjam-se à menor parte possível da rede. Um algoritmo de reconfiguração de redes de distribuição pode ser utilizado em diversos problemas da operação do sistema, sendo uma ferramenta importante na automação da distribuição. Em condições normais de operação, pode-se utilizar a reconfiguração de redes, através da manobra de chaves NA e NF, para reduzir as perdas totais por efeito Joule [6, 7] e/ou para o balanceamento de carga entre os alimentadores [8, 9], aliviando os alimentadores que estão com carregamento crítico. Assim, opera-se a rede também para a redução de queda de tensão [10] e para aliviar trechos da rede com sobrecarga [11] (corrente elétrica com níveis acima do suportado pelos cabos).

A reconfiguração de redes também pode ser aplicada na ocorrência de contingências [12-14], sendo o foco principal desta pesquisa. Nesse caso, é necessário determinar quais chaves NF deverão ser abertas, a fim de isolar trechos da rede com defeito. Para minimizar a quantidade de cargas fora de serviço, existe a possibilidade de conectar a outros alimentadores trechos da rede que se encontram a jusante da falta através do fechamento de chaves NA. Cargas fora de serviço correspondem aos consumidores que ficaram sem energia após o setor¹ em falta ter sido isolado, sendo que nem sempre é possível a conexão dessas cargas a outros alimentadores. Esse procedimento é mais complexo e será discutido adiante. Com o mesmo objetivo de isolar áreas em contingência, esse tipo de reconfiguração pode ser utilizado também para manutenção preventiva da rede.

Outra aplicação da reconfiguração de redes ocorre no planejamento de Sistemas de Distribuição. Com o crescimento populacional nas zonas urbanas nas últimas décadas, o consumo de energia elétrica tem aumentado significativamente, sendo necessário expandir as redes de distribuição. Essa expansão pode ser realizada de maneira planejada, definindo a topologia em que a rede irá operar no futuro, geralmente dentro de um horizonte de 5 a 10 anos [15].

Um plano genérico adequado de reconfiguração de redes pode envolver as seguintes necessidades práticas:

1. Encontrar um plano em um curto intervalo de tempo (em tempo real);
2. Minimizar o número de manobras;
3. Reduzir o número de consumidores interrompidos;
4. Nenhum componente sobrecarregado;

¹ Um setor corresponde a um conjunto de barras e linhas sem a presença de chaves seccionadoras. Neste trabalho, será utilizada a palavra setor no lugar de diversos sinônimos utilizados por outros autores (zona, seção, etc.) sempre que tiver o mesmo significado.

5. Manter a estrutura radial do sistema;
6. Reduzir o total de perdas de potência;
7. Reduzir quedas de tensão.

Dependendo da necessidade da companhia de energia elétrica, outros objetivos podem ser considerados. Com base nesses objetivos e restrições, pode-se verificar que a reconfiguração da rede de energia é um problema com múltiplas restrições e múltiplos objetivos, sendo alguns deles conflitantes.

Diversos métodos têm sido utilizados para resolver problemas de reconfiguração de redes de distribuição de energia elétrica. Os mesmos podem ser divididos em dois grandes grupos: os de Programação Matemática (PM) e os baseados em heurísticas.

Dentre os métodos de PM, destacam-se a programação inteira mista [16-19], a programação não-linear [20], a programação dinâmica [21, 22] e a programação linear [23]. Os métodos matemáticos utilizados em PM representam explicitamente as principais restrições: leis de Kirchhoff, capacidade dos cabos e equipamentos, queda de tensão, etc. Porém, a aplicação de PM está limitada a problemas de pequeno porte. Devido a grande quantidade de variáveis envolvidas na modelagem matemática por PM de problemas de reconfiguração de redes de distribuição de médio e grande portes, o problema torna-se computacionalmente intratável, mesmo para redes que não sejam grandes. Deve-se lembrar que esse problema é combinatorial no número de chaves² do sistema, isso significa que o número de soluções possíveis aumenta exponencialmente com o número de chaves.

Por essas razões, algoritmos baseados em heurísticas têm sido sugeridos como alternativas para resolver o problema de reconfiguração. Os mesmos não

² Mais especificamente, o número de soluções potenciais para o problema é dado pelo número de chaves NA (n_{NA}) vezes o número de chaves NF (n_{NF}) para 1 par de manobras. Para k pares, ter-se-iam $(n_{NA} \times n_{NF})^k$ potenciais soluções para serem investigadas por um algoritmo de busca exaustiva.

garantem que a solução encontrada seja ótima, mas permitem a determinação de soluções adequadas ou quase ótimas.

Um método pode ser chamado de heurístico quando não há conhecimentos matemáticos completos sobre seu comportamento ou utiliza conhecimento e experiência de especialistas na construção do algoritmo. Seu objetivo é resolver problemas complexos utilizando uma quantidade não muito grande de recursos, especialmente no que diz respeito ao consumo de tempo para encontrar soluções com qualidade. Algoritmos heurísticos geralmente são utilizados para resolver problemas onde não são conhecidos métodos exatos eficientes que ofereçam garantias de encontrar a solução ótima.

Ao longo dos anos, diversas pesquisas têm sido realizadas para aumento do desempenho de métodos heurísticos em problemas combinatoriais. Vários desses métodos são metaheurísticas, algoritmos baseados em um conjunto de conceitos heurísticos que definem um método aplicável a um extenso conjunto de diferentes problemas [24]. Alguns exemplos de metaheurística são: *branch exchange* [25, 26], *Simulated Annealing* (SA) [27], Busca Tabu (BT) [28, 29], Sistemas Especialistas [30], Algoritmos Evolutivos (AE) [31-36], Algoritmos baseados em Colônias de Formigas (ACF) [37], otimização por enxame de partículas (PSO) [38, 39], busca em vizinhança variável (VNS) [40-42], *Greedy Randomized Adaptive Search Procedure* (GRASP) [43], busca dispersa [44, 45], *Estimations of Distributions Algorithms* (EDAs) [46, 47], etc. Dentre esses, os AEs têm sido a metaheurística mais explorada para o problema de reconfiguração de sistemas de distribuição. Além disso, possuem um suporte relevante da literatura em problemas multiobjetivos [34, 48-50].

1.2 Objetivos

O objetivo principal deste trabalho é desenvolver um AE eficiente para resolver problemas de reconfiguração de redes de distribuição de energia elétrica de grande porte. Os seguintes problemas são considerados: (i) restabelecimento de energia, após a ocorrência de uma falta, ao maior número de consumidores possível (ou seja, isolar a área em falta e interligar setores a jusante do local em falta a outros alimentadores); (ii) encontrar configurações que minimizem a quantidade de chaves manobradas, a queda de tensão, o carregamento da rede, o carregamento das subestações e o total de perdas de potência. Mesmo no restabelecimento de energia às áreas desenergizadas, buscar-se-á configurações que também reduzam o total de perdas de potência e o número de chaves alteradas, uma vez que o número de chaves a ser alteradas influencia no tempo de restabelecimento de energia, pois a maioria das chaves é alterada manualmente pelos operadores.

Ao se utilizar sistemas de distribuição de grande porte, as possibilidades de novas configurações aumentam consideravelmente comparadas com sistemas de pequeno e médio porte. A maior quantidade de planos alternativos investigados possibilita encontrar melhores soluções. Ao se considerar mais pontos de cargas, obtêm-se soluções que correspondem a um fluxo de carga mais próximo do real para uma dada configuração. Assim, uma metodologia capaz de lidar com redes completas tende a melhorar a qualidade dos planos de restabelecimento encontrados para redes de grande porte. Para isso, é preciso o desenvolvimento de um AE capaz de lidar eficientemente com problemas de reconfiguração de grandes redes envolvendo múltiplos objetivos.

A estratégia para obter um AE eficiente é a utilização de uma estrutura de dados especial para manipulação e armazenamento de um conjunto de árvores

geradoras de grafos. Esta estrutura é chamada de Representação Nó-Profundidade (RNP) [51]. Com base nessa estrutura de dados, um AE multiobjetivo baseado no método em Tabela (ver Seção 6.4.1) é proposto, que utilizando a RNP produz exclusivamente configurações radiais em que todos os consumidores possíveis são atendidos após a área em falta ter sido isolada do sistema, isto é, sem a presença de malhas em um mesmo alimentador ou entre alimentadores diferentes. Por essas razões, o AE desenvolvido é chamado de AE Multi-objetivo em Tabela (AEMT).

O AEMT requer tempo de processamento e quantidade de memória RAM consideravelmente menores. Com isso, a metodologia proposta tem mostrado ser uma alternativa eficiente para problemas de projeto de grandes redes [51-54] e também adequada para problemas que requerem soluções em tempo real.

1.3 Organização do Documento

O presente Capítulo apresentou uma introdução à SEP, especificando o problema – reconfiguração de redes de distribuição para restabelecimento de energia e redução de perdas de potência – e especificou os objetivos da pesquisa.

O Capítulo 2 descreve os principais trabalhos publicados nas últimas décadas abordando problemas de reconfiguração de redes para restabelecimento de energia após contingências, redução de perdas de potência e outras aplicações.

O Capítulo 3 apresenta inicialmente conceitos de grafos, pois um sistema de distribuição radial pode ser visto como uma floresta geradora em que cada alimentador corresponde a uma árvore dessa floresta. Posteriormente, o Capítulo apresenta especificamente o problema de restabelecimento, por fim, a formulação matemática do problema.

O Capítulo 4 apresenta a RNP e dois Operadores para manipulação dessa representação utilizada pelo AEMT para construção de novas configurações para uma rede elétrica.

O Capítulo 5 mostra como cada configuração gerada pelos Operadores da RNP é avaliada. Detalhes do fluxo de carga utilizado para esse fim também são apresentados, bem como um método para determinar de forma eficiente o número de manobras realizadas para se obter uma dada configuração.

O Capítulo 6 introduz os conceitos principais envolvendo AEs multiobjetivos, destacando-se o NSGA-II e o SPEA2. Em seguida, a proposta do AE multiobjetivo empregado é descrita.

O Capítulo 7 apresenta os resultados experimentais para um SDR real de grande porte. Testes são realizados considerando modelos com um único objetivo e situações com falta única, até modelos com múltiplos objetivos e ocorrência de múltiplas faltas. Comparações com outros métodos também são realizadas nesse Capítulo. Testes com sistemas maiores (dezenas de milhares de chaves e barras) também são realizados para verificar o comportamento do método com o aumento da escala dos sistemas.

O Capítulo 8 apresenta as conclusões do trabalho e perspectivas futuras.

2 Revisão da Literatura

Existem diversas aplicações para problemas envolvendo reconfiguração de redes para sistemas de distribuição, dentre as quais se destacam: i) restabelecimento de energia após a ocorrência de uma contingência; ii) redução de perdas potência nas linhas; iii) planejamento de sistemas; iv) fluxo de carga; v) balanceamento de carga; vi) balanceamento de fases. Esse Capítulo revisa os principais trabalhos publicados sobre o problema de reconfiguração de redes de distribuição de energia elétrica com principal atenção para os de restabelecimento que envolvem o objetivo adicional relativo ao número de chaveamentos e a restrição de tempo de computação, que tornam o problema de reconfiguração mais complexo. Os trabalhos englobam os seguintes problemas de reconfiguração: Restabelecimento em Contingência (Seção 2.1); Redução de Perdas (Seção 2.2) e Outras Aplicações de Reconfiguração de Redes (Seção 2.3). Em cada Seção, os artigos são apresentados em ordem cronológica de publicação. Informações como tempo de processamento, linguagem computacional e ambiente utilizado na programação são apresentados sempre que a fonte possuir esses dados, uma vez que esses são relevantes para o desenvolvimento de uma técnica computacionalmente eficiente para reconfiguração de redes de larga-escala, foco desta pesquisa.

2.1 Restabelecimento em Contingência

Desde a década de 80 que vários trabalhos têm sido publicados sobre o problema de restabelecimento de energia para sistemas de distribuição após a ocorrência de uma falta. A maioria das publicações nesse tema envolve sistemas especialistas ou processos de busca heurística [55]. A análise dos trabalhos relativos a restabelecimento em contingência apresentada aqui busca observar os seguintes critérios para construção de uma visão crítica dos mesmos:

1. Características da rede utilizada nos testes: dimensão, real, simplificada, de pequeno ou grande porte, etc.;
2. Tempo de processamento requerido pela técnica de reconfiguração;
3. Estratégias para lidar com problemas multi-objetivos;
4. Procedimento para geração e avaliação das soluções, bem como as rotinas para detectar e corrigir infactibilidades³;
5. Capacidade de lidar com múltiplas faltas;
6. Restrições elétricas e operacionais consideradas na modelagem do problema de reconfiguração.

Em 1996, Curcic et al. [56] publicaram um trabalho onde artigos de restabelecimento de energia em sistemas de distribuição publicados entre 1987 e 1994 foram analisados. Nesse artigo, problemas de restabelecimento de energia em Sistemas de Distribuição Radiais (SDRs) foram revisados, no qual os objetivos e restrições comuns a vários trabalhos são apresentados pelos autores. A importância de processamento rápido é evidenciada nesse artigo, que investiga também as vantagens e desvantagens em utilizar uma topologia de rede radial. Foram analisados 19 artigos sendo classificados segundo a técnica utilizada: 7 deles

³ Aspectos que tornam uma configuração (solução) infactível.

utilizaram sistemas inteligentes [57-63], outros 7 utilizaram busca heurística [64-70], 2 utilizaram o método do gradiente efetivo dual [71, 72], 2 BT e caminho mínimo [73, 74], finalmente, 1 utilizou programação inteira binária juntamente com a técnica *branch & bound* [75]. Foram explorados também os tipos de faltas possíveis para sistemas de distribuição: nas linhas, nas barras e nos transformadores. Faltas ocorridas nos sistemas de proteção dos transformadores foram consideradas, pelos autores, como faltas nos próprios transformadores. Outra divisão para os 19 artigos está no nível de tensão utilizado nos testes, classificado pelos autores em: baixa tensão (de 110V/220V), média tensão (de 5kV a 30kV), alta tensão (de 33kV a 220kV). Dentre os 19 artigos analisados, somente 3 consideram em seus planos de restabelecimento prioridades para consumidores especiais (hospitais, centros de transfusão de sangue, grandes supermercados, etc.). Este artigo não apresenta os dados necessários para análise segundo os critérios estabelecidos no início desta Seção.

No mesmo ano, Fukuyama et al. [76] desenvolveram um Algoritmo Genético (AG) paralelo para resolver problemas de restabelecimento de energia em SDRs. Esse AG foi considerado pelos autores uma ferramenta apropriada para aplicações práticas envolvendo um balanceamento entre o tempo de processamento e o custo de *hardware*. O problema é formulado de forma a minimizar a área fora de serviço e balancear o suprimento de cargas entre as subestações após uma falta ter sido identificada e isolada. Nesse trabalho o comprimento de uma *string* (representação de uma solução em um AG, ver Capítulo 6) é igual ao número de cargas da rede e, cada posição na *string* representa uma carga a jusante ou uma fonte de potência da carga de cada posição. O trabalho utiliza os operadores convencionais de um AG, cruzamento e mutação (ver Capítulo 6), além de uma rotina para verificar

infactibilidades a cada vez que o operador cruzamento é aplicado. O modelo de injeção de corrente constante é utilizado no cálculo do fluxo de carga, assim, é necessário encontrar as tensões em todos os nós, supondo que a tensão na fonte seja conhecida. Utiliza-se para isso, o método de varredura direta/inversa [77, 78], onde as tensões nas barras são calculadas iniciando na barra em uma subestação em direção às barras terminais e os fluxos de corrente ou potência são obtidos a partir das barras terminais em direção à barra na subestação. Nesse AG, a população total é distribuída em várias sub-populações. Cada sub-população é alocada para um único processador e um AG convencional [79] é executado para cada sub-população em cada processador. A cada época (um período de gerações do AG), *strings* com a melhor adequação migram entre os processadores vizinhos.

A seguir o método é analisado segundo os critérios apresentados no início desta Seção:

1) Testes foram realizados em um sistema modelo simples de 6,6kV com 3 fontes de potência e 25 cargas. Devido a simplicidade da rede utilizada, não se pode tirar conclusões de seu desempenho para diversos tipos de rede;

2) Não foram apresentados dados do computador utilizado. O tempo de execução variou de 1,34s a 10s para uma rede de pequena dimensão;

3) A redução do número de consumidores sem energia e o balanceamento entre os transformadores são conjuntamente utilizados na função objetivo. Além disso, algumas restrições devem ser atendidas, sendo verificadas através de um fluxo de carga;

4) Indivíduos são gerados aleatoriamente e cada configuração da rede é representada por uma *string* de *bits* contendo as barras de carga. Diferente de outros trabalhos em que uma *string* contém o estado das chaves, este contém as barras de carga conectadas a uma fonte de potência. Após a geração de uma

configuração, a mesma é avaliada e, se alguma restrição deixar de ser atendida, operadores de mutação e cruzamento são aplicados para produzir modificações no indivíduo gerado com o objetivo de melhorar sua adequação;

5) Apenas a ocorrência de uma única falta foi abordada no problema;

6) Restrições de carregamento nas subestações, carregamento nas linhas, radialidade da rede foram consideradas no problema.

Em 2000, Augugliaro et al. [80] utilizaram estratégias evolutivas [81] com uma definição *fuzzy* de objetivos conflitantes para resolver o problema de restabelecimento de energia em sistemas de distribuição. Considerou-se que o estado de operação normal permite o controle remoto das chaves, de bancos de capacitores e conexões de cargas. Assim, quando uma falta permanente ocorre, podem-se executar remotamente ações para restabelecer energia nas áreas afetadas. Para lidar com múltiplos objetivos geralmente conflitantes, foram desenvolvidas estratégias baseadas em lógica *fuzzy*. Os critérios de avaliação para esse trabalho resultam em:

1) Uma rede de distribuição inicialmente malhada contendo 98 seções de carga (definido em nosso trabalho como setores), 81 barras de carga e 24 bancos de capacitores foi utilizada nos testes. Uma simplificação foi realizada localizando as barras e linhas não separadas por chaves em um só ponto. Esta rede difere das outras geralmente utilizadas em problemas de restabelecimento de energia em sistemas de distribuição devido sua natureza malhada e a utilização de bancos de capacitores para melhorar o perfil de tensão nas barras;

2) Um computador pessoal com processador Pentium de 166MHz foi utilizado nos testes. O programa foi desenvolvido em Delphi 2.0 e utilizou o tempo de processamento de 50s para executar 80 iterações da estratégia evolutiva;

3) Múltiplos objetivos foram considerados em sua formulação matemática envolvendo a minimização de perdas de potência como objetivo principal e a maximização da quantidade de cargas a ser restabelecida. As configurações geradas são avaliadas através de conjuntos *fuzzy*;

4) Nada foi apresentado sobre a geração dos indivíduos, os quais são modificados com base nos operadores convencionais de mutação e recombinação. As taxas de mutação e recombinação alteram-se durante o processo e a recombinação pode até desaparecer, restando apenas mutação, quando os indivíduos gerados aproximam-se de uma solução ótima;

5) Apenas a ocorrência de uma única falta foi abordada no problema;

6) As restrições consideradas na formulação do problema foram: permanência da rede em uma topologia radial, carregamento nas subestações, carregamento nas linhas e queda de tensão nas barras.

No mesmo ano, Hsiao e Chien [82] apresentaram um algoritmo híbrido combinando *fuzzy* com AG para resolver problemas de restabelecimento de energia. A formulação do problema contempla simultaneamente 5 objetivos a serem minimizados: total de carga deixada fora de serviço, número de chaveamentos, queda de tensão nas barras, carregamento nas linhas e carregamento nos transformadores. Restrições envolvendo estrutura radial e seqüência de operações de chaveamento (ordem em que as chaves serão abertas para isolar a área em falta e fechadas para conectar áreas fora de serviço) também são incluídas na formulação do problema. Devido a natureza imprecisa das funções objetivo, regras *fuzzy* baseadas no conhecimento prévio de um especialista são criadas e aplicadas a cada indivíduo gerado pelo AG. O indivíduo é avaliado segundo essas regras e classificado de acordo com sua adequação ao problema. Segundo o autor, a

utilização de conjuntos *fuzzy* possibilitou encontrar soluções mais satisfatórias quando múltiplos objetivos são utilizados. Testes foram realizados em um SDR da *Taiwan-Power Company* com 2 transformadores, 10 alimentadores, 102 ramos, 102 barras, 13 chaves NA e 217 chaves NF. Os testes realizados contemplaram 3 casos diferentes: uma única falta, múltiplas faltas (em um total de 5) e, múltiplas faltas deixando uma grande área fora de serviço.

1) Simplificações foram realizadas agrupando em um único ponto (setor) as barras e linhas não separadas por chaves. Apesar de o sistema ser baseado em um SDR real com um número razoável de chaves (230), as simplificações podem comprometer os resultados, pois podem afetar diretamente as restrições operacionais do sistema;

2) O método foi desenvolvido em linguagem computacional C++ e o tempo de processamento médio para uma rede de médio porte foi 49s utilizando um Processador Pentium-CELERON 300A PC;

3) O método avalia cada objetivo separadamente através de conjuntos *fuzzy*. Essa forma de avaliação é interessante, pois para cada geração do algoritmo, o grau de adequação do indivíduo a cada objetivo do problema é avaliado. Assim, é possível tratar cada objetivo separadamente agregando-os através de conjuntos *fuzzy*, não dependendo de uma única função ponderada. Por outro lado, é necessário determinar os parâmetros para cada conjunto, pois podem variar quando diferentes SDRs são utilizados e também em testes com objetivos diferentes;

4) Os primeiros indivíduos são gerados aleatoriamente através da seleção de chaves para serem abertas ou fechadas dentro de uma *string* contendo os estados de todas as chaves da rede. O grande problema é a geração de configurações infactíveis como trechos desconexos e/ou com ciclos ou conexão de

um alimentador com outro. Uma rotina para detectar e corrigir essas infactibilidades é aplicada a cada novo indivíduo gerado. Os descendentes são gerados por um AG, em seguida, os operadores convencionais de reprodução, mutação e cruzamento são aplicados. A rotina para verificar e corrigir infactibilidades é aplicada novamente após a aplicação dos operadores;

5) Múltiplas faltas são abordadas nos testes, sendo esse tema bem explorado no artigo, tratando diferentes aspectos envolvidos no retorno da conexão dos setores fora de serviço ao restante do sistema. Foram realizadas de 1 até 5 faltas simultâneas mostrando com clareza as dificuldades enfrentadas por algoritmos ao lidar com problemas de restabelecimento;

6) Praticamente todas as restrições operacionais são consideradas na formulação do problema, incluindo queda de tensão nas barras, carregamento nas linhas e carregamento nos transformadores aumentando significativamente o grau de confiabilidade da técnica.

Em 2002, Toune et al. [55] apresentaram um estudo comparativo entre 4 algoritmos heurísticos utilizados em restabelecimento de energia em SDRs: BT, busca tabu reativa (BTR), SA e AGs. Na formulação do problema foram considerados: fonte de potência formulada como fonte de injeção de corrente, a tensão na fonte é conhecida, utiliza o conceito de setor, cada setor concentra um ponto de carga e utiliza-se o modelo de corrente constante. O objetivo é encontrar planos de reconfiguração que restabeleça energia ao maior número de consumidores possível após a ocorrência de uma falta. A representação de variável de estado (pontos de busca) é verificada na formulação geral de um problema envolvendo esses algoritmos. Um algoritmo específico é utilizado para verificar se alguma restrição foi violada. Caso seja positivo, alterações são realizadas na

configuração a fim de torná-la factível. Segundo os autores, para restabelecimento de energia não se pode afirmar qual estratégia seria mais efetiva, pois AGs podem realizar tanto busca local quanto global, enquanto SA, BT e BTR executam somente busca local. Por outro lado, AGs podem gerar muitas configurações infactíveis elevando o tempo de processamento. AGs e SA podem visitar soluções já encontradas anteriormente, enquanto BT e BTR proíbem essas visitas, aumentando a eficiência da busca. Quando BT gera ciclos de busca, BTR pode gerar soluções muito melhores. Portanto pode tomar caminhos bem diferentes dependendo da trajetória da modificação realizada. Uma análise para o AG utilizado neste artigo segundo os critérios estabelecidos no início dessa Seção é realizada a seguir:

- 1) Um SDR de 6,6kV foi utilizado nos testes. Sua dimensão foi variada de 18 até 60 setores;
- 2) Não foram disponibilizados dados referentes ao computador utilizado. O tempo de processamento para o maior caso (60 setores) foi de 180s;
- 3) Trabalho com o objetivo de restabelecer energia ao maior número possível de consumidores, atendendo algumas restrições operacionais do sistema. Não foram disponibilizados maiores detalhes da técnica;
- 4) Operadores de cruzamento e mutação foram utilizados para gerar uma nova solução. As configurações a serem alteradas são selecionadas através do método da roleta. Para evitar a geração de muitas configurações infactíveis, as chaves próximas umas das outras são escolhidas para serem alteradas;
- 5) Múltiplas faltas foram consideradas na formulação do problema, inclusive acontecendo em um transformador, isolando vários alimentadores do mesmo;
- 6) As restrições consideradas na formulação do problema foram: rede radial, queda de tensão, carregamento nos transformadores e carregamento nas linhas.

A modelagem matemática dos outros algoritmos explorados, os operadores utilizados e resultados de simulação, além de uma análise qualitativa e quantitativa podem ser verificados no próprio artigo.

Em 2004, Shin et al. [83] apresentaram um método híbrido para restabelecimento de energia e reconfiguração ótima de redes combinando AGs e BT. No referido trabalho, uma configuração é considerada ótima quando o plano de restabelecimento encontra as perdas mínimas e atende restrições operacionais do sistema mantendo a rede radial. Algumas regras foram aplicadas ao calcular o custo total de investimento: nenhum alimentador com sobrecarga, fluxo de potência do SDR é alterado pelo serviço de restabelecimento, a estrutura radial deve ser mantida, o número de chaveamentos deve ser minimizado e a nova configuração deve ser próxima da configuração original, ou seja, somente as chaves para conectar a área fora de serviço devem ser operadas se com isso as restrições são respeitadas. O método procura trabalhar com as propriedades que AGs e BT têm de melhor, compondo o método denominado AG-Tabu. Enquanto o AG encontra resultados satisfatórios em buscas globais; a BT é mais adequada para buscas locais. Testes para minimização de uma função objetivo contemplam dois casos: um com o custo total envolvendo a soma das perdas de potência e o custo de interrupção e outro somente o custo de perdas. Em todos os testes, tanto o AG quanto o AG-Tabu encontraram a mesma solução, porém este encontrou soluções ótimas em menos iterações. O custo das mínimas perdas encontradas é apresentado em dólar (\$) ao longo das iterações; enquanto o custo de interrupção é dado em dólar por quilowatt (\$/kW) ao longo do tempo em que o serviço ficou interrompido.

1) Um SDR com 7 alimentadores, 38 barras foi utilizado nos testes. Essa rede simula uma rede de distribuição real com vários alimentadores, porém de pequena dimensão. Um destaque desse trabalho está no fato das cargas serem mapeadas segundo o tipo de consumidor: residencial, pequenos usuários e comercial. O tipo de consumidor influencia diretamente no custo tanto de perdas de potência, quanto no custo total;

2) O computador utilizado nas simulações e o tempo de processamento não foram discriminados;

3) A função objetivo avalia o custo de perdas de potência e o custo total após a interrupção e reconexão do sistema devido a ocorrência de uma falta. Apesar da busca por um indivíduo com menor número de chaves alteradas estar implícita, o que consta na função objetivo é o custo para reduzir as perdas ou o custo total para proceder a alteração na topologia. As restrições operacionais e físicas do sistema são consideradas na avaliação da configuração;

4) Nada foi apresentado sobre a geração dos indivíduos. Porém, o AG-Tabu utiliza as soluções presentes na lista tabu para ajustar a probabilidade de mutação do AG. Dessa forma, espera-se que a convergência do AG-Tabu seja mais rápida, em um espaço de busca global, do que um AG convencional. Os operadores de reprodução e cruzamento do AG são aplicados para gerar uma nova configuração. Em seguida, verifica-se a existência de genes na lista tabu, caso positivo, realiza-se a mutação com taxa baseada em indivíduos presentes na lista tabu; caso contrário, atualiza-se a configuração;

5) Este artigo contempla a ocorrência de uma única falta;

6) As restrições de carregamento nos transformadores, carregamento nas linhas, radialidade da rede são atendidas na formulação do problema.

Em 2005, Delbem et al. [32] propuseram uma codificação para SDRs baseada em cadeias de grafos buscando melhorias no desempenho dos AEs. Operadores de reprodução não convencionais utilizando a codificação proposta foram desenvolvidos com o objetivo de gerar novas configurações (soluções) a partir de uma existente. A técnica foi aplicada a problemas de reconfiguração de redes sendo testada em restabelecimento de energia, redução de perdas de potência e planejamento de SDRs. Utilizando conceitos de grafos e, partindo do princípio que uma árvore de grafo pode ser representada por “cadeias” conectando a raiz as folhas (ver Capítulo 3), o conjunto de todas essas cadeias (denominadas cadeias principais) armazenadas adequadamente representa um alimentador de um SDR. Portanto, o conjunto de todos os alimentadores representa um sistema completo.

1) Um SDR de grande porte com 1.471 barras, 249 chaves, 3 subestações e 23 alimentadores foi utilizado no teste envolvendo restabelecimento de energia. Esse SDR é um dos maiores encontrados na literatura sendo capaz de representar um sistema real;

2) Um computador Pentium III com processador de 450MHz foi utilizado nos testes. O tempo de processamento para avaliar 4.000 configurações foi de aproximadamente 4s, possibilitando utilizar a metodologia proposta em problemas que requerem solução em tempo real;

3) A técnica proposta lida com problemas multi-objetivos através de sub-populações (conhecido como método em Tabelas [84]), em que a população consiste em um conjunto de sub-populações ou tabelas. Cada tabela corresponde a uma característica do sistema a ser minimizada (carregamento da rede e transformadores, perdas de potência, queda de tensão e função ponderada). Para a tabela de minimização de número de manobras, cada linha da tabela armazena o

melhor indivíduo com um determinado número de pares de manobras. Por exemplo, na primeira linha estão os melhores indivíduos com 1 par de manobras, na segunda linha com 2 pares e assim sucessivamente. Se um indivíduo é gerado com adequação melhor que qualquer indivíduo presente em cada tabela, o pior é substituído por esse indivíduo;

4) A partir de um indivíduo (configuração radial inicial) outros são criados aplicando-se um dos operadores. Para as tabelas com perfil de tensão, carregamento na rede e carregamento nos transformadores os 3 melhores indivíduos são armazenados. A tabela com a função ponderada armazena os 12 melhores indivíduos. Novos indivíduos são gerados a cada nova configuração através de um dos operadores escolhidos aleatoriamente. Uma grande vantagem do método consiste na não necessidade de rotinas para verificar nem corrigir infactibilidades relacionadas à radialidade da rede, pois somente configurações radiais são geradas pelos operadores propostos. Outras restrições são verificadas através de um fluxo de carga específico para SDRs;

5) O artigo trata de uma falta por vez. Foram testados com faltas em setores críticos da rede, por exemplo, que isolem todo um alimentador;

6) Restrições operacionais do sistema consideradas na formulação do problema foram: queda de tensão, carregamento nas linhas, carregamento nos transformadores. Além das tabelas com cada objetivo e restrições separadamente, uma tabela a mais foi criada contendo uma função ponderada [48] com as perdas de potência, pares de manobras, além das restrições inseridas como penalidades. A restrição de radialidade da rede é sempre garantida através dos operadores propostos.

Em 2006, Inagaki [12] resolveram o problema de restabelecimento de energia utilizando AGs. Os AGs são utilizados para encontrar soluções pertencentes ao conjunto de Pareto, que consiste de soluções que não são dominadas⁴ por nenhuma outra solução [48], disponibilizando um maior número de configurações possíveis para que o operador decida qual se adapta melhor. Uma combinação de AG com SA é realizada a fim de melhorar a precisão das soluções. A população total é dividida em duas sub-populações e o total de cargas sem energia após uma falta e o total de chaves utilizadas no restabelecimento são minimizados em cada sub-população respectivamente.

1) Uma rede de dimensão pequena é utilizada nos testes com 4 transformadores, 6 alimentadores, 78 barras de carga;

2) O computador utilizado nas simulações e o tempo de processamento não foram discriminados;

3) Utiliza dois objetivos bem definidos: reduzir a área fora de serviço após uma falta e o número de operações de chaveamentos. Busca configurações que minimizem esses dois objetivos através da fronteira de Pareto. Portanto, as funções objetivo não se alteram ao serem aplicadas a outras redes. Os objetivos priorizam consumidores como hospitais, *shopping centers*, etc;

4) Indivíduos são gerados e modificados através de AGs convencionais em que cada configuração é representada por uma *string* contendo o estado de todas as chaves do SDR. Operadores convencionais dos AGs são utilizados para produzirem descendentes. O operador de mutação é aplicado seguindo algumas heurísticas para evitar que o método fique preso em ótimos locais. O número total de indivíduos gerados é 1.000 e, a cada 50 indivíduos gerados pelos AGs, aplica-se SA para

⁴ A solução **x** é dita dominar **y** se **x** é melhor ou igual a **y** em todos os atributos e, estritamente melhor em pelo menos um atributo.

melhorar os resultados através de um operador semelhante a mutação. Como configurações ineficazes podem ser geradas, uma rotina para verificar e corrigir tais ineficazibilidades (ciclos), caso existam, é aplicada.

5) Testes foram realizados contemplando uma única falta;

6) As seguintes restrições convencionais de SDRs são consideradas na formulação deste método: rede radial, a energia deve ser restabelecida as áreas a jusante do setor em falta, carregamento nas linhas, carregamento nos transformadores e queda de tensão.

Em 2008, Kumar et al. [13] apresentaram um algoritmo para resolver problemas de otimização multi-objetivo aplicado ao restabelecimento de energia em SDRs. A metodologia foi baseada em um algoritmo desenvolvido em 2000 por Deb et al. [85] denominado *Nondominated Sorting Genetic Algorithm II – NSGA-II*. As regras de tomada de decisão contemplam a presença de consumidores prioritários e a presença de chaves controladas remota e manualmente. Algumas restrições e considerações são realizadas partindo do princípio que houve uma falta: (1) a energia deve ser restabelecida ao maior número possível de consumidores sem energia; (2) do ponto de vista econômico, deve-se considerar um plano de restabelecimento em que as perdas de potência sejam mínimas; (3) a topologia radial da rede deve permanecer após o restabelecimento de energia a área fora de serviço; (4) restrições de operação das linhas e dos transformadores devem ser atendidas; (5) buscar configurações que utilize o menor número de chaveamentos possível; (6) buscar configurações que atenda as cargas prioritárias como hospitais e grandes indústrias; (7) procurar apresentar os planos de restabelecimento no menor tempo possível. Para cada indivíduo gerado uma busca em largura [86] é realizada para determinar se a configuração é radial. Quando um mesmo nó é

visitado pela segunda vez significa que existe um ciclo na configuração, portanto uma chave NF deste ciclo é alterada para NA mantendo a rede radial. Operadores convencionais de AGs são utilizados para produzir novas soluções através de cruzamento e mutação. Resultados de testes do método proposto, denominado NSGA-II avançado foram comparados com resultados do NSGA-II básico e AG tradicional. As seguintes características foram comparadas: capacidade de reduzir perdas, capacidade de restabelecer energia ao maior número de consumidores, tempo de processamento e a capacidade de restabelecer energia primeiramente às cargas prioritárias. A técnica desenvolvida conseguiu reproduzir os resultados encontrados pelas outras com tempo de processamento reduzido. As taxas de cruzamento e mutação são diferentes para cada sistema, dependendo da experiência do programador. A diferença no tempo de processamento deve-se a implementação do NSGA-II conforme o trabalho de Jensen em 2003 [87], que conseguiu melhor tempo de processamento devido a estrutura de dados adotada. Assim, somente o tempo de processamento é alterado quando comparado com o NSGA-II básico.

1) A rede é mapeada segundo um grafo com ramos e nós representando, respectivamente, chaves fechadas e barras do mesmo. Quatro redes de distribuição são utilizadas nos testes, todas radiais de pequeno porte. As dimensões variam de 13 barras e 10 chaves até 173 barras e 75 chaves;

2) O computador utilizado nos testes e o número de indivíduos gerados não foram especificados. O processo é encerrado quando todos os indivíduos permanecerem inalterados de uma geração para outra. O tempo de processamento variou de 4,38s para a rede com 13 barras até 700,4s para a rede com 173 barras;

3) A maneira como a metodologia lida com problemas multi-objetivo é relevante. Utiliza o NSGA-II para agrupar as melhores soluções em fronteiras em que não se pode afirmar qual a melhor solução, pois uma solução não é dominada por outras. A vantagem do NSGA-II em relação ao método da soma ponderada é a possibilidade de ser generalizada para praticamente qualquer SDR através da fronteira de Pareto [48, 79] não havendo necessidade de pesos diferentes para cada rede testada. A solução final é selecionada dentre aquelas pertencentes à primeira fronteira de Pareto através de critérios de prioridades estabelecidos pelo engenheiro, tornando-se uma desvantagem do método, uma vez que cada rede tem suas particularidades. As prioridades utilizadas nos testes deste trabalho foram: (i) área fora de serviço; (ii) chaves controladas remotamente; (iii) chaves controladas manualmente; (iv) redução de perdas de potência;

4) Uma configuração é representada por um *string* de *bits*, representando todas as chaves do sistema em que chave aberta é 0 e chave fechada é 1 assim como em AGs tradicionais. A população inicial é gerada aleatoriamente colocando-se 0 ou 1 para cada posição da *string*. Duas regras devem ser observadas ao gerar um indivíduo: (i) a barra raiz (barra que se encontra em uma subestação) deve estar conectada ao restante do sistema, portanto a chave que conecta a barra raiz deve estar fechada; (ii) as chaves ao redor de um setor em falta são sempre geradas com o estado 0, ou seja, abertas. Assim, indivíduos podem ser gerados com ciclos e áreas desconexas. A geração de novos indivíduos é realizada por operadores convencionais de cruzamento de um ponto e mutação. A desvantagem desta técnica está na geração e alteração dos indivíduos, pois necessita de rotinas que demandam um grande esforço computacional para identificar ciclos e eliminá-los. É muito provável que não consiga lidar com problemas de grande porte, pois poderia

incorrer no problema de explosão combinatorial ao gerar muitos ciclos e áreas desconexas, elevando consideravelmente o tempo de processamento. A avaliação é realizada através de um fluxo de carga de varredura direta/inversa [88];

5) A metodologia é aplicada a problemas com uma única falta e também com múltiplas faltas, 2, 3 e 4 faltas simultâneas. Múltiplas faltas são comuns em sistemas reais de distribuição e aumenta significativamente a complexidade do problema, destacando-se como uma vantagem do método;

6) Apesar do carregamento da rede e queda de tensão nas linhas serem consideradas na formulação do problema, o carregamento nos transformadores não é, sendo ilimitado o suprimento de potência às cargas. Como havia apenas uma subestação nos SDRs abordados nos testes, a relevância do carregamento no transformador é menor. Porém, em sistemas com mais de um alimentador, é de extrema importância que o carregamento nos transformadores seja verificado, pois os chaveamentos podem produzir transferências de cargas de vários alimentadores para um mesmo alimentador sobrecarregando o transformador desse alimentador.

Também em 2008, Men-Shen Tsai [14] utilizou sistemas inteligentes para resolver problemas de restabelecimento de energia em SDRs. Foram criadas regras utilizando programação orientada a objeto [89]. Variações de cargas durante períodos do dia são consideradas tornando os planos de restabelecimentos mais confiáveis. Deve-se haver um conhecimento prévio do sistema, bem como o perfil de cargas para cada hora do dia. Múltiplas faltas também são consideradas e vários planos de restabelecimentos são apresentados, ficando a critério do operador a escolha do mais adequado em cada situação.

1) Um SDR da *Taiwan Power Company* foi utilizado nos testes e consta de 6 alimentadores, 50 setores, 44 chaves NF e 9 chaves NA. Não foram disponibilizados

maiores detalhes dos setores, o que provavelmente consta de uma simplificação para o sistema, onde todas as barras de cargas e linhas não separadas por chaves são agrupadas em um único ponto. Apesar de utilizar um sistema mais próximo de um SDR real, a simplificação diminui a confiabilidade dos resultados, pois o fluxo de carga real poderá não ser adequadamente aproximado;

2) Detalhes do tempo de processamento não foram especificados;

3) Como em todos os problemas de restabelecimento de energia, este também possui múltiplos objetivos. Porém, o algoritmo busca configurações que restabeleçam energia ao maior número possível de consumidores com poucas operações de chaveamento. As perdas não são consideradas na avaliação da geração e configurações que violem restrições operacionais são descartadas;

4) Após a ocorrência de uma falta a mesma deve ser isolada e a área fora de serviço restabelecida se possível. Várias regras foram desenvolvidas através de programação orientada a objeto oferecendo planos de restabelecimento dessas áreas. O artigo concentra-se na investigação de formas como trechos fora de serviço podem ser re-energizados. Os setores fora de serviço são agrupados em zonas, onde cada zona contém os setores conectados por chaves fechadas. A rotina de restabelecimento é realizada para cada zona separadamente. Se a configuração violar restrições operacionais, aplica-se uma rotina para transferência de cargas entre alimentadores;

5) Múltiplas faltas são tratadas isoladamente aplicando-se as mesmas rotinas utilizadas para uma única falta;

6) Capacidade dos transformadores e redes não foram declarados.

2.2 Redução de Perdas

Uma análise dos trabalhos de redução de perdas de potência em SDRs é apresentada baseada nos seguintes itens:

1. Rede utilizada nos testes (dimensão, real, simplificada, de pequeno ou grande porte, etc.);
2. Computador utilizado e tempo de processamento;
3. Geração e avaliação dos indivíduos, rotinas para detectar e corrigir infactibilidades;
4. Restrições elétricas e operacionais do sistema.

Em 1975 Merlin e Back [90] publicaram o primeiro modelo de reconfiguração de redes de distribuição para a minimização de perdas de potência. Alterações no estado das chaves de seccionamento foram realizadas utilizando técnicas de otimização em conjunto com heurísticas. A metodologia proposta parte da rede com todas as chaves inicialmente fechadas e, em cada passo, seleciona-se convenientemente chaves da rede para serem abertas. O processo continua até que a rede se torne radial. Segundo os autores, o fato de inicialmente todas as chaves estarem fechadas garante que a solução final independe do estado inicial do sistema. Em cada passo do problema um fluxo de potência é executado na rede em malha e a chave com o menor fluxo passante, que não torne a rede desconexa, é selecionada para ser aberta. Em geral, esse método determina a configuração de rede com as perdas mínimas, ou quase-mínimas, baseado em uma variação especial de uma técnica conhecida por *branch-and-bound*. Deve-se notar que este método de construção de uma solução não leva em consideração outras restrições impostas pela rede elétrica como, por exemplo, carregamento da rede e queda de tensão.

Em 2002, Zhu [91] apresentou um AG para minimização de perdas de potência em SDRs. O mesmo é denominado AG refinado devido algumas alterações na codificação do cromossomo, na função adequação e no padrão de mutação do AG proposto por Nara et al. em 1992 [92]. Melhorias relativas à codificação do cromossomo foram realizadas no AG proposto por Nara et al. Uma configuração factível era representada por uma *string* contendo o estado de todas as chaves da rede. Agora, a mesma configuração é representada por uma *string* contendo somente o número das chaves NA representadas por números binários. Esse procedimento possibilitou reduzir significativamente a memória RAM utilizada para armazenar as configurações no computador possibilitando trabalhar com sistemas maiores.

A fim de reduzir o esforço computacional, aproximações na representação das perdas de potência foram consideradas na função objetivo do AG proposto em [92], aumentando o risco de fornecer soluções duvidosas. Por outro lado, no AG proposto em [91], as perdas de potência e penalidades para configurações que violem o carregamento máximo das linhas e o nível de tensão nas barras foram consideradas na formulação do problema. Uma melhoria significativa refere-se à mutação adaptativa onde sua taxa varia com o número de gerações obtidas, ou seja, quanto mais o processo evolutivo se aproxima do final, menor é a taxa de mutação. Segundo os autores, esse procedimento pode evitar a convergência prematura do AG para soluções ruins. O método da roleta é utilizado para selecionar uma configuração para reprodução. Uma análise segundo os critérios estipulados no início desta Seção é apresentada a seguir:

- 1) Dois pequenos SDRs foram utilizados nos testes, um de 16 barras e 3 alimentadores e outro de 33 barras e 1 alimentador. A tensão base é de 12,66kV;

2) Dados referentes ao tempo de processamento e computador utilizado na simulação computacional não foram apresentados;

3) Não foram disponibilizados dados referentes a tamanho da população, geração da população inicial e correção das configurações ineficazes. Cada nova configuração é avaliada segundo uma função objetivo contendo as perdas de potência. As restrições operacionais do sistema são inseridas na função objetivo através de fatores penalidades quando as mesmas são violadas. Operadores convencionais foram utilizados diferindo apenas na taxa de mutação adaptativa;

4) Restrições de carregamento nas linhas, níveis de tensão (máximo e mínimo), leis de *Kirchhoff* da tensão e da corrente e radialidade do sistema foram considerados na formulação do problema.

Em 2007, Siti et al. [9] introduziram um método heurístico para balanceamento de cargas e balanceamento de fases em SDRs. Ao mesmo tempo em que o balanceamento de cargas e fases do sistema é realizado, uma rede neural artificial do tipo *backpropagation* é utilizada para reduzir perdas de potência através de reconfiguração de redes. O seguinte princípio foi utilizado: se o alimentador e as cargas/fases estão balanceados, existe uma tendência de se obter as mínimas perdas de potência.

1) Testes foram realizados com 15 e 45 cargas sendo selecionadas em 3 diferentes períodos do dia dos mesmos consumidores utilizados no treinamento da rede neural;

2) Um computador com processador Intel Celeron de 1,9 GHz, 256 MB de memória RAM foi utilizado nos testes. Simulação computacional foi realizada utilizando o *toolbox* de redes neurais do Matlab 6 [93]. O tempo de processamento para 15 cargas foi de 0,17s e para 45 cargas de 0,49s;

3) A rede neural foi treinada utilizando um conjunto de 500 dados históricos reais selecionados aleatoriamente de um trecho de um SDR de uma cidade da África do Sul. Esse conjunto de dados reais consistiu de cargas desbalanceadas selecionadas em diferentes períodos do dia durante um mês. A entrada para as redes testadas consiste de 15 e 45 neurônios e as saídas são seqüências de fechamento de chaves/fases (1, 2, 3) para cada carga;

4) Restrições de carregamento nas linhas e tensão dentro de limites permissíveis (máximo e mínimo) são considerados na formulação do problema.

Em 2008, Wang et al. [94] apresentaram um método de otimização para configuração de redes baseado em “*plant growth simulation algorithm* - PGSA” para minimizar o total de perdas de potência na rede. O PGSA caracteriza o mecanismo de crescimento de plantas e determina a possibilidade de novos ramos crescerem em diferentes nós de uma planta de acordo com mudanças na função objetivo. Assim, o modelo simula o processo de crescimento de uma planta em direção a fonte de luz (ótimo global). Segundo os autores, a não necessidade de nenhum parâmetro externo como fatores de barreiras, taxas de cruzamento e mutação etc., é a principal vantagem do método em relação a algoritmos baseados em modificações aleatórias utilizados até então. Muitos trabalhos consideram o estado das chaves como variáveis de decisão. Neste, os ciclos independentes são as variáveis de decisão, evitando o problema de explosão combinatorial causado pelo aumento de chaves em grandes SDRs. Mesmo com a redução do espaço de busca ao considerar os ciclos independentes ao invés de todas as chaves da rede, configurações infactíveis ainda podem ser produzidas. As seguintes regras foram criadas para diminuir a produção de configurações infactíveis: i) chaves essenciais deverão permanecer sempre fechadas, ii) do conjunto de chaves comuns a dois

ciclos diferentes, apenas uma deverá ser aberta. Porém, esse método requer conhecimento prévio do circuito, o que restringe sua utilização. Resultados encontrados pelo PGSA para o maior SDR testado foram idênticos aos resultados encontrados por outros métodos: evolução diferencial híbrida inteiro-mista melhorada [95], SA, AG, e algoritmo de busca baseado em colônia de formigas [96]. O tempo de processamento do PGSA foi razoavelmente menor que os 3 últimos métodos e maior que evolução diferencial híbrida.

1) Testes foram realizados em dois SDRs, um de 12,66kV, 33 barras, 1 alimentador, 32 chaves NF e 5 chaves NA e o sistema de *Taiwan Power Company* de 11,4kV, com 84 barras, 11 alimentadores, 83 chaves NF e 13 chaves NA;

2) Um programa computacional foi desenvolvido em Visual C++ e executado em um computador com processador de 2,2 GHz e 256 MB de memória RAM. O tempo de processamento para encontrar uma boa solução foi de 0,5s para o SDR de 33 barras e de 113,25s para o SDR de 84 barras;

3) O processo inicia-se com o sistema radial, ou seja, uma chave aberta em cada ciclo independente, que é identificado ao se fechar uma chave. O conjunto de soluções para o ciclo 1 é composto de todas as chaves do mesmo, onde cada configuração para esse ciclo consta de uma chave aberta e todas as outras fechadas. O conjunto de soluções possíveis para todos os ciclos são as possibilidades de reconfiguração da rede;

4) Restrições de carregamento nas linhas, queda de tensão, radialidade da rede e todos os consumidores atendidos foram considerados na formulação do problema.

Também em 2008, Enacheanu et al. [97] apresentaram um AG para minimizar perdas de potência através de reconfiguração de redes. O trabalho

concentra esforços na utilização de operadores genéticos a fim de melhorar a habilidade de exploração do espaço de busca. O método utiliza grafos e teorias de *matroid* para melhorar o desempenho dos operadores convencionais de AGs. O grupo $(A, B) = (\text{conjunto de arestas de um grafo, o conjunto de todas as árvores geradoras desse grafo})$ é considerado uma *matroid* [98]. O método gera somente configurações factíveis para SDRs. Resultados de simulação computacional foram comparados com outros trabalhos da literatura.

1) Três SDRs foram utilizados nos testes. O primeiro encontra-se em Civanlar [99] e consta de 17 barras, 1 alimentador, o segundo é o SDR de Baran [100] com 33 barras e 1 alimentador; o terceiro é o SDR de Das [101] com 70 barras divididas em 2 alimentadores;

2) O AG foi desenvolvido em Matlab e as simulações computacionais foram executadas em um computador pessoal com processador Pentium IV, 1,6 GHz, 512MB de memória RAM. O tempo de processamento para as redes de Civanlar, Baran e Das foram 2,1s, 7,2s e 160s respectivamente.

3) Uma configuração radial é representada por um cromossomo contendo o número das chaves abertas. A população inicial é gerada a partir de uma configuração factível aplicando-se operador de mutação que altera uma posição na *string*. Operador de mutação gera sempre configurações radiais, pois se ao alterar uma chave na *string* um ciclo for gerado (uma busca em profundidade [86] é executada), outra chave desse ciclo deverá ser aberta. O operador de cruzamento utiliza teoria de *matroid* [98] para produzir sempre configurações factíveis, reduzindo o tempo de processamento, pois não necessita de rotina extra para localizar ciclos. O método utiliza elitismo e a seleção dos indivíduos para gerar novos indivíduos é realizada através de torneio ou roleta [48];

4) As restrições utilizadas neste trabalho segue o padrão de outros métodos: carregamento nas linhas, níveis de tensão (máximo e mínimo) determinados através de um fluxo de potência. A radialidade da rede é garantida pelos operadores utilizados através de teoria de *matroid*.

Ainda em 2008, Zhang et al. [102] desenvolveram um algoritmo de reconfiguração de SDRs para redução de perdas de potência combinado com controle de capacitor. Um AG adaptativo melhorado foi desenvolvido de modo a satisfazer diversas restrições, modificando o AG adaptativo apresentado em [103]. Melhorias foram atingidas ao mudar o tamanho do passo de busca adaptativamente e ao empregar uma estratégia de mutação aritmética para codificação binária. Para encontrar uma estrutura de rede ótima para cada instância genética, o controle de chaveamentos de capacitor e uma troca de ramos simplificada são realizados em cada iteração do algoritmo. O tamanho do passo de busca muda de acordo com a adequação encontrada em cada instância. Assim, se a adequação é alta (indica boa instância), somente uma busca local será realizada. Se a adequação é baixa (indica instância ruim), uma busca global é necessária, buscando evitar uma convergência prematura. O método foi testado em diferentes casos: *i*) somente otimização de capacitor, *ii*) somente reconfiguração de rede, *iii*) primeiro otimização de capacitor e depois reconfiguração de rede, *iv*) primeiro reconfiguração de rede e depois otimização de capacitor por chaveamentos, *v*) alternadamente em cada iteração, reconfiguração de redes e controle de capacitor (algoritmo de otimização conjunto), *vi*) utilizando o algoritmo de otimização conjunto, porém com AG adaptativo para otimização de capacitores e, *vii*) utilizando o algoritmo de otimização conjunto com o AG adaptativo melhorado para otimização de capacitor. Uma considerável redução

de perdas foi alcançada com o algoritmo conjunto quando comparada aos métodos de reconfiguração de redes e chaveamentos de capacitor separadamente.

1) O SDR utilizado nos testes consta de 119 barras, 118 ramos com chaves NF e 15 ramos com chaves NA divididas em 4 alimentadores;

2) O algoritmo foi desenvolvido utilizando um pacote de programas e executado em um sistema de computação PIII 866MHz/128MB. O tempo de processamento para o caso (vii) (utilizando o algoritmo de otimização proposto em conjunto com o AG adaptativo melhorado) foi de 28,39s;

3) Codificação binária é utilizada na representação do SDR e o número máximo de gerações é utilizado como critério de parada;

4) Restrições utilizadas na formulação matemática foram: queda de tensão, carregamento nas linhas e atendimento das leis de *Kirchoff* da corrente e da tensão (determinadas através de um fluxo de carga). A restrição de rede radial também é observada. As restrições de queda de tensão e carregamento nas linhas são inseridas na função objetivo através de penalidades.

2.3 Outras Aplicações de Reconfiguração de Redes

Muitos algoritmos têm investigado resolução de problemas envolvendo reconfiguração de redes. A seguir alguns trabalhos relevantes são apresentados, principalmente os que utilizam técnicas de inteligência artificial.

Em 2004, Gohokar et al. [104] propuseram uma formulação para problemas de reconfiguração de SDRs através da topologia de rede. A proposta visa facilitar a obtenção de uma solução representando SDRs através de matrizes incidência que representam diretamente a topologia da rede. A aplicação do algoritmo proposto é genérica e pode ser utilizada em qualquer SDR, independente do número de

bifurcações. Um método para calcular fluxo de carga através dessa representação também é discutido. A formulação aumenta o desempenho do algoritmo no cálculo do fluxo de carga ao aplicar as leis de *Kirchoff* da tensão e da corrente. De uma maneira simples, um ciclo pode ser identificado após o fechamento de uma chave através de uma matriz incidência de ciclos C . A matriz C é obtida através das matrizes incidência de barras e incidência de caminhos para os ramos. Outra vantagem citada pelos autores diz respeito a numeração dos ramos e nós que pode ser realizada em qualquer seqüência, não influenciando no desempenho do fluxo de carga. Aplicações da metodologia foram realizadas em um SDR hipotético com 5 transformadores de 11kV, 118 ramos e 15 possibilidades de chaveamentos para redução de perdas de potência. A principal vantagem citada pelos autores está na identificação de ciclos. Assim, a escolha de qual chave deverá ser aberta é baseada na tensão em cada nó. Detalhes do computador utilizado bem como o tempo de processamento não foram apresentados.

Ainda em 2004, Parada et al. [27] propuseram um algoritmo que utiliza SA para planejamento de SDRs através de configuração e reconfiguração dos mesmos. Trata-se de um problema de otimização combinatorial com objetivo de minimizar o custo total de investimentos e o custo total de perdas de potência ao encontrar planos de expansão. Algumas restrições foram consideradas: a rede deve permanecer radial, cada consumidor deve ser atendido por uma única subestação e a capacidade dos transformadores não deve ser excedida. Resultados numéricos demonstraram que a qualidade das soluções aumenta com o número de iterações para cada análise de temperatura realizada pelo SA. Testes com a metodologia demonstraram que a média do número de iterações independe do tamanho do problema, enquanto a média do tempo de computação aumenta com o tamanho do

problema. O autor expõe que devido ao alto tempo de processamento requerido pelos métodos exatos, uma ótima alternativa para esse problema seria utilizar métodos heurísticos com a garantia de obter soluções quase ótimas.

Em 2006, Mendoza et al. [34] apresentaram aplicações de duas técnicas de otimização multiobjetivo denominadas NSGA (*nondominated sorting genetic algorithm*) e SPEA (*strength Pareto evolutionary algorithm*) em planejamento de sistemas de distribuição de energia elétrica. A redução do custo total envolvido e minimização da área sem energia são consideradas na função objetivo. As restrições consideradas são: lei da corrente de *Kirchoff* para todos os nós, carregamento nos alimentadores e subestações, bem como queda de tensão. De acordo com o autor, sistemas de distribuição geralmente operam de forma radial, porém no processo de planejamento não é necessário utilizar a radialidade como restrição. Ambos os algoritmos incorporam várias características de AEs conhecidos na literatura. O NSGA trabalha da seguinte maneira: identifica soluções não-dominadas (indivíduos) na população em cada geração (iteração) para formar fronteiras não-dominadas baseado no conceito de não-dominância de Pareto [48, 79]. Em seguida, utiliza operadores convencionais de AG como seleção, cruzamento e mutação. Dessa forma, cria-se um conjunto com todas as soluções não-dominadas por nenhuma outra. Para criar uma nova fronteira de soluções não-dominadas, omitem-se os indivíduos da primeira fronteira e aplica-se o algoritmo aos restantes. Procede-se dessa maneira até identificar várias fronteiras de soluções. O SPEA utiliza elitismo através da manutenção de um conjunto externo de melhores soluções encontradas durante cada ciclo completo de iteração. O elitismo garante que soluções com alta adequação não serão eliminadas durante a aplicação de um algoritmo de otimização. As soluções não-dominadas no conjunto externo são

utilizadas para determinar a adequação da população atual (conjunto de soluções) e também participam do processo de seleção para a reprodução. A população externa é limitada a certa quantidade de indivíduos e, um procedimento de redução da população é necessário quando esse limite é excedido. Para isso, aplica-se um algoritmo baseado em conjuntos *fuzzy* para agrupar os indivíduos que permanecerão nessa população. Esse procedimento foi utilizado pela primeira vez nesse trabalho. Um SDR com 41 barras e 2 subestações de 20MVA foi utilizado nos testes. Ambas as técnicas – NSGA e SPEA – apresentaram resultados computacionais similares tanto em relação ao tempo de processamento quanto às fronteiras de Pareto encontradas.

Em 2008, Prasad et al. [11] apresentaram uma técnica para resolver problemas de reconfiguração de sistemas de distribuição para balanceamento de cargas utilizando AG. Reconfiguração de redes foi utilizada para transferir cargas de alimentadores mais carregados para alimentadores menos carregados. Duas equações são requeridas para realizar o balanceamento de cargas. De modo geral, o objetivo é minimizar um índice que mede o balanceamento de cargas entre os alimentadores, não violando restrições de queda de tensão, carregamento nas linhas, carregamento nos transformadores e radialidade da rede. Como consequência, as perdas de potência tendem a reduzir. Um fator relevante neste artigo é a diferenciação dos tipos de carga (residencial, comercial ou industrial) e também o período de pico em dias da semana normais e em finais de semana. Juntamente com o balanceamento do índice de carregamento nos alimentadores como um todo, um balanceamento da carga nos ramos é realizado. Operadores de reprodução, cruzamento e mutação são utilizados no AG para gerar um novo indivíduo (configuração factível do sistema). Quando uma configuração infactível é

gerada, aplica-se uma rotina para alterá-la até que atenda todas as restrições. A seleção de indivíduos para reprodução é realizada através do método da roleta. O número máximo de gerações é utilizado como regra de parada. Testes foram realizados em um SDR de 69 barras.

Neste Capítulo as principais técnicas e modelos computacionais para o tratamento de problemas de reconfiguração de redes foram apresentadas, principalmente envolvendo restabelecimento de energia em SDRs e redução de perdas de potência.

O próximo Capítulo trata o problema de restabelecimento de energia. Conceitos de Grafos são introduzidos e a formulação matemática para o problema tratado é apresentada.

3 O Problema de Reconfiguração

Neste Capítulo apresenta-se o problema de reconfiguração de redes de energia e uma formulação matemática genérica que modela em conjunto o problema de restabelecimento, redução de perdas e planejamento da expansão do sistema, entre outros.

Para apresentar essa formulação, é adequado rever alguns conceitos de teoria de grafos. Muitas situações podem ser convenientemente descritas por meio de diagramas que consistem de um conjunto de pontos, juntamente com linhas que ligam alguns pares desses pontos. Por exemplo, os pontos podem representar centros de comunicações e as linhas ligações entre os centros. A abstração matemática de situações desse tipo dá lugar ao conceito de grafo.

Um grafo G é um par $(N(G), E(G))$, onde $N(G)$ é um conjunto finito de elementos denominados nós e $E(G)$ é um conjunto não ordenado de elementos denominados arestas [105]. SDRs podem ser representados por grafos. Considere o grafo abaixo.

Figura 3.1 – Exemplo de um grafo.

Se k e y são dois nós do grafo, e o par (k,y) é uma aresta, denotada por e , diz-se que e conecta k e y . O **grau** de um nó x em um grafo G é dado pelo número de arestas que incidem em x . Por exemplo, na Figura 3.1, o grau do nó y é 3. Uma sequência de arestas $(s_0, s_1), (s_1, s_2), \dots, (s_{m-2}, s_{m-1}), (s_{m-1}, s_m)$ de um Grafo G , se todas as arestas forem distintas, é chamada de **caminho**. Se os nós s_0 e s_m são iguais, esse caminho é denominado de **ciclo**. Na Figura 3.1, têm-se exemplos de:

Caminho: $(k,y), (y,w), (w,u), (u,z), (z,y)$; ou $(k,y), (y,w), (w,u), (u,v)$ e outros;

Ciclo: $(y,w), (w,u), (u,z), (z,y)$.

Um par de nós em um grafo é um **par conexo** se existe um caminho entre esses nós. Um grafo G é um **grafo conexo** se todo par de nós em G é um par conexo. Um **grafo acíclico** é um grafo sem ciclos. Uma **árvore** é um grafo acíclico conexo. Uma **floresta** é um grafo formado por um conjunto de árvores. Quando uma floresta tem apenas uma árvore, ela é uma floresta conexa, assim, uma árvore é uma floresta conexa.

Chama-se de nó raiz ou raiz, um nó tomado como referência, que pode ter grau maior ou igual a um. O significado da raiz depende do problema modelado pelo grafo, em geral, é o início da árvore, o ponto de origem de recursos, etc. Nós que possuem grau **um** são chamados de **nós terminais**, exceto se for o nó raiz.

A topologia de um sistema de distribuição de energia elétrica pode ser representada por um grafo. A Figura 3.2 mostra um SDR com 4 alimentadores. Cada barra (nó) do sistema representa um setor e as arestas interligando as barras são chaves seccionadoras. As arestas em linha cheia representam as chaves NF, enquanto as arestas em linha tracejada representam as chaves NA. As barras 1, 2, 3 e 4 encontram-se em uma subestação.

Na ocorrência de um apagão (*blackout*) devido a uma falta em alguma parte do sistema (barra, alimentador, linha ou seção de carga), o setor em falta deve ser isolado abrindo-se todas as chaves que o conectam ao restante do sistema. Em consequência, além do setor em falta, os setores a jusante do mesmo ficarão sem energia. Áreas desenergizadas devem ser reconectadas ao sistema através do fechamento de chaves que as conectem a setores supridos de eletricidade.

Figura 3.2 – Sistema de Distribuição com 4 alimentadores.

A Figura 3.3 mostra um exemplo de reconfiguração de rede para restabelecimento de energia. Considere o setor 14 em falta. Então, esse setor deve ser isolado do restante do sistema. Esse procedimento pode ser realizado abrindo-se as chaves A e B. Após a abertura dessas chaves, os setores 15, 16 e 17 estarão em uma área fora de serviço, representada pela área sombreada na Figura 3.4. O propósito da reconfiguração, nesse caso, é encontrar outro caminho que forneça energia a esses setores sem violar restrições operacionais do sistema. Nesse exemplo, existem três caminhos para restaurar energia aos setores fora de serviço:

- 1) Fecha-se a chave C, conectando a área fora de serviço ao alimentador 1;
- 2) Fecha-se a chave D, conectando a área fora de serviço ao alimentador 3;
- 3) Fecha-se a chave E, conectando a área fora de serviço ao alimentador 4.

Figura 3.3 – Sistema de Distribuição em falta.

Figura 3.4 – Setores a jusante do setor em falta desconectados do SDR.

A Figura 3.5 mostra a nova configuração optando pelo fechamento da chave E. Após a zona faltosa ter sido isolada e a área fora de serviço ter sido reconectada ao sistema, as seguintes restrições devem ser satisfeitas [12]:

1. A estrutura radial deve permanecer após o serviço de restabelecimento;
2. Áreas a jusante do setor em falta (que ficaram fora de serviço) devem ser atendidas quando possível;
3. O montante de carga de cada alimentador do sistema não deve exceder a capacidade limite da subestação;

4. A corrente elétrica em cada ramo não deve ultrapassar a capacidade das linhas e chaves;
5. A queda de tensão em qualquer barra do SDR não deve exceder o limite permissível.

Figura 3.5 – Nova configuração.

Geralmente, além da busca por configurações que minimize a área fora de serviço, outros dois objetivos são acrescentados ao problema:

- a) Minimizar o total de perdas de potência;
- b) Minimizar o número de chaveamentos.

Uma formulação geral para problemas de reconfiguração de redes de distribuição pode ser obtida considerando juntos todos os objetivos e restrições envolvidas. Um problema genérico de reconfiguração de redes de energia pode envolver os seguintes objetivos: minimização da(s) área(s) fora de serviço, número de chaveamentos e total de perdas de potência sem violar restrições de carga e tensão. Além disso, o problema deve respeitar as 5 restrições apresentadas anteriormente.

Assim, um problema geral de SDR pode ser formulado como segue [32]:

$$\begin{aligned}
 & \text{Min. } E(F) \\
 & \text{s.a. } \begin{cases} H(F) = 0 \\ I(F) \leq 0 \\ F \text{ ser uma floresta,} \end{cases} \quad (1)
 \end{aligned}$$

onde,

- F – grafo correspondente a uma configuração do sistema, representado por uma floresta de grafo, onde cada árvore dessa floresta corresponde a um alimentador conectado a uma subestação;
- $E(F)$ – função objetivo;
- $H(F)$ – restrições de igualdade representando as equações de fluxo de carga;
- $I(F)$ – restrições de desigualdade representando as restrições operacionais da rede.

A função $E(F)$ contém, em geral, um ou mais dos seguintes componentes:

- $\phi(F)$ – número de barras com cargas fora de serviço para uma topologia de rede radial (uma floresta F);
- $\varphi(F)$ – perdas de potência no sistema para F ;
- $\psi(F, F_0)$ – número de operações de chaveamentos requerido para obter uma dada configuração F a partir da configuração original F_0 .

As restrições de igualdade correspondem às equações de fluxo de carga, que podem ser representadas como um sistema linear do tipo $Ax = b$, onde:

- A – matriz incidência de F ;
- x – vetor de corrente de linha;
- b – vetor contendo as correntes de carga nas barras ($b_i \leq 0$) ou as correntes injetadas nas subestações ($b_i > 0$).

As restrições operacionais $I(F)$ em problemas de reconfiguração de sistemas de distribuição geralmente incluem:

- um limitante superior de corrente $\bar{x}_j$ para cada corrente de linha x_j . A maior taxa $x_j / \bar{x}_j$ é denominada de carregamento da rede;

- a máxima injeção de corrente $\bar{b}_i$ em cada subestação, onde i significa a subestação i . A maior taxa $b_i / \bar{b}_i$ é denominada carregamento da subestação;
- o limitante inferior para a tensão no nó $\underline{v}$. Seja v_i a tensão na barra i e, v_b a tensão base do sistema; a maior taxa v_i / v_b é denominada maior taxa de tensão.

O vetor de tensão v é dado por $Yv=b$, onde Y é a matriz de admitância nodal, que pode ser calculada por meio da expressão $Y=AY_xA^T$, onde Y_x é a matriz admitância diagonal.

Em SDRs é possível utilizar o modelo de corrente constante e as barras serem armazenadas segundo o Modelo Pai-Filho (MPF) – do inglês, *terminal-substation order* [32]. Assim, através de um fluxo de carga, pode-se calcular o fluxo de corrente partindo dos nós terminais (nós folhas) em direção à subestação (nó raiz); enquanto as tensões podem ser obtidas de forma encadeada partindo da subestação até as barras terminais. Detalhes do fluxo de carga podem ser vistos no Capítulo 5.

A função objetivo para problemas envolvendo SDRs geralmente é não linear, descontínua e com vários ótimos locais, dificultando a utilização de PM. Por outro lado, quando AEs são empregados para resolução desse tipo de problema, algumas modificações são realizadas na formulação apresentada em (1). A fim de penalizar as configurações da rede que violarem as restrições operacionais $I(F)$, são inseridos fatores de penalidades [79]. Dessa forma, o problema em SDR pode ser reformulado como segue:

$$\begin{aligned} & \text{Min.} \quad E(F) + |\Omega I(F)| \\ & \text{s.a.} \quad \begin{cases} H(F) = 0 \\ F \text{ ser uma floresta,} \end{cases} \end{aligned} \quad (2)$$

onde Ω é uma matriz diagonal com os seguintes elementos:

$$w_{11} = \begin{cases} w_x, & \text{se, pelo menos para um } j, x_j > \bar{x}_j \\ 0, & \text{caso contrário;} \end{cases}$$

$$w_{22} = \begin{cases} w_s, & \text{se, pelo menos para um } i, b_i > \bar{b}_i \\ 0, & \text{caso contrário;} \end{cases}$$

$$w_{33} = \begin{cases} w_v, & \text{se, pelo menos para um } i, v_i < \underline{v} \\ 0, & \text{caso contrário;} \end{cases}$$

os pesos w_x , w_s e w_v são valores positivos e, $|\cdot|$ é a norma usual [106], isto é, a norma L_1 de um vetor z de tamanho n é dada por $\sum_{r=1}^n |z_r|$.

A formulação do problema anterior pode ser simplificada adaptando a estrutura de dados utilizada para representar grafos. Denominada Representação Nó-profundidade (RNP), essa estrutura é descrita no Capítulo 4. Essa representação possui operadores de modificação de uma floresta para produção de novas florestas, ou seja, gera configurações de SDRs sempre radiais em que todos os consumidores são atendidos (configurações em árvores). Dessa forma, utilizando a RNP, o problema descrito em (2) pode ser reescrito como segue:

$$\begin{aligned} & \text{Min. } E(F) + |\Omega I(F)| \\ & \text{s.a. } \begin{cases} H(F) = 0 \\ F \text{ ser dado pelos operadores da RNP.} \end{cases} \end{aligned} \quad (3)$$

A RNP de um SDR possui, naturalmente, as barras de cada árvore (alimentador) ordenadas segundo o MPF. Com isso, evita-se o uso de um algoritmo de busca [86] para obter tal modelo. Assim, o fluxo de carga pelo MPF com RNP é mais eficiente que fluxos de carga convencionais para SDRs [32, 78]. Além disso, o uso do MPF garante que as restrições de igualdade ($H(F)$) em (3) sejam satisfeitas. Assim, o problema genérico de reconfiguração de SDRs pode ser reescrito utilizando-se a propriedade da RNP possuir os nós na ordem segundo o MPF, da seguinte forma:

$$\begin{aligned}
 & \text{Min. } E(F) + |\Omega I(F)| \\
 & \text{s.a. } \begin{cases} \text{Utilizar MPF com RNP.} \\ F \text{ ser dado pelos operadores da RNP} \end{cases}
 \end{aligned} \tag{4}$$

Neste Capítulo, conceitos básicos de grafos, detalhes do problema de reconfiguração de redes de energia e uma formulação matemática genérica para esse problema foram apresentados. A formulação do problema utilizando as informações da RNP com MPF e conectividade da rede para produção de configurações em árvores, reduz o tempo de processamento pela produção exclusiva de configurações apropriadas para SDRs e pela avaliação relativamente rápida de cada configuração por um fluxo de carga mais eficiente. Outra vantagem da técnica desenvolvida é que, como são produzidas somente configurações em árvores, não há a necessidade de utilizar uma rotina específica para verificar e muito menos para corrigir configurações com ciclos ou desconexas. A utilização da RNP e seus Operadores juntamente com o fluxo de carga pelo MPF com RNP tornam também a modelagem matemática do problema mais simples, compare a Equação (4) com a Equação (1). Além disso, devido ao método desenvolvido produzir somente configurações em árvores, somente as restrições de queda de tensão, carregamento na rede e carregamento nas subestações serão consideradas na formulação matemática do problema.

4 Representação Nó-Profundidade

Neste Capítulo uma estrutura de dados eficiente para armazenar e manipular grafos é apresentada – Representação Nó-profundidade (RNP). Dois Operadores que modificam dados na RNP são utilizados para obter, de forma eficiente, novas topologias de rede. As Seções 4.1, 4.2, 4.3 e 4.4 apresentam a RNP, seus operadores e características intrínsecas da RNP que capacitam a mesma a ter alta eficiência computacional.

4.1 Fundamentos da RNP

Um sistema de distribuição pode ser visto como um conjunto de alimentadores, sendo cada alimentador composto por um ou mais setores (ver Seção 5.1). A Figura 4.1 apresenta uma floresta com três árvores. Os nós 1, 2 e 3 são raízes das árvores 1, 2 e 3 respectivamente. Esse grafo pode ser visto como um sistema de distribuição com 3 alimentadores onde os nós são setores e as arestas são chaves seccionadoras. As arestas em linhas cheias representam chaves NF e as arestas em linha tracejada representam chaves NA. Os nós 1, 2 e 3 podem estar conectados ao barramento de uma mesma subestação ou ao barramento de subestações distintas. Em geral, um adicional conectaria os nós 1, 2 e 3 para representar que todos os alimentadores estão ligados a um mesmo barramento de subestação. Assim, sua interpretação usual é de 3 alimentadores ligados a 3 barramentos diferentes de subestação.

4.2 Operadores da RNP

Esta Seção apresenta 2 operadores para gerar novas florestas utilizando a RNP. Ambos Operadores geram uma floresta F' de um grafo G quando esses são aplicados a outra floresta F de G .

Os resultados produzidos pela aplicação de ambos Operadores são similares. Dada uma floresta com duas árvores ou mais, a aplicação do Operador 1 (ou 2) a essa floresta é equivalente a transferir uma subárvore de uma árvore T_{de} para uma outra árvore T_{para} da floresta. Ao aplicar o Operador 1, a raiz da subárvore cortada de T_{de} será também a raiz dessa subárvore na nova árvore T_{para} . Ao aplicar o Operador 2, a subárvore cortada terá um novo nó raiz que poderá ser qualquer nó da subárvore cortada diferente da raiz original. Essa é a diferença básica entre os Operadores 1 e 2.

Como resultados, o Operador 1 pode produzir mudanças simples e pequenas nas árvores da floresta; enquanto o Operador 2 pode gerar alterações mais complexas.

O Operador 1 requer um conjunto de dois nós determinados previamente: o nó de poda p que indica a raiz da subárvore a ser transferida e o nó adjacente a , que é um nó em uma árvore diferente da árvore cortada (T_{de}) também adjacente a p em G . Os nós adjacentes a p podem ser armazenados em uma lista de adjacência⁵ [107]. A lista de adjacência para o grafo da Figura 4.1 é mostrada na Tabela 4.1.

⁵ A lista de adjacência considera todas as conexões possíveis de cada nó, isto é, conexões referentes tanto a chaves abertas quanto fechadas, desta forma, não se altera de uma configuração para outra.

Tabela 4.1 – Lista de adjacência para os nós do grafo da Figura 4.1.

Nó	Adjacentes				
1	4				
2	9				
3	27				
4	1	5	10		
5	4	6	11		
6	5	7	12		
7	6	8	13		
8	7	9	13	14	
9	2	8	15		
10	4	11	16		
11	5	10	12	17	
12	6	11	13	18	
13	7	8	12	14	19
14	8	13	15	20	
15	9	14	21		
16	10	17	22	23	
17	11	16	18	23	
18	12	17	19	24	
19	13	18	20	25	26
20	14	19	21	26	
21	15	20	27		
22	16	23			
23	16	17	22	24	
24	18	23	25		
25	19	24	26		
26	19	20	25	27	
27	3	21	26		

Operador 2 requer um conjunto de 3 nós: o nó de poda p , o nó adjacente a e o novo nó raiz r da subárvore a ser transferida. O funcionamento dos Operadores, considerando que o conjunto de nós requerido foi determinado previamente, é apresentado a seguir. A Seção 4.4 mostra como obter eficientemente esse conjunto.

4.2.1 Operador 1

A geração de uma floresta F' de um grafo G a partir de outra floresta F de G pode ocorrer pela aplicação do Operador 1. Esse processo é equivalente a transferência de uma subárvore de uma árvore T_{de} para outra T_{para} da mesma

floresta. Para o Operador 1, a raiz da subárvore cortada também será a raiz da subárvore na nova árvore T_{para} .

Considerando que a RNP foi desenvolvida utilizando vetores e que os nós p e a foram previamente determinados, podemos assumir que os índices de p (i_p) e de a (i_a) nos vetores T_{de} e T_{para} , respectivamente, também são conhecidos.

Para ilustrar o funcionamento do Operador 1, do grafo da Figura 4.1, as árvores T_1 e T_3 foram selecionadas como as árvores T_{de} e T_{para} , respectivamente, para aplicação do Operador 1. Os nós p e a são, respectivamente, 11 e 17. O Operador 1 pode ser descrito pelos seguintes Passos (ver Figura 4.3 a Figura 4.6):

1. Determine as posições ($i_p - i_u$) dos índices em T_{de} correspondente a subárvore que contém o nó p como raiz. Como i_p é conhecido, é necessário encontrar apenas i_u , que corresponde ao índice do último nó na subárvore que tem o nó p como raiz. O conjunto ($i_p - i_u$) corresponde ao nó p em i_p e aos consecutivos nós x no vetor T_{de} de forma que $i_x > i_p$ e $p_x > p_p$ (retângulo tracejado na Figura 4.3), onde p_x é a profundidade do nó x e p_p é a profundidade do nó p ;
2. Copie os dados do conjunto ($i_p - i_u$) de T_{de} em um vetor temporário T_{tmp} (subárvore que está sendo transferida), ver Figura 4.4. A profundidade de cada nó x do conjunto $i_p - i_u$ é atualizado como segue:

$$p_x = p_x - p_p + p_a + 1;$$
3. Crie um vetor T'_{para} contendo os dados de T_{para} e T_{tmp} , inserindo T_{tmp} a partir da posição i_a , ver Figura 4.6;
4. Construa um vetor T'_{de} compreendendo os nós de T_{de} sem os nós de T_{tmp} , ver Figura 4.5;

5. Copie a estrutura de ponteiros da floresta F para F' trocando os ponteiros para os *arrays* T_{de} e T_{para} por ponteiros para os *arrays* T'_{de} e T'_{para} , respectivamente.

Figura 4.3 – T_{de} , T_{para} e suas respectivas RNPs.

Figura 4.4 – T_{tmp} e sua RNP.

Figura 4.5 – T'_{de} e sua RNP.

Figura 4.6 – T'_{para} e sua RNP.

4.2.2 Operador 2

Assim como o Operador 1, o Operador 2 também produz uma floresta F' de um grafo G quando aplicado a outra floresta F de G . No entanto, transferindo uma subárvore de uma árvore T_{de} com uma nova raiz para outra árvore T_{para} da mesma floresta. A nova raiz pode ser qualquer nó da subárvore diferente da raiz original.

A descrição do Operador 2 assume que um conjunto de nós seja previamente determinado: o nó de poda p , o novo nó raiz r e o nó adjacente a . Os nós p e r pertencem à árvore T_{de} e o nó a pertence à T_{para} . As diferenças entre o Operador 1 e o Operador 2 estão nos Passos 2 e 3 do procedimento do Operador 1 (ver a Seção 4.2.1), isto é, a formação da subárvore cortada e o armazenamento da mesma no *array* temporário T_{tmp} são diferentes.

Os Passos 2 e 3 para o Operador 2 são descritos a seguir (ver Figura 4.7 a Figura 4.10). Para o Operador 2, o procedimento de copiar a árvore cortada pode ser separado em dois Passos: o primeiro Passo é similar ao Passo 2 do Operador 1, diferindo apenas na troca de i_p por i_r . O segundo Passo considera os nós no caminho de r a p (isto é, $r_0, r_1, r_2, \dots, r_n$, onde $r_0 = r$ e $r_n = p$) como raízes das subárvores (retângulo tracejado na Figura 4.7). O algoritmo para o segundo Passo deverá copiar a subárvore com a raiz r_i ($i = 1, \dots, n$) sem a subárvore com raiz r_{i-1} (Figura 4.8) e

armazenar a subárvore resultante em um *array* temporário T_{tmp} , ver Figura 4.9. O Operador 2 utilizará os *arrays* temporários, T_{tmp} para construir T'_{para} .

A seguir apresenta-se um exemplo de aplicação do Operador 2. As árvores T_{de} e T_{para} utilizadas na aplicação do Operador 2 são as mesmas utilizadas pelo Operador 1 na Seção 4.2.1. Os nós p e r são 10 e 16 respectivamente. As profundidades na Figura 4.8 e na Figura 4.9 são relativas ao sorteio do nó 17 como adjacente.

Figura 4.7 – T_{de} e sua RNP.

Figura 4.8 – Subárvores enraizadas nos nós do caminho de r a p .

Figura 4.9 – RNP da subárvore podada.

Figura 4.10 – Árvore T'_{para} e sua RNP.

4.3 Localizando um nó na RNP de uma floresta

A posição de um nó em F pode ser eficientemente determinada usando matrizes Π_x e um array π . Cada nó x de G possui uma matriz Π_x correspondente. Para a floresta original F_0 de G , Π_x é a matriz de uma coluna:

$$\Pi_x = \begin{bmatrix} 0 \\ i_0 \\ j_0 \\ k_0 \end{bmatrix},$$

onde i_0 é o índice da árvore que contém x (T_{i_0}), j_0 é o índice correspondente a x no array T_{i_0} e k_0 é a profundidade de x na árvore.

Suponha que uma floresta F_h está sendo gerada de outra floresta F_g ($g < h$) e que x está na subárvore que será transferida para uma nova árvore gerada F_h . Dessa maneira, x terá uma nova posição em F_h diferente da posição em F_g . Deve-se inserir uma nova coluna em Π_x com os índices correspondentes a essa nova posição, resultando em:

$$\Pi_x = \begin{bmatrix} 0 & h \\ i_0 & i_h \\ j_0 & j_h \\ k_0 & k_h \end{bmatrix}.$$

A atualização da posição é executada para todos os nós da subárvore transferida (T_{tmp}), para os nós com índices maiores que i_u na árvore T_{de} e, para os nós de T_{para} a partir da posição i_a .

O array π armazena a floresta g , da qual a floresta F_h foi gerada, na posição h de π , isto é, $\pi(h)=g$. O pai de g é $\pi(g)$, o pai de $\pi(g)$ é $\pi(\pi(g))$ e assim por diante. Assim, π é uma lista encadeada com todos os predecessores de F_h . Como a última mudança de posição de x ocorreu em um dos predecessores de h , pode-se procurar pelo predecessor de h nas colunas de Π_x . A busca inicia-se com $\pi(h)$. Se $\pi(h)$ não é encontrada em Π_x procura-se por $\pi(\pi(h))$, e assim por diante. O processo de procurar por tais colunas em Π_x pode ser eficientemente implementado através de uma busca binária [108] na lista dada por $\Pi_x(0, \cdot)$ na primeira linha de Π_x .

Uma vez identificada a coluna com o predecessor de h , é necessário apenas ler os índices de x armazenados nessa coluna.

4.4 Determinação dos nós p , r e a

Os Operadores propostos requerem um conjunto especial de nós para gerar uma floresta F' de G baseada em outra floresta F de G . Para o Operador 1, esse conjunto pode ser eficientemente obtido pela seguinte estratégia:

1. Selecione aleatoriamente um índice de uma árvore em F e, para esta árvore, selecione aleatoriamente o índice de um nó que não seja raiz, chamado de nó de poda (p);
2. Selecione aleatoriamente um nó adjacente a p (utilize para isso a lista de adjacências de G), chamado de nó adjacente (a). Se $a \notin T_{de}$, determine a posição em F utilizando o array π e a matriz Π_a ; caso contrário, selecione aleatoriamente outro a ou retorne ao Passo 1.

A estratégia para a determinação de p , r e a para o Operador 2 segue aos seguintes passos:

1. Selecione aleatoriamente um índice de uma árvore em F e, para essa árvore, selecione aleatoriamente um índice de um nó que não seja raiz. Denomine-o de nó de poda (p);
2. Determine o conjunto de nós na subárvore que tenha o nó p como raiz (faixa da RNP de i_p a i_u , ver Passo 1 do Operador 1, Seção 4.2.1). Escolha aleatoriamente um índice desse conjunto para ser o novo nó raiz (r);
3. Selecione aleatoriamente um nó adjacente a r (utilize os nós da lista de adjacências de G). Chame-o de nó adjacente a . Se $a \notin T_{de}$, determine a posição em F utilizando o array π e a matriz Π_a ; caso contrário, selecione outro a ou retorne ao Passo 1.

Este Capítulo apresentou uma estrutura de dados eficiente para armazenar e manipular grafos, a RNP. Em nosso contexto, essa representação armazena a configuração dos alimentadores de um SDR. Juntamente com a RNP, descreveram-se dois Operadores utilizados para modificar de maneira eficiente a topologia da rede. Tais operadores são capazes de gerar novas configurações sempre radiais, conexas e em pequeno tempo de processamento. Foi também apresentado como determinar o conjunto de nós requeridos pelos operadores, bem como uma maneira de encontrá-los quando armazenados em computador.

O Capítulo 5 apresenta detalhes do fluxo de carga utilizado para determinar os parâmetros da rede, o qual é baseado na RNP e também como determinar, de forma eficiente, o número de manobras necessário para implantar uma configuração de um SDR.

5 Avaliação Eficiente das Soluções

Os objetivos e restrições podem ser obtidos a partir do cálculo de um fluxo de carga, com exceção do número de manobras necessário para implantar uma nova configuração. Neste Capítulo, procedimentos computacionalmente eficientes para o cálculo de fluxo de carga (Seção 5.1) e de número de manobras (Seção 5.2) são apresentados.

5.1 Fluxo de Carga

As condições operacionais de uma rede elétrica são obtidas através de um fluxo de carga que provém a tensão complexa em cada barra e os fluxos de potência ativa e reativa em todas as linhas e transformadores dessa rede. Para o cálculo do fluxo de carga, assume-se que a topologia da rede, a potência instalada e a potência nas cargas (ou correntes de carga) são conhecidas.

Métodos convencionais de fluxo de potência para sistemas de transmissão como Newton-Raphson [109], Newton-Raphson desacoplado rápido [110] e seus derivados [111-113] em geral não são adequados para resolver fluxo de carga para sistemas de distribuição [114]. Isto ocorre devido a necessidade de fatoração de matrizes exigido por esses métodos sendo que tais matrizes associadas a sistemas de distribuição são mal condicionadas e a quantidade de barras é grande, sendo necessário a utilização de matrizes com milhares de linhas e colunas. Algumas

características particulares dos sistemas de distribuição são responsáveis pelo mau condicionamento das matrizes: alta relação R/X (resistência/reatância), alto número de cargas distribuídas e partes da rede com alta impedância associadas a trechos de baixa impedância⁶.

Nas últimas décadas, com a automatização dos sistemas de distribuição, muitas aplicações têm requerido métodos de fluxo de potência eficientes e robustos. Surgiram então, métodos específicos para distribuição que exploram sua natureza radial. Esses métodos têm apresentado desempenho melhor que os métodos convencionais (em termos de tempo de processamento, esforço computacional, convergência, etc.). Devido a robustez e simplicidade na implementação, métodos de varredura direta/inversa [77, 78, 88, 114] têm obtido a preferência de muitos pesquisadores. Esses métodos exploram uma característica especial dos sistemas de distribuição radiais, a existência de um único caminho entre qualquer barra da rede e a barra raiz, na subestação [114]. Assim, a solução de um fluxo de potência para sistemas radiais pode ser obtida sem a necessidade de resolução de um sistema de equações $Ax = b$ (ver Capítulo 3) como nos métodos tradicionais. Existem muitas variações dos métodos de varredura direta/inversa: método da soma de potência [115], método da soma de corrente [88], método da soma de impedância e outros.

Na Seção 5.1.1, o método da soma de potência é apresentado. O método proposto é apresentado nas Seções 5.1.2, 5.1.3 e 5.1.4. O mesmo é baseado no modelo de corrente constante e utilizado para soma de correntes com a RNP. Na Seção 7.8 resultados encontrados por esses dois métodos de fluxo de carga serão comparados.

⁶ Chaves seccionadoras, reguladores de tensão e, pequenos trechos de rede.

5.1.1 Métodos de soma de potência

Nos métodos de varredura direta/inversa existem dois estágios para obter as grandezas do sistema:

Estágio 1: Varredura Inversa

Este estágio inicia pela última barra da árvore e termina na barra raiz. No método de Soma de Potência, a potência ativa na iteração k , $P^{(k)}$ e a potência reativa na iteração k , $Q^{(k)}$ são calculadas para todas as barras da rede.

Estágio 2: Varredura Direta

Este estágio começa pela barra raiz e termina na barra terminal. Nesse passo, a tensão $V^{(k)}$ deve ser atualizada para todas as barras da rede. Assim, o problema de fluxo de carga de uma rede radial pode ser resolvido iterativamente a partir do conjunto de duas equações recursivas. Inicialmente, a amplitude e ângulo da tensão da barra raiz são atribuídos a todas as barras da rede. Em seguida, na varredura inversa, as seguintes equações são utilizadas:

$$P_{i-1} = P_i + r_i \frac{P_i'^2 + Q_i'^2}{V_i^2} + P_{Li}, \quad (5)$$

$$Q_{i-1} = Q_i + x_i \frac{P_i'^2 + Q_i'^2}{V_i^2} + Q_{Li}, \quad (6)$$

onde

P_i é o fluxo de potência ativa no ramo i ;

Q_i é o fluxo de potência reativa no ramo i ;

P_{Li} é a potência ativa de carga na barra i ;

Q_{Li} é a potência reativa de carga na barra i ;

$$P_i' = P_i + P_{Li};$$

$$Q_i' = Q_i + Q_{Li}.$$

Na varredura direta, as seguintes equações são utilizadas:

$$V_{i+1}^2 = V_i^2 - 2(r_i P_i + x_i Q_i) + (r_i^2 + x_i^2) \frac{P_i^2 + Q_i^2}{V_i^2}, \quad (7)$$

$$\delta_{i+1} = -\tan^{-1}(k_1 / k_2), \quad (8)$$

$$k_1 = \frac{(P_i x_i - Q_i r_i)}{V_i},$$

$$k_2 = V_i - \frac{(P_i r_i - Q_i x_i)}{V_i}.$$

Se o ângulo da tensão na barra i é δ_i (ao invés de zero), δ_i torna-se

$$\delta_{i+1} = \delta_i - \tan^{-1}(k_1 / k_2), \quad (9)$$

onde

V_i é a amplitude da tensão na barra i ;

δ_i é o ângulo da tensão na barra i ;

r_i é a resistência série do ramo i ;

x_i é a reatância série do ramo i .

Computacionalmente, o problema de fluxo de carga pelo método da soma de potência pode ser dividido em 4 passos:

1. Leia os dados do sistema. Construa as árvores armazenando as barras da rede na estrutura de dados. Assuma a tensão inicial para todas as barras sendo a mesma da barra raiz;
2. Pelo estágio inverso, calcule o fluxo de potência ativo e reativo em cada ramo da árvore pelas equações (5) e (6), respectivamente;
3. Pelo estágio direto, calcule a tensão complexa na barra final de cada ramo usando as equações (7) e (8) ou (9);
4. Encontre as diferenças de tensão entre a iteração atual e a última iteração para todas as barras (amplitude e ângulo separadamente).

Se todas as diferenças são menores que uma tolerância especificada, pare o processo; caso contrário, vá para o passo 2.

5.1.2 Extensão da RNP para fluxo de carga

Em sistemas reais de distribuição de energia elétrica, nem todos os trechos entre barras são separados por chaves. Conjuntos de barras e linhas não separadas por chaves são denominados setores. Em muitos trabalhos, as barras de cargas de um setor são modeladas como se estivessem concentradas em um único ponto. Tal procedimento reduz o grau de confiabilidade do sistema.

Neste trabalho, procura-se reproduzir um sistema real com a maior fidelidade possível. Utiliza-se para isso a RNP em dois níveis diferentes: a RNP do alimentador e a RNP do setor. Considere os 2 alimentadores mostrados na Figura 5.1. As barras em vermelho pertencem ao alimentador 1, enquanto as barras em preto são do alimentador 2. Na Figura 5.1 e na Figura 5.2, retângulos representam barras em uma subestação, círculos são barras do SDR (barras de carga, extremidades de chaves, ponto de conexão de duas ou mais linhas), linhas cheias representam as linhas do SDR, linhas tracejadas são chaves seccionadoras NF e a linha interrompida representa uma chave seccionadora NA.

Figura 5.1 – SDR com dois alimentadores.

Ao agrupar as barras e linhas não separadas por chaves da Figura 5.1 (ver Figura 5.2), tem-se um grafo em que todas as arestas são chaves seccionadoras. Veja a Figura 5.3.

Figura 5.2 – Agrupamento das linhas e barras em setores.

Figura 5.3 – Grafo representando setores do SDR da Figura 5.2.

Para a Figura 5.3 há 2 RNPs, uma para o alimentador 1 (em vermelho) e outra para o alimentador 2 (em preto), denominadas RNP do alimentador. As arestas em vermelho e preto são chaves seccionadoras NF e a aresta em verde é uma chave aberta. Assim, temos uma estrutura T_1 que armazena o endereço de memória da RNP do alimentador 1 e a estrutura T_2 armazena o endereço de memória da RNP do alimentador 2:

$$T_1 = \begin{bmatrix} \text{Profundidade} \\ \text{Nó} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 & 2 \\ A & B & C & D \end{bmatrix},$$

$$T_2 = \begin{bmatrix} \text{Profundidade} \\ \text{Nó} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 & 2 \\ E & F & G & H \end{bmatrix}.$$

A partir da Figura 5.2, é possível também associar RNPs aos setores. Em outras palavras, cada trecho de linhas e barras não separadas por chaves podem ser analisados como uma árvore de grafo. Cada setor pode ter mais de um nó raiz, dependendo do sentido em que está sendo alimentado. Assim, pode haver mais de uma árvore representando cada setor. Para a finalidade de fluxo de carga, acrescenta-se a cada uma dessas árvores, o nó adjacente ao seu nó raiz. Para o acréscimo do nó adequado, pode-se utilizar a seguinte regra: todo nó i adjacente a um nó x , tal que i pertença a um setor diferente do setor de x , sendo x raiz, é acrescentado à árvore correspondente ao setor de x . Exceções ocorrem para os nós em uma subestação, que possuem somente um setor vizinho. A Tabela 5.1 mostra as árvores geradas a partir da Figura 5.2, acrescentando os nós adjacentes.

A RNP do setor pode ser representada computacionalmente de forma semelhante a RNP do alimentador, onde as árvores foram armazenadas em estruturas denotadas por T_i . Para RNP do setor, denotar-se-á por B_{sr} , onde s representa o setor em análise e r refere-se ao setor pelo qual a energia chega ao setor s , setor pai. Para um mesmo setor s podem existir mais de uma RNP conforme mostra a Tabela 5.2.

Como exemplo, considere o setor D da Figura 5.3. Fluxo de corrente pode chegar ao setor D por dois caminhos diferentes, através do setor B ou do setor H. Logo, pelo setor B, tem-se $B_{sr} = B_{DB}$ e para o setor H, tem-se $B_{sr} = B_{DH}$. As estruturas que armazenam as RNPs possíveis para o setor D são mostradas a seguir:

$$B_{DB} = \begin{bmatrix} 0 & 1 & 2 & 2 & 3 \\ 6 & 7 & 8 & 9 & 10 \end{bmatrix},$$

$$B_{DH} = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 \\ 15 & 10 & 9 & 7 & 8 \end{bmatrix}.$$

A análise da RNP do alimentador nos informa qual das configurações acima utilizar. Através da RNP do alimentador 1, notamos que o setor D está conectado ao setor B. Portanto, no exemplo acima o *r* correto é o setor **B**, conforme RNP abaixo.

$$T_1 = \begin{bmatrix} 0 & \mathbf{1} & 2 & 2 \\ A & \mathbf{B} & C & D \end{bmatrix}.$$

Tabela 5.1 – Árvores dos setores com o nó adicional.

Setor	Árvores	
A
B
C
D
E
F
G
H

Tabela 5.2 – RNPs dos setores mostrados na Tabela 5.1.

Setor	RNPs do Setor	
A	0	
	1	
B	0 1 2	0 1 2
	1 3 6	7 6 3
C	0 1 2	
	3 4 5	
D	0 1 2 2 3	0 1 2 3 4
	6 7 8 9 10	15 10 9 7 8
E	0	
	2	
F	0 1	0 1
	14 11	2 11
G	0 1 2	
	11 12 13	
H	0 1 2	0 1 2
	10 15 14	11 14 15

Deve-se observar que a determinação de todas as RNPs de cada setor pode ser executada por um procedimento *off-line*, deixando todos os cálculos prontos para serem utilizados pelo fluxo de carga *on-line*.

5.1.3 Varredura direta/inversa com RNP

Devido características específicas de SDRs e a necessidade de encontrar configurações em um curto tempo de processamento, optou-ser por utilizar o modelo de corrente constante no fluxo de carga do método proposto neste projeto. Dentre as características dessas redes, destacam-se: a natureza da rede ser puramente radial e as barras estarem naturalmente armazenadas segundo o MPF na RNP. O modelo

da linha da rede correspondente a uma impedância constante Z_{mn} para cada linha entre as barras m e n .

O algoritmo de fluxo de carga finaliza seu ciclo após a execução de 2 sub-rotinas: uma obtém as correntes a jusante por meio da varredura inversa (**CORRENTES**, ver Algoritmo 1) para todas as barras de um alimentador e a outra utiliza as correntes a jusante para obter as tensões nas barras do mesmo alimentador utilizando a varredura direta (**TENSÕES**, ver Algoritmo 2). Com o modelo de corrente constante, a convergência é atingida em um único ciclo, diferindo de vários modelos usuais de varredura direta/inversa [77, 88, 114].

A sub-rotina **CORRENTES** percorre os nós (barras) do sistema partindo dos nós terminais até o nó raiz, ver Algoritmo 1. É importante observar que a ordem dos nós visitados é no sentido dos nós terminais para a raiz, ordem que está pré-determinada nas RNPs do alimentador e do setor. A carga de uma barra m é denominada $I(m)$ e a corrente a jusante dessa barra m é denominada $J(m)$.

Algoritmo 1. Pseudocódigo para calcular a corrente a jusante de uma barra.

CORRENTES

PARA k :=última posição em T , **ATÉ** K :=1, **FAÇA**:

u := próxima posição (menor que k) em que a profundidade do nó seja $T.prof(k) - 1$;

s := $T.no(k)$;

r := $T.no(u)$;

$J(T)$:= 0 //A corrente a jusante em todas as barras em T recebem zero

PARA q :=última posição em B_{sr} , **ATÉ** q :=1, **FAÇA**:

p := próxima posição (menor que q) em que a profundidade do nó seja $B_{sr}.prof(q) - 1$;

m := $B_{sr}.no(p)$;

n := $B_{sr}.no(q)$;

corrente_acumulada:= $J(n) + I(n)$;

$J(m)$:= $J(m) +$ corrente_acumulada;

FIM

FIM

A sub-rotina **TENSOES** é mostrada no Algoritmo 2. Novamente, a ordem em que os nós serão visitados está determinada nas RNPs do alimentador e do setor.

As tensões são obtidas partindo da barra raiz em direção às barras terminais. O algoritmo **TENSOES** necessita que as correntes a jusante de cada barra tenham sido obtidas previamente pela rotina **CORRENTES**. Assim, para se obter as tensões em cada barra é necessária a corrente na barra $I(n)$, a tensão a montante da barra n $V(m)$, e a impedância Z_{mn} entre as barras m e n .

Algoritmo 2. Pseudocódigo para calcular as tensões nas barras.

TENSOES

PARA $k:=1$, **ATÉ** $K:=\text{última posição em } T$, **FAÇA**:

$u :=$ próxima posição (*menor que* k) em que a profundidade do nó seja $T.\text{prof}(k) - 1$;

$s:=T.\text{no}(k)$;

$r:=T.\text{no}(u)$;

$B_{sr}.\text{no}(0) := V_{\text{sub}}$; // A tensão nas barras da subestação geralmente é 13.800V

PARA $q:=1$, **ATÉ** $q:=\text{última posição em } B_{sr}$, **FAÇA**:

$p :=$ próxima posição (*menor que* q) em que a profundidade do nó seja $B_{sr}.\text{prof}(q) - 1$;

$m:=B_{sr}.\text{no}(p)$;

$n:=B_{sr}.\text{no}(q)$;

$\Delta V := Z_{mn} \cdot (J(n) + I(n))$; // Queda de tensão na rede entre as barras m e n .

$V(n) := V(m) - \Delta V$; // tensão na barra n

FIM

FIM

Por fim, o Algoritmo 3 apresenta a rotina principal do fluxo de carga utilizando a RNP. Como os Operadores 1 e 2 modificam pares de árvores de uma floresta, para cada nova floresta, apenas 2 alimentadores são alterados. Assim, as sub-rotinas **CORRENTES** e **TENSOES** são aplicadas somente a esses alimentadores, pois o fluxo de carga para os outros alimentadores não se alteram.

Algoritmo 3. Pseudocódigo para determinar o fluxo de carga em SDRs.

ALGORITMO PRINCIPAL

// para cada estrutura T_i (indicada por um ponteiro) de F

// chama-se as sub-rotinas **CORRENTES** e **TENSOES**

PARA $i := 0$ **ATÉ** $i :=$ número de ponteiros em F **FAÇA**

CORRENTES ($F \rightarrow T_i$);

TENSOES ($F \rightarrow T_i$);

FIM

5.1.4 Exemplo de aplicação do fluxo de carga proposto

O funcionamento dos Algoritmos 1, 2 e 3 (ver Seção 5.1.3) são ilustrados a seguir para calcular o fluxo de carga do alimentador 2 da Figura 5.1. A RNP do alimentador 2 é a seguinte:

$$T_2 = \begin{bmatrix} \text{Profundidade} \\ \text{Nó} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 & 2 \\ E & F & G & H \end{bmatrix}.$$

Algoritmo 1 – Cálculo das correntes a jusante

O primeiro passo é zerar as correntes a jusante de todas as barras do alimentador 2, ou seja,

$$J[i] = 0, \text{ para } i = \text{todas as barras de } T_2.$$

Ao aplicar a rotina para calcular as correntes a jusante, tem-se: $s = H$, $r = F$.

Portanto, utiliza-se a árvore B_{HF} da Tabela 5.2, isto é,

$$B_{HF} = \begin{bmatrix} 0 & 1 & 2 \\ 11 & 14 & 15 \end{bmatrix}.$$

A ordem dos nós na B_{HF} resulta na seguinte seqüência de cálculos de correntes a jusante:

$$J[14] = J[14] + I[15] + J[15] = I[15],$$

$$J[11] = J[11] + I[14] + J[14] = I[14] + I[15].$$

Seguindo a ordem da RNP do alimentador 2, determinam-se a representação adequada do setor G: $s = G$, $r = F$, ou seja, B_{GF} ,

$$B_{GF} = \begin{bmatrix} 0 & 1 & 2 \\ 11 & 12 & 13 \end{bmatrix}.$$

Dessa RNP, obtém-se a seguinte seqüência de cálculo de corrente a jusante:

$$J[12] = J[12] + I[13] + J[13] = I[13],$$

$$J[11] = J[11] + I[12] + J[12] = I[14] + I[15] + I[12] + I[13].$$

Note que as cargas dos nós 14 e 15 já estavam armazenadas em $J[11]$ quando utilizou B_{HF} . Agora, as cargas dos nós 12 e 13 também são armazenadas.

O setor F é o próximo da seqüência, assim $s = F$, $r = E$, corresponde a B_{EF} :

$$B_{FE} = \begin{bmatrix} 0 & 1 \\ 2 & 11 \end{bmatrix},$$

a qual implica na seguinte seqüência de cálculo:

$$J[2] = J[2] + I[11] + J[11] = I[11] + I[14] + I[15] + I[12] + I[13].$$

Com isso tem-se as correntes a jusante de todas as barras do alimentador 2 calculadas. Em seguida, aplica-se o procedimento para o cálculo das tensões (Algoritmo 2).

Algoritmo 2 – Cálculo das tensões

Considere V_{sub} sendo a tensão complexa nas barras em uma subestação.

Seguindo o Algoritmo 2 e a RNP do setor 2, tem-se $s = F$, $r = E$. Assim, utilizando a estrutura B_{FE} , procede-se como segue:

$$V[2] = V_{sub},$$

$$V[11] = V[2] - Z_{2-11}^*(J[11] + I[11]).$$

Em seguida, $s = G$, $r = F$, utilizando a estrutura B_{GF} :

$$V[12] = V[11] - Z_{11-12}^*(J[12] + I[12]),$$

$$V[13] = V[12] - Z_{12-13}^*(J[13] + I[13]).$$

Por fim, $s = H$, $r = F$, utilizando a estrutura B_{HF} :

$$V[14] = V[11] - Z_{11-14}^*(J[14] + I[14]),$$

$$V[15] = V[14] - Z_{14-15}^*(J[15] + I[15]).$$

Desta forma, tem-se todas as tensões e correntes a jusante das barras do alimentador 2 calculadas.

5.2 Cálculo do número de manobras

A adequação de cada configuração gerada pela aplicação dos Operadores 1 ou 2 é avaliada através de parâmetros determinados por um fluxo de carga (ver Seção 5.1) e pelo número de manobras necessárias para implementar essa nova configuração. Em geral, o número de manobras é determinado a partir da comparação entre vetores que guardam o estado das chaves (de forma binária) de cada configuração. Dessa forma, para obter o número de manobras necessárias para implantar uma dada configuração, comparam-se o vetor com o estado de todas as chaves de uma dada configuração com o vetor que armazena o estado das chaves da configuração inicial. Porém, o processo envolve custo computacional relativamente alto, pois para cada nova configuração gerada, é necessário criar um vetor com o estado atual das chaves e realizar a comparação com o vetor da configuração inicial. Como esses vetores têm o comprimento igual ao número de chaves (m) no sistema, o tempo para percorrê-lo uma vez é bem maior que o tempo de realizar uma modificação no sistema pelos operadores 1 ou 2, que requerem tempo de computação da ordem do tamanho dos alimentadores envolvidos, em geral, bem menor que m .

Para melhorar o desempenho computacional do método, um algoritmo que determina o número de manobras de forma eficiente é proposto. Para esta formulação é necessário apenas um vetor com o estado das chaves na configuração inicial e outro vetor de tamanho g_{max} que guarda a quantidade de chaves alteradas em relação à configuração inicial, onde g_{max} é o número máximo de gerações.

O procedimento de isolar o setor em falta e conectar os setores a jusante do mesmo ao SDR exige manobras de chaves que nem sempre ocorrem aos pares. Para ilustrar esse procedimento, a Figura 3.3 foi reproduzida na Figura 5.4, onde o

setor 14 sofreu uma falta. O primeiro procedimento é isolar o setor em falta, que pode ser realizado abrindo-se as chaves A e B (2 manobras de chaves). Porém, os setores 15, 16 e 17 ficaram desconectados do sistema. Para conectá-los ao SDR é necessário fechar uma chave que os conectam a algum alimentador, que pode ser realizado fechando a chave C, D ou E (1 manobra de chave). Portanto, na primeira alteração de topologia, 3 manobras de chaves foram necessárias.

Figura 5.4 – Operações necessárias para isolar o setor em falta.

Após o procedimento descrito anteriormente, as chaves alteradas sempre ocorrerão aos pares, ou seja, quando uma chave é aberta, outra deve ser fechada. Para determinar o número de manobras para originar uma dada configuração, considere o estado das chaves em 3 configurações específicas: configuração inicial (o), configuração alterada (x) e configuração final (y). Dessa forma, considerando que uma configuração y originou-se de alterações em uma configuração x , temos 3 possibilidades para computar o número de chaves alteradas da configuração y :

1) Os estados das duas chaves alteradas em y , em relação a x , são diferentes dos estados dessas chaves em o . Portanto, o número de chaves alteradas de y será o número de chaves alteradas de x **mais** 2. Considere que as chaves 1 e 4 da Tabela 5.3 foram alteradas em x para originar y . Como $y_1 \neq o_1$ e $y_4 \neq o_4$, duas

manobras de chaves a mais que as manobras para implantar x são necessárias. Como de o para x , 2 alterações foram realizadas, de o para y são necessárias 4;

Tabela 5.3 – Manobras de chaves: Caso 1.

		Configurações				
		<i>o</i>	<i>...</i>	<i>x</i>	<i>...</i>	<i>y</i>
<i>Chaves / estados</i>	1	1		1		0
	2	1		0		0
	3	0		1		1
	4	0		0		1
	5	1		1		1

2) Os estados das duas chaves alteradas em y , em relação a x , são iguais aos estados dessas chaves em o . Portanto, o número de chaves alteradas de y será o número de chaves alteradas de x **menos** 2. Considere alterações realizadas nas chaves 2 e 3 da Tabela 5.4. Como $y_2 = o_2$ e $y_3 = o_3$, os estados dessas duas chaves em y retornaram aos seus estados em o . Portanto, para implantar y serão necessárias 2 manobras a menos que o número de manobras para implantar x ;

Tabela 5.4 – Manobras de chaves: Caso 2.

		Configurações				
		<i>o</i>	<i>...</i>	<i>x</i>	<i>...</i>	<i>y</i>
<i>Chaves / estados</i>	1	1		1		1
	2	1		0		1
	3	0		1		0
	4	0		0		0
	5	1		1		1

3) O estado de uma das chaves alteradas em y , em relação a x , é igual ao estado dessa chave em o e, o estado da outra chave alterada é diferente. Portanto, o número de chaves alteradas de y será igual ao número de chaves alteradas de x .

Na Tabela 5.5, nota-se que $y_2 = o_2$ e $y_5 \neq o_5$. Portanto, para originar a configuração y assim como para originar x , são necessárias 2 manobras de chaves.

Tabela 5.5 – Manobras de chaves: Caso 3.

		Configurações				
		o	...	x	...	y
Chaves / estados	1	1		1		1
	2	1		0		1
	3	0		1		1
	4	0		0		0
	5	1		1		0

Em todos os casos acima, 2 mudanças de chaves em x para originar y são necessárias, porém não garante que essas duas mudanças sejam efetivas, ou seja, em relação à configuração inicial o . Por esse motivo, não se pode simplesmente dizer que o número de manobras necessárias para implantar y é o número de manobras para implantar x mais 2.

Exemplo ilustrativo de cálculo do número de manobras

O Algoritmo 4 mostra os passos necessários para determinar o número de manobras realizadas para obter uma configuração y a partir de uma configuração x .

Algoritmo 4. Pseudocódigo para determinar o número de manobras.

MANOBRAS

// para cada configuração y gerada a partir de uma configuração x , comparam-se os estados // das 2 chaves alteradas com os estados delas na configuração inicial o

PARA $y := 1$ **ATE** $y := g_{max}$, **FACA**

 COMPARAM-SE os estados das chaves i e j alteradas em y em relação a o ;

SE $y_i \neq o_i$ e $y_j \neq o_j$ //Caso 1

 //atribui à configuração y o número de manobras da configuração x mais 2
 manobras $_y$ = manobras $_x$ + 2;

SE $y_i = o_i$ e $y_j = o_j$ //Caso 2

 //atribui à configuração y o número de manobras da configuração x menos 2
 manobras $_y$ = manobras $_x$ - 2;

SE $y_i = o_i$ ($y_i \neq o_i$) e $y_j \neq o_j$ ($y_j = o_j$) //Caso 3

 //atribui à configuração y o número de manobras da configuração x
 manobras $_y$ = manobras $_x$;

FIM

Para ilustrar o Algoritmo 4, considere o SDR representado na Figura 5.3 com uma chave NA (linha verde tracejada) acrescentada entre as barras C e G (ver Figura 5.5) como sendo a configuração inicial (o). O vetor com o estado das chaves para a configuração o é representado a seguir:

Chaves	1	2	3	4	5	6	7	8
Estado Inicial	1	1	1	0	0	1	1	1

Figura 5.5 – Cálculo das manobras: configuração inicial.

O algoritmo de reconfiguração inicialmente criou um vetor para armazenar as manobras necessárias para gerar cada configuração:

configuração	0	1	2	3	4	...	g_{max}
manobras	0					...	

Para gerar a configuração 1 a partir da configuração o , considere que a chave 3 foi aberta e a chave 5 fechada. Comparam-se, então, os estados dessas duas chaves com os estados das mesmas na configuração o . Portanto, o estado da chave 3 é (0) (foi aberta) e o estado da chave 5 é (1) (foi fechada). Como ambos são diferentes de seus estados na configuração inicial, são necessárias duas manobras para implantar a configuração 1 (y). A Figura 5.6 mostra a topologia do SDR da configuração 1.

Figura 5.6 – Cálculo das manobras: configuração 1.

Aplicando o Algoritmo 4 e verificado que $y_i \neq o_i$ e $y_j \neq o_j$ (ver Figura 5.6), caracterizou-se o caso 1. O vetor que armazena as manobras necessárias para implantar uma nova configuração é então atualizado para:

configuração	0	1	2	3	4	...	g_{max}
manobras	0	2				...	

Para originar a configuração 2 (y) a partir da configuração 1 (x), suponha que a chave 8 é aberta e a chave 3 fechada. Ao comparar o estado da chave 3 (agora fechada) com seu estado na configuração o , verifica-se os estados iguais. Porém, o estado da chave 8 (aberta) é diferente de seu estado na configuração o . Portanto, o número de manobras realizadas para obter a configuração 2 (y) em relação a configuração inicial (o) continua sendo 2, ou seja, $y_i = o_i$ e $y_j \neq o_j$ (Caso 3). A Figura 5.7 mostra a nova topologia do SDR.

Figura 5.7 – Cálculo das manobras: configuração 2.

configuração	0	1	2	3	4	...	g_{max}
manobras	0	2	2			...	

Suponha agora, que a configuração 1 foi escolhida para ser alterada (x) e originar a configuração 3 (y). A chave 7 é aberta e a chave 4 é fechada, alterando

seus estados para 0 e 1 respectivamente. Como ambos estados são diferentes de seus estados na configuração inicial, somam-se 2 manobras ao número de manobras da configuração 1 (x). Portanto, são necessárias 4 manobras para obter a configuração 3 (y) (ver Figura 5.8).

Figura 5.8 – Cálculo das manobras: configuração 3.

configuração	0	1	2	3	4	...	g_{max}
manobras	0	2	2	4	...	...	...

Uma terceira possibilidade ocorre quando os estados das duas chaves alteradas se tornam iguais seus estados na configuração inicial. Considere as seguintes alterações realizadas na configuração 2 (x) para gerar a configuração 4 (y): A chave 5 é aberta e a chave 8 é fechada. Nesse caso subtraem 2 manobras do vetor de manobras da configuração 2 (x), obtendo 0 manobras, ou seja, a configuração y se tornou igual a configuração 0 mostrada na Figura 5.5. O vetor atualizado com o número de manobras é:

configuração	0	1	2	3	4	...	g_{max}
manobras	0	2	2	4	0	...	...

Detalhes de algoritmos específicos para Sistemas de Distribuição, bem como o Algoritmo para o cálculo do fluxo de carga em SDRs utilizando o modelo de corrente constante foram apresentados na Seção 5.1, o qual explora principalmente a natureza radial das redes de distribuição e a forma como os nós são armazenados pela RNP para se obter um algoritmo de fluxo de carga mais eficiente. Uma forma

eficiente de obter o número de manobras é descrito na Seção 5.2. Ao programar esses algoritmos, houve uma redução considerável no tempo de processamento conforme é demonstrado no Capítulo 7.

6 Algoritmos Evolutivos

Métodos de otimização e busca estocástica têm recebido crescente interesse nas últimas décadas, principalmente os baseados em princípios da teoria da evolução. O desenvolvimento de modelos computacionais inspirados nos mecanismos evolutivos caracteriza-se pela configuração de algoritmos de otimização robustos e sistemas adaptativos [116]. AEs são algoritmos computacionais pertencentes a uma classe de métodos regidos por princípios evolutivos. Tais princípios são oriundos do mundo biológico e baseados na teoria da evolução Darwiniana [117, 118]. Os AEs tentam abstrair e imitar mecanismos evolutivos para resolução de problemas que requerem adaptação, busca e otimização.

Em um AE, um conjunto de soluções (população) modelado computacionalmente é manipulado em cada iteração de modo a melhorar a adequação⁷ média dos indivíduos que formam essa população em relação ao ambiente em que está. Os indivíduos da população são avaliados e as soluções com maior aptidão são selecionadas para a reprodução formando uma nova geração. Com o decorrer das gerações, as soluções mais adequadas acabam dominando a população, resultando em um aumento na qualidade das soluções.

Os AEs apresentam algumas vantagens em relação a outras técnicas de busca e otimização: (i) trabalham em paralelo com um conjunto de soluções (população de indivíduos em potencial), em contraste com vários outros métodos de busca e otimização que analisam apenas uma solução para o problema a cada

⁷ *Fitness*, aptidão, adequabilidade são sinônimos comuns para adequação.

iteração; (ii) não requerem o conhecimento prévio das características do problema e não dependem de certas propriedades da função objetivo tais como convexidade ou diferenciabilidade; (iii) são guiados pela aptidão dos indivíduos, avaliados por uma função adequação; (iv) a estrutura do AE possui certa independência do tipo de problema que está sendo resolvido; (v) possuem maior capacidade de escapar de ótimos locais (por isso são muito utilizados em problemas que envolvem otimização global); (vi) resolvem problemas com modelos matemáticos complexos de modo simples; (vii) apresentam fácil acoplamento com outras técnicas (hibridização) ou aplicações. Por esses motivos, os AEs são aptos a resolver dentre outros, um amplo campo de problemas não lineares, descontínuos, discretos, multivariáveis e de otimização combinatorial de grande escala.

Devido às restrições das técnicas de PM, os AEs têm tido destaque especial em problemas de otimização com múltiplos objetivos, principalmente os que envolvem objetivos conflitantes. Existem na literatura, alguns trabalhos de revisão de AEs multi-objetivos, dentre os quais se destacam: Deb [48], Coello et al. [119], Fonseca e Fleming [120]; Tamaki et al. [121]; Veldhuizen e Lamont [122].

Existem várias subáreas na Computação Evolutiva. Dentre elas, destacam-se: Programação Evolucionária [123], Estratégias Evolucionárias [123, 124] e Algoritmos Genéticos [79, 92, 123, 125, 126]. Apesar dos AEs terem sido desenvolvidos muito tempo atrás, foram popularizados a partir dos trabalhos de Holland em 1975 [127] e Goldberg em 1989 [79]. Esses algoritmos tornaram-se tão difundidos e aplicados em engenharia que mais de 95% dos artigos envolvendo métodos heurísticos publicados em sistemas de potência são baseados em AGs [128]. O Algoritmo 5 apresenta os principais passos de um AE genérico.

Algoritmo 5. Pseudocódigo de um AE genérico.

ALGORITMO AE

// inicializa o contador de gerações

$g := 0;$

// gera aleatoriamente uma população inicial $P_i(g_0)$

$P(g) := \text{POP_INIC}(P_i(g_0));$

// avalia os indivíduos da população inicial segundo uma função de adequação

AVALIE ($P(g)$);

// teste do critério de parada (g_{max} , por exemplo)

ENQUANTO critério de parada não atingido **FAÇA**

 // aleatoriamente selecione uma sub-população de (P_i) para gerar descendentes.

$P_i := \text{ALEATORIAMENTE-SELECIONE}(\text{sub-pop.});$

 // alteração desses indivíduos através dos Operadores para gerar nova população.

$P' := \text{APLICA_OPERADORES}(P_i(g));$

 // avaliação dos novos indivíduos de P' segundo a função adequação.

AVALIE(P');

 // selecione sobreviventes entre $P(g)$ e P'

$P(g + 1) := \text{SOBREVIVENTES}(P(g), P');$

 //incrementa o contador de gerações

$g := g + 1;$

FIM

Os AGs são modelos computacionais inspirados não somente na teoria evolutiva das espécies, mas também nos processos de reprodução celular. Os AGs baseiam-se nos mecanismos de seleção natural e da genética. Essas técnicas têm sido utilizadas com resultados relevantes para a solução de uma grande variedade de problemas de otimização. Esses algoritmos realizam uma estratégia de busca paralela e estruturada, capaz de reforçar a busca de pontos de “alta aptidão”. Apesar de não garantirem a convergência para a solução ótima, é possível em muitos casos, atingir-se soluções quase ótimas para problemas que requerem uma resposta rápida ou em que, por exemplo, o tamanho do problema inviabiliza a convergência de métodos exatos para a solução ótima. Os cromossomos são, geralmente, compostos por cadeias de *bits*, representando os indivíduos. Os operadores mais comuns dos AGs são a recombinação (ou *crossover*) e a mutação. Para ilustrar o processo de recombinação, considere o par de indivíduos abaixo. Cada cromossomo

possui k posições (sendo k um número inteiro) e será selecionado um local l (lócus) do cromossomo para ser o ponto de *crossover* dos cromossomos. Sejam os cromossomos:

$$A_1 = 011/01 \quad \text{e} \quad A_2 = 110/10,$$

supondo que $l = 3$, como indicado pelo símbolo “/”, o operador de *crossover* produz 2 novos cromossomos:

$$A_1' = 01110 \quad \text{e} \quad A_2' = 11001.$$

Mutação é a alteração aleatória do valor de uma posição na *string*. Na codificação binária do exemplo em questão, a mutação implica em simplesmente mudar um *bit* 1 (0) para 0 (1), conforme alterações realizadas no 3º bit de A_1' e A_2' mostradas em A_1'' e A_2'' :

$$A_1'' = 01010 \quad \text{e} \quad A_2'' = 11101.$$

Dentre os inúmeros AEs conhecidos até o momento, dois tem se destacado na última década pelas suas inúmeras aplicações e resultados relevantes obtidos em diversas áreas para problemas envolvendo múltiplos critérios de avaliação: o NSGA-II e o SPEA2. Na Seção 6.1 conceitos básicos sobre otimização multiobjetivo são apresentados. Na Seção 6.2, o NSGA-II é apresentado. Na Seção 6.3 o SPEA2 é apresentado. Finalmente, o Algoritmo Evolutivo proposto com RNP é apresentado na Seção 6.4.

6.1 Conceitos Básicos

O processo de tomar decisões envolve vários fatores. Em certos casos, podem existir várias soluções que atendam a esses fatores, porém não se pode afirmar que, quantitativamente, uma seja melhor que outra. Um exemplo clássico, explorado em [48], aborda tomada de decisão de compra de um automóvel.

Suponha a procura por um automóvel com preço baixo e alto conforto. A Figura 6.1 ilustra algumas alternativas de escolha.

Figura 6.1 – Opções de compra de um automóvel envolvendo custo e conforto.

Assim, busca-se minimizar o custo total e maximizar o conforto. Pela Figura 6.1 tem-se 6 opções de compra. Intuitivamente, descarta-se a opção 5, pois possui o mesmo conforto que a opção 3, com maior custo. Analisando somente as soluções 1 e 6, não se pode afirmar qual das duas é melhor, pois a solução 6 tem maior conforto que a 1, porém maior custo. Ao analisar as soluções 2 e 6, observa-se que a solução 6 também deve ser descartada, pois possui menor conforto e maior custo em relação a solução 2. Tem-se então 4 opções (1, 2, 3 e 4) como boas alternativas para a compra. Dentre essas soluções, não se pode afirmar qual é a melhor, pois quando uma tem maior conforto, também tem maior custo e vice versa.

Diz-se que uma solução *domina* outra solução, quando é melhor que a outra em pelo menos 1 objetivo e não pior em todos os outros objetivos. Da Figura 6.1, tem-se que a solução 3 domina a solução 5 e, a solução 2 domina a solução 6. Portanto, as soluções 5 e 6 são *dominadas* e, as soluções 1, 2, 3 e 4 são *não dominadas* por nenhuma outra. Temos então o conjunto das soluções *não dominadas* (1, 2, 3, 4) e o conjunto das soluções *dominadas* (5, 6).

Assim, o conceito de dominância pode ser definido por:

Definição: a solução $x^{(1)}$ é dita *dominar* outra solução $x^{(2)}$, se satisfaz duas condições:

1. A solução $x^{(1)}$ é não pior que $x^{(2)}$ em todos objetivos;
2. A solução $x^{(1)}$ é estritamente melhor que $x^{(2)}$ em pelo menos um objetivo.

Conjunto Pareto-ótimo – seja o conjunto de soluções P envolvendo o espaço de busca total, o conjunto com as soluções não dominadas P' é chamado de conjunto de Pareto-ótimo.

Fronteira de Pareto – interligando-se os pontos do conjunto de Pareto ótimo obtém-se uma curva que é usualmente chamada de fronteira de Pareto (ver Figura 6.1). Deve-se observar que o conceito de fronteira de Pareto é precisamente definido no espaço contínuo.

6.2 Elitist Non-Dominated Sorting Genetic Algorithm

Um dos primeiros métodos a lidar com AEs multi-objetivos sem a necessidade de uma função ponderada foi proposto por Srinivas e Deb em 1995 [129]. O método foi denominado NSGA (do inglês *Non-dominated Sorting Genetic Algorithms*). Porém, no decorrer dos anos, esse método recebeu várias críticas: (i) alta complexidade computacional $O(MN^3)$ ao ordenar as soluções não dominadas, (onde M é o número de objetivos e N é o tamanho da população); (ii) ausência de um método de elitismo; (iii) necessidade da especificação pelo usuário de um parâmetro externo (σ_{share}) de compartilhamento. Devido algumas críticas, em 2000 Deb et al. [85], apresentaram uma versão melhorada do NSGA, onde as dificuldades do primeiro método foram amenizadas. Esse método foi denominado NSGA-II, do inglês, “Non-dominated Sorting GA-II”.

A seguir, apresenta-se o NSGA-II em 4 módulos: método de ordenação rápida por não dominância, preservação da diversidade, operador comparação e algoritmo principal. Parte do texto a seguir está baseado no trabalho original apresentado em [85].

6.2.1 Método de ordenação rápida por não dominância

Para ordenar uma população P de tamanho N de acordo com o nível de não-dominância, cada solução de P é comparada com parte dessa população por dominância. Cria-se um conjunto P' , inicialmente vazio, e coloca-se a primeira solução da população P em P' . Em seguida, cada solução p (da segunda solução em diante) é comparada uma a uma com todos os membros de P' . Se a solução p domina algum membro q de P' , então a solução q é removida de P' . Assim, todas as soluções não pertencentes ao conjunto de dominância serão excluídas de P' . Caso contrário, se a solução p é dominada por algum membro de P' , a solução p é ignorada. Se a solução p é não dominada por nenhum membro de P' , a mesma é acrescentada em P' . Dessa forma, o conjunto P' cresce com soluções não-dominadas. Ao final do processo, quando todas as soluções tiverem sido cheçadas, as remanescentes em P' constituem o conjunto de soluções não-dominadas (primeiro nível de não dominância). As soluções pertencentes ao segundo nível de não-dominância podem ser encontradas através do procedimento anterior, omitindo as soluções pertencentes ao primeiro nível. Da mesma forma, as soluções pertencentes ao nível k , são encontradas através desse procedimento, omitindo as soluções pertencentes aos níveis de não-dominância menores que k .

Para encontrar a primeira fronteira não dominada, a complexidade máxima requerida é $O(MN^2)$ [85], portanto, menor que a complexidade requerida pelo método antigo.

6.2.2 Preservação da diversidade

Durante o processo de convergência ao conjunto de Pareto ótimo, é esperado que o AE mantenha um bom espalhamento no conjunto de soluções obtido. O NSGA-II propõe a manutenção da diversidade sem a necessidade de nenhum parâmetro externo imposto pelo usuário e, sugere um método com melhor complexidade computacional comparado ao NSGA.

Para a estimação da densidade de soluções ao redor de uma solução particular na população, calcula-se a *crowding distance* (d)⁸ dessa solução, que representa uma estimativa do perímetro formado pelo cubóide cujos vértices são seus vizinhos mais próximos. Dessa forma, quanto maior o cubóide de i , mais afastada se encontra i de seus vizinhos. Para as soluções extremas em cada objetivo são atribuídos cubóides infinitos. A Figura 6.2 apresenta a *crowding distance* para a solução i e para as soluções extremas.

Figura 6.2 – Cálculo da crowding distance.

⁸ O termo *crowding distance* foi utilizado em inglês devido a dificuldade em encontrar uma tradução adequada para o mesmo.

Após a atribuição da *crowding distance* a todos os indivíduos de uma mesma fronteira da população, as mesmas são ordenadas, onde as soluções com menor *crowding distance* são as últimas da lista.

6.2.3 Operador comparação

O operador comparação ($\prec_c$) guia o processo de seleção do algoritmo em direção à fronteira de Pareto uniformemente espalhada. Assim, em um torneio entre duas soluções, uma solução i é considerada vencedora sobre uma solução j , se:

1. A solução i possui melhor nível de não dominância ($r_i < r_j$);
2. Se ambas as soluções estão no mesmo nível de dominância, porém i possui uma *crowding distance* maior, $d_i > d_j$.

Dessa forma, entre duas soluções com diferentes níveis de dominância, preferem-se as soluções no menor nível. Caso contrário, se ambas as soluções estão no mesmo nível, então se preferem soluções localizadas na região de menor densidade (maior *crowding distance*).

6.2.4 Algoritmo principal e complexidade computacional

Inicialmente, uma população pai P_0 é criada. A população é ordenada baseada no conceito de não dominância. Para cada solução é atribuído um *fitness* (ou *ranking*) igual ao seu nível de não dominância (1 é o melhor nível, 2 é o próximo melhor nível, e assim por diante). Assim, buscam-se soluções com menor *fitness*. Em seguida utilizam-se os operadores convencionais de seleção por torneio, recombinação e mutação para criar uma população filha Q_0 também de tamanho N . Como o elitismo é introduzido pela comparação da população corrente com as

melhores soluções não-dominadas encontradas anteriormente, o procedimento é diferente após a geração inicial. O Algoritmo 6 descreve uma geração do NSGA-II.

O procedimento do NSGA-II é demonstrado na Figura 6.3. A nova população P_{t+1} de tamanho N é agora utilizada para seleção, cruzamento e mutação para criar uma nova população Q_{t+1} também de tamanho N .

Algoritmo 6. NSGA-II.

ALGORITMO NSGA-II

ENQUANTO critério de parada não satisfeito, **FAÇA**

// Combine a população pai com a população de descendentes

$R_t := P_t \cup Q_t$

// $F := (F_1, F_2, \dots)$, todas as fronteiras não-dominadas de R_t

$F := \text{ordenação por não dominância de } (R_t)$

// Cria a população para próxima geração (P_{t+1})

$P_{t+1} := \emptyset$ e $i = 1$

// Até que a população pai esteja completa

ENQUANTO $|P_{t+1}| + |F_i| \leq N$, **FAÇA**

// Calcule a *crowding distance* em F_i

$F_i := d_i$

// Inclua a i -ésima fronteira não dominada na população pai

$P_{t+1} := P_{t+1} \cup F_i$

// Cheque a próxima fronteira para inclusão

$i := i + 1$

FIM

// Ordene em ordem decrescente utilizando $\prec_c$

Ordene ($F_i, \prec_c$)

// Escolha os $(N - |P_{t+1}|)$ elementos de F_i

$P_{t+1} := P_{t+1} \cup F_i[1 : N - |P_{t+1}|]$

// Use seleção, cruzamento e mutação para criar uma nova população Q_{t+1}

$Q_{t+1} := \text{gere_nova_população}(P_{t+1})$

// Incremente o contador de geração

$t := t + 1$

FIM

Finalmente, a complexidade computacional do pior caso para as operações básicas, descritas anteriormente, é apresentada a seguir:

1. Ordenação por não dominância é $O(M(2N)^2)$;
2. Atribuição da *crowding distance* é $O(M(2N)\log(2N))$;
3. Ordenação por $\prec_c$ é $O(2N\log(2N))$.

Nota-se que a maior complexidade computacional do NSGA-II, dentre os algoritmos listados acima, é $O(MN^2)$, requerido pelo algoritmo de ordenação por não-dominância.

Figura 6.3 – Esquema do modelo NSGA-II apresentado em [48].

6.3 Strength Pareto Evolutionary Algorithm

Assim como o NSGA-II surgiu de melhorias realizadas em sua primeira versão (NSGA), o SPEA2 [130], do inglês, “Strength Pareto Evolutionary Algorithm2”, surgiu para introduzir melhorias em sua primeira versão, SPEA, publicada em 1998 [131]. As principais melhorias propostas em [130] são relativas à: (i) atribuição de um *fitness* para as soluções; (ii) estimativa de densidade; (iii) método de truncamento para a população externa. Detalhes da primeira versão, denominada SPEA, podem ser encontrados em [131] e também em [48] páginas 249 a 258.

O SPEA2 é um AE elitista que utiliza duas populações em paralelo, uma população convencional P e uma população externa P' , ambas de tamanho fixo N e N' respectivamente. O elitismo garante que soluções com alta adequação não serão eliminadas durante a aplicação de um algoritmo de otimização. As Seções seguintes apresentam o SPEA2 dividido em suas principais partes: em 6.3.1 a atribuição de

fitness é discutida, em 6.3.2 a seleção de indivíduos e o método de truncamento para a população externa são apresentados e em 6.3.3 o algoritmo principal e a complexidade computacional são apresentados.

6.3.1 Método de atribuição de *fitness*

A atribuição de *fitness* no SPEA2 é realizada em dois passos:

Primeiro, em uma geração t , para cada indivíduo i na população externa P_t' e na população P_t é atribuído um valor $S(i)$, representando o número de soluções dominadas pela solução i :

$$S(i) = |\{j \mid j \in P_t + P_t' \wedge i \succ j\}|,$$

onde, $|\cdot|$ a cardinalidade do conjunto, $+$ representa a união dos conjuntos, t é a geração atual e, o símbolo $\succ$ corresponde a relação de dominância de Pareto, onde $i \succ j$ indica que uma solução i domina uma solução j ;

Segundo, o *fitness* $R(i)$ de um indivíduo i é determinado pela força com que as soluções em P_t e P_t' dominam a solução i

$$R(i) = \sum_{j \in P_t + P_t', j \succ i} S(j).$$

Note que o *fitness* nesse caso é para ser minimizado, isto é, $R(i) = 0$ corresponde a um indivíduo não-dominado, enquanto um alto valor de $R(i)$ significa que i é dominada por muitos indivíduos.

Embora a atribuição de um *fitness* $R(i)$ ordene as soluções baseadas no conceito de dominância de Pareto, ela pode falhar quando muitos indivíduos não são dominados uns pelos outros. Nesse caso, foi incorporada a informação de densidade $D(i)$ entre os indivíduos com idêntico $R(i)$. A técnica de informação de densidade proposta em [130] é uma adaptação do método do k -ésimo vizinho mais

próximo [132] e pode ser estimada como o inverso da distância do indivíduo i ao k -ésimo vizinho mais próximo. Assim, o *fitness* total de um indivíduo i é definido por:

$$F(i) = R(i) + D(i).$$

Algumas aplicações necessitam de informações de densidade $D(i)$ com maior precisão. Nesses casos, em [130] foi sugerido que se calcule, para cada indivíduo i , a distância (no espaço dos objetivos) para todos os indivíduos j tanto na população normal quanto na população externa em seguida armazene-as em uma lista, ordenando-as em ordem crescente. O k -ésimo elemento apresenta a distância procurada, denotada por σ_i^k , onde k pode ser obtido pela raiz quadrado da soma do tamanho das populações ($k = \sqrt{N + N'}$). Assim, a densidade $D(i)$ correspondente ao indivíduo i é calculada por

$$D(i) = \frac{1}{\sigma_i^k + 2}.$$

No denominador, é adicionado 2 para garantir que seu valor seja maior que zero e que $D(i) < 1$.

6.3.2 Seleção e truncamento

No método de seleção, o primeiro passo é copiar todos os indivíduos não-dominados de P_t e de P_t' para a população externa da próxima geração P_{t+1}' . Nesse caso, podem ocorrer 3 possibilidades:

1. O número de indivíduos no conjunto não-dominado é exatamente o mesmo da população externa ($|P_{t+1}'| = N'$);
2. O número de indivíduos no conjunto não-dominado é menor que o tamanho da população externa ($|P_{t+1}'| < N'$);

3. O conjunto de indivíduos não-dominados excede o tamanho da população externa ($|P'_{t+1}| > N'$).

No primeiro caso, o processo de seleção está completo. No segundo caso, os melhores $N' - |P'_{t+1}|$ indivíduos dominados, incluindo a população regular e população externa na geração anterior, são copiados para a nova população externa. Finalmente, no terceiro caso, um procedimento de truncamento da população externa é executado para remover indivíduos de P'_{t+1} até ($|P'_{t+1}| = N'$). O método de truncamento é explicado a seguir.

Em cada iteração, um indivíduo i é selecionado para ser removido se:

- $\sigma_i^k = \sigma_j^k$ para todo valor de k no intervalo $0 < k < |P'_{t+1}|$ e $j \in P'_{t+1}$ ou
- $\sigma_i^l = \sigma_j^l$ e $\sigma_i^k < \sigma_j^k$ para todo valor de l no intervalo $0 < l < k$ e para algum valor de k no intervalo $0 < k < |P'_{t+1}|$.

Onde, σ_i^k denota a distância (no espaço de objetivos) do indivíduo i ao k -ésimo vizinho mais próximo. Em outras palavras, o indivíduo que apresentar a menor distância em relação a outro indivíduo é selecionado para ser removido em cada estágio.

6.3.3 Algoritmo principal e complexidade computacional

A complexidade computacional para a atribuição de *fitness* é dominada pela complexidade do procedimento de estimação de densidade ($O(M^2 \log M)$), enquanto o cálculo de S e R tem complexidade $O(M^2)$, onde $M = N + N'$.

Embora, para o pior caso, a complexidade computacional do operador de truncamento é $O(M^3)$, em média, a complexidade será menor que $O(M^2 \log M)$.

Isso porque geralmente os indivíduos diferem consideravelmente do segundo ou terceiro vizinho mais próximo. Assim, o método de ordenação governa a complexidade total do SPEA2.

Os passos do SPEA2 são apresentados no Algoritmo 7. Considere como entrada o tamanho da população (N), tamanho da população externa (N'), máximo número de gerações (T) e como dados de saída o conjunto com as soluções não-dominadas (A).

Algoritmo 7. SPEA2.

ALGORITMO SPEA2

// **1. Inicialização:** Gere uma população inicial P_0 e crie uma população externa P'_0 vazia
 $P'_0 := \emptyset$ e $t := 0$

// **2. Atribuição de fitness:** Calcule o *fitness* dos indivíduos em P_t e P'_t

$F_i := S(i) + D(i)$

// **3. Seleção e Truncamento:** Copie os indivíduos não-dominados de P_t e P'_t para P'_{t+1}

$P'_{t+1} := \text{soluções_não-dominadas_}(P_t + P'_t)$

// Se o número de elementos em P'_{t+1} exceder N'

SE $|P'_{t+1}| > N'$ **FAÇA**

 // Aplique operador de truncamento

 Truncamento_ P'_{t+1}

// Se número de elementos em P'_{t+1} for menor que N'

SE $|P'_{t+1}| < N'$ **FAÇA**

 // Complete P'_{t+1} com elementos dominados de P_t e P'_t

$P'_{t+1} := P'_{t+1} + \text{soluções_dominadas_}(P_t + P'_t)$

// **4. Critério de Parada:** Se algum critério de parada é satisfeito, forneça saída

SE $t \geq T$ ou outro critério de parada é satisfeito

 // Apresente solução e pare o processo

$A := \text{indivíduos_não_dominados_de_}P'_{t+1}$

PARE

// **5. Variação:** Aplique operador para reprodução

SENÃO

 // Aplique operadores de recombinação e mutação em P'_{t+1}

$P_{t+1} := \text{Operadores_}P'_{t+1}$

// Incremente contador de gerações e vá para a o **passo 2**

$t := t + 1$

vá para **passo 2**

FIM

6.4 Algoritmo Evolutivo Multi-objetivo Proposto

Esta Seção apresenta o AE proposto, chamado de **AE Multi-objetivo** em Tabela (AEMT, ver Seção 6.4.1). Este algoritmo busca o restabelecimento de energia em SDRs de grande porte após a ocorrência de uma ou múltiplas faltas. Devido a radialidade do Sistema, após a ocorrência de uma ou mais faltas, as áreas fora de serviço representam subárvores de grafo. Assim, os Operadores da RNP (ver Capítulo 4) podem ser aplicados para conectá-la(s) a outra árvore (alimentador), quando possível. Cada aplicação de um dos Operadores gera um indivíduo novo na população, isto é, uma floresta de grafo modelado em RNP que represente um SDR sem o(s) setor(es) em falta (isolados) e com a área fora de serviço re-conectada a algum ponto energizado do sistema. Esse indivíduo pode violar uma ou mais restrições operacionais do sistema, potencialmente inviabilizando o uso dessa configuração na prática. Assim, é necessário encontrar uma nova configuração que melhore a qualidade do sistema. O AEMT inicia a busca por melhores soluções partindo de configurações geradas a partir da configuração original com o setor em falta isolado e, a partir dessas, é capaz de produzir soluções mais adequadas para utilização.

Dependendo do mecanismo de seleção adotado, os AEs podem ser atraídos para ótimos locais. Quando busca resolver problemas com múltiplos objetivos por meio de uma função agregação (função ponderada), o desempenho do AE é diretamente afetado pela escolha dos pesos (δ 's) para a função. Inadequados δ 's podem direcioná-lo a convergir para soluções ruins. Além disso, determinados δ 's podem ser bons para uma rede específica e muito ruins para outras.

6.4.1 Algoritmo Evolutivo Multi-objetivo em Tabela

O AE Multi-objetivo proposto trabalha com várias subpopulações em paralelo armazenadas em tabela [32, 84], onde os melhores indivíduos para cada característica do problema são armazenados em sua respectiva subpopulação. Por essa razão, o AE desenvolvido é chamado de **AE Multi-objetivo em Tabela**. O AEMT é programado utilizando a RNP.

Uma subpopulação é criada para armazenar indivíduos avaliados por uma função agregação, também é conhecido como método da soma ponderada [48, 133]. Cada subpopulação armazena indivíduos de modo a minimizar características como: (i) perdas de potência; (ii) menor tensão na rede, reduzindo assim a queda de tensão; (iii) carregamento da rede; (iv) carregamento das subestações; (v) função agregação. Neste trabalho utilizou-se apenas uma tabela de função agregação.

O indivíduo selecionado para a reprodução pode ser proveniente de qualquer subpopulação da tabela. Essa estratégia de seleção aumenta a diversidade entre os indivíduos que reproduzem de forma que as características de um indivíduo de uma subpopulação possam migrar para as demais subpopulações da tabela. Em consequência, aumenta-se a possibilidade do algoritmo escapar de ótimos locais, aproximando-se de soluções com avaliações próximas de um ótimo global na fronteira de Pareto-ótima (ver Seção 6.1). O método de seleção de indivíduos para a reprodução é discutida mais adiante.

Alguns parâmetros são importantes para o desempenho deste AEMT:

1. O tamanho de uma subpopulação S_{P_i} , que indica o número máximo de indivíduos que podem permanecer na subpopulação P_i de uma geração para outra;
2. O número máximo de gerações (g_{max}).

Soluções geradas pelo AEMT podem ser armazenadas ou descartadas, dependendo do grau de adaptação do indivíduo a cada objetivo do problema (característica do problema em uma subpopulação P_i). Indivíduos são gerados pelo Operador 2 a partir da configuração inicial F_0 (com o setor em falta isolado) até que todas as subpopulações estejam completas. No processo de seleção de sobreviventes, um novo indivíduo é acrescentado a uma subpopulação P_i se sua adequação ao objetivo de P_i for melhor que pelo menos um indivíduo da mesma. O mesmo indivíduo pode ser incluído em mais de uma tabela de acordo com esse critério de seleção. Como a população é estacionária, os novos indivíduos substituem os piores. Nesse caso, a adequação do indivíduo é um vetor de seus valores relativos a cada objetivo (perdas, número de chaveamentos, etc.) ou restrições (carregamento nas linhas, queda de tensão, etc.).

Os Operadores são selecionados para a reprodução segundo um ajuste dinâmico dos dois. O algoritmo inicia sua busca utilizando a mesma taxa de probabilidade para ambos Operadores (por exemplo, $P_1 = P_2 = 0,50$). Suponha que o Operador 1 tenha sido escolhido para gerar um novo indivíduo. Se o indivíduo gerado tiver sucesso, ou seja, for acrescentado a uma ou mais subpopulações, aumenta-se P_1 para 0,51 e, como consequência, P_2 é reduzido para 0,49 e, assim por diante. Esse ajuste dinâmico do processo de escolha dos Operadores melhora o desempenho do algoritmo consideravelmente, conforme mostrado em testes na Seção 7.1.

A avaliação dos indivíduos é realizada através do fluxo de carga específico para SDRs apresentado no Capítulo 5. A cada novo indivíduo gerado, a rotina de fluxo de carga é executada para o par de alimentadores modificados do sistema.

6.4.2 Algoritmo principal do AEMT

O pseudocódigo do AEMT é apresentado no Algoritmo 8. Vale ressaltar que, diferentemente do NSGA-II e SPEA2 que geram vários indivíduos por iteração, no AEMT apenas 1 indivíduo é gerado por iteração.

Algoritmo 8. Pseudocódigo para o AEMT.

ALGORITMO AEMT

```
// inicia o contador de gerações
 $g := 0$ ;
// gera subpopulações iniciais  $P_i(g_0)$  a partir da floresta original  $F_0$ 
 $P(g) := \text{POP\_INIC}(P_i(g_0, F_0))$ ;
// avalia os indivíduos das subpopulações iniciais
AVALIE ( $P(g)$ );
// teste do critério de parada ( $g_{max}$ )
ENQUANTO critério de parada não atingido FAÇA
    // aleatoriamente selecione uma subpopulação ( $P_i$ ).
     $P_i := \text{ALEATORIAMENTE-SELECIONE}(pop.)$ ;
    // aleatoriamente selecione um indivíduo ( $F_s$ ) de  $P_i$ .
     $F_s := \text{ALEATORIAMENTE-SELECIONE}(P_i(g))$ ;
    // decide se aplica Operador 1 ou Operador 2
     $OP := \text{DECIDE\_OPERADOR}(op1, op2)$ ;
    // aplicar OP para produzir um novo indivíduo  $F_g$  de  $F_s$ 
     $F_g := OP(F_s)$ ;
    // avaliação do novo indivíduo  $F_g$ 
    AVALIE( $F_g$ );
    // selecione sobreviventes entre  $P(g)$  e  $F_g$ 
     $P(g + 1) := \text{ALTER\_POP}(P(g), F_g)$ ;
    //incrementa o contador de gerações
     $g := g + 1$ ;
```

FIM

AEs de uma forma geral foram apresentados neste Capítulo, com destaque para o NSGA-II, SPEA2 e o AEMT. No próximo Capítulo, resultados de testes do AEMT utilizando o SDR real da cidade de São Carlos-SP são apresentados, além de comparações de resultados entre esses algoritmos.

7 Resultados Experimentais

Resultados de simulação computacional são apresentados neste Capítulo. Testes foram realizados em um SDR real de grande porte sem simplificações, ou seja, todas as linhas, chaves e barras foram consideradas. A rede é constituída de 3.860 barras, 533 setores, 632 chaves sendo 509 chaves NF e 123 chaves NA, 3 subestações e 23 alimentadores. A subestação Bela Vista possui 2 transformadores de 25MVA; a subestação São Carlos tem 2 transformadores de 25MVA e a subestação Paraíso tem 1 transformador de 25MVA. Os tipos de cabos existentes na rede são A47, A33, A10, A04 e A02. Esse SDR corresponde à rede da cidade de São Carlos-SP (SDR-SC) no ano de 1994. Para verificar que o AEMT pode ser aplicado em SDRs maiores que o SDR-SC não incorrendo no problema de explosão combinatorial, testes foram realizados para reconfiguração de redes utilizando o SDR-SC aumentado, ou seja, com tamanho duplicado, quadruplicado e octuplicado (ver Seção 7.6). Foram também realizados testes com o SDR da *Taiwan Power Company* (TPC), consistindo em um sistema trifásico equilibrado de 11,4kV com 11 alimentadores, 83 chaves NF, 13 chaves NA, com 28,35MW de carga ativa total e 20,70MVA_r de carga reativa total.

A metodologia proposta pode ser utilizada em problemas de reconfiguração de redes com diversos propósitos, dentre os quais se destacam: restabelecimento de energia após uma contingência, redução de perdas de potência, de queda de tensão, de carregamento na rede e nas subestações, balanceamento de cargas nos

transformadores, expansão da rede e alocação ótima de componentes na rede tais como capacitores, chaves automáticas, etc. Os testes apresentados concentram-se no restabelecimento de energia após a ocorrência de uma única falta e de múltiplas faltas simultâneas. Optou-se pela maior exploração deste tipo de problema devido sua maior complexidade computacional e pela necessidade de resposta em tempo real. De certa forma, pode-se dizer que os demais problemas de reconfiguração são casos particulares do problema de restabelecimento.

O AEMT foi implementado utilizando um computador com processador Pentium IV, 2,1GHz, 4Gbytes de memória RAM, Sistema Operacional Linux, versão Ubuntu 8.04 e compilador de linguagem C gcc.

Na Seção 7.1, um estudo de seleção dos Operadores da RNP é apresentado. Testes com uma única falta empregando o modelo mono-objetivo do AE com a RNP são apresentados na Seção 7.2. Testes com múltiplas faltas também empregando o modelo mono-objetivo do AE usando a RNP são apresentados na Seção 7.3. Testes com uma única falta usando o modelo multi-objetivo, isto é o AEMT são apresentados na Seção 7.4. Testes realizados com múltiplas faltas empregando o AEMT são apresentados na Seção 7.5. Na Seção 7.6 são apresentados testes com SDRs sintetizados a partir do SDR-SC, com tamanhos que correspondem a duplicação do SDR-SC várias vezes, possibilitando a verificação do comportamento do AEMT com o aumento da escala do problema. Resultados de testes utilizando o SDR da TPC são apresentados na Seção 7.7. Na Seção 7.8 é apresentada a comparação de resultados obtidos por dois métodos de fluxos de carga, um deles utilizando o modelo de potência constante e outro com modelo de corrente constante. Finalmente, na Seção 7.9, são apresentados os resultados obtidos pelo NSGA-II e SPEA2 (AEs multiobjetivos apresentados no Capítulo 6)

adaptados com a RNP, bem como a comparação desses resultados com os do AEMT.

7.1 Diversas formas de utilização dos Operadores da RNP

Conforme descrito na Seção 6.4.1, a seleção dos Operadores é realizada segundo um ajuste dinâmico entre o Operador 1 e o Operador 2. Existem diversas formas de seleção dos Operadores. Na Tabela 7.1, a média de perdas de potência em soluções obtidas de 15 execuções do AEMT para falta única é apresentada para cada um dos seguintes casos: (i) uso exclusivo do Operador 1, (ii) uso exclusivo do Operador 2, (iii) seleção aleatória entre os Operadores 1 e 2 e (iv) ajuste dinâmico da probabilidade de seleção de cada Operador, (ver Seção 6.4.1).

Tabela 7.1 – Média de perdas de potência nas soluções obtidas de 15 execuções do AEMT para falta única.

Seleção	Perdas de Potência [kW]	Tempo de Processamento [s]	Iteração do Melhor Indivíduo
Somente Operador 1	293,08	1,94	25.309,07
Somente Operador 2	291,36	1,95	27.325,13
Aleatória	283,47	2,10	27.628,80
Ajuste Dinâmico	281,66	2,10	26.479,73

Na Figura 7.1, o *fitness* do melhor indivíduo da subpopulação de perdas de potência é apresentado ao longo das iterações para os 4 casos de testes descritos na Tabela 7.1. Esse gráfico ilustra o perfil de convergência mais freqüente ocorrido nas 15 execuções do AEMT para cada caso. Pela Figura 7.1 é possível observar que o caso utilizando o ajuste dinâmico é o que possibilitou tirar o melhor proveito dos Operadores 1 e 2 da RNP. A média do tempo de processamento do AEMT utilizando o ajuste dinâmico foi igual para utilização do ajuste aleatório e levemente superior para os outros 2 casos, porém as perdas de potência, em média, são menores.

Figura 7.1 – Perdas de potencia obtidas usando os Operadores 1 e 2 separadamente, seleção aleatória e ajuste dinâmico de probabilidade de seleção de cada Operador.

Nesse trabalho, utilizou-se 100% como o limite de capacidade da rede ($x_j / \bar{x}_j \leq 100\%$) e das subestações ($b_i / \bar{b}_i \leq 100\%$), ver Capítulo 3. Considerou-se 7% como a máxima taxa de queda de tensão ($v_i / v_b \geq 93\%$) permissível, conforme especificação da ANEEL [134] para tensões entre 1kV e 69kV. Para todos os casos apresentados nas Seções 7.1 a 7.5, foram utilizadas 30.000 iterações como o número máximo de indivíduos gerados (g_{max}). No restante do Capítulo 7, valores diferentes de 30.000 para g_{max} são informados quando ocorrem. Nas Seções 7.1 a 7.7 e 7.9 foi utilizado o fluxo de carga com o modelo de corrente constante.

7.2 Modelagem mono-objetivo para falta única

Testes foram realizados para restabelecimento de energia após a ocorrência de uma única falta. Nesses testes o número total de indivíduos gerados g_{max} é

30.000 e uma população de tamanho 20 é empregada. Verificou-se experimentalmente que testes com g_{max} maior que 30.000 não produzem melhorias significativas para os casos investigados. Nesse trabalho, os indivíduos da população inicial são gerados a partir da configuração com o setor em falta eliminado do SDR.

O problema foi modelado utilizando um único objetivo. Os testes foram agrupados em 5 casos de acordo com o objetivo empregado:

- 1º Caso: redução de perdas;
- 2º Caso: redução da maior queda de tensão;
- 3º Caso: redução do maior carregamento da rede;
- 4º Caso: redução do maior carregamento das Subestações;
- 5º Caso: minimização da função agregação.

No último caso, a seguinte função agregação foi utilizada $f(x) = \delta_1 x_1 + \delta_2 x_2 + \delta_3 x_3 + \delta_4 x_4 + \delta_5 x_5$, onde x_1, x_2, x_3, x_4 e x_5 correspondem, respectivamente, à somatória das perdas de potência em kW, ao número de chaveamentos, ao máximo carregamento da rede em p.u., ao máximo carregamento das subestações em p.u., à máxima queda de tensão em p.u., e δ_i é o peso relativo ao objetivo x_i . Os seguintes parâmetros foram utilizados:

$$\delta_1 = 1;$$

$$\delta_2 = 1;$$

$$\delta_i = \begin{cases} 100, & \text{se pelo menos um elemento da rede tiver a restrição relativa a } x_i \text{ violada} \\ 0, & \text{caso contrário;} \end{cases}$$

$$\text{para } i = 3, 4, 5.$$

A função agregação pode ser ajustada de acordo com a necessidade de cada rede de distribuição. Os parâmetros acima foram utilizados em todos os testes que envolveram a função agregação. Apesar dessa função envolver vários objetivos, o modelo matemático resultante do problema possui apenas uma função mono-objetivo e, portanto, técnicas de otimização mono-objetivo podem ser empregadas nesse caso. Deve-se observar também que o modelo de fluxo de carga de corrente constante (ver Seção 7.8) foi utilizado nos testes apresentados nas Seções 7.1 a 7.7 e na Seção 7.9.

Considere que uma falta tenha ocorrido no setor 504, interrompendo o serviço ao maior alimentador do sistema com 17 setores, 231 barras e 33 chaves NF (ver Figura 7.2). Os valores médios dos 20 primeiros indivíduos gerados (compondo a população inicial) a partir da configuração em falta são mostrados a seguir:

- Somatória de perdas de potência igual a 404,58kW;
- Maior queda de tensão igual a 4,96 %;
- Maior carregamento na rede igual a 139,60%;
- Maior carregamento nas subestações igual a 52,74%.

Nota-se que, pela média dessas configurações o carregamento da rede ultrapassa os 100%, não respeitando uma restrição operacional do sistema.

A Tabela 7.2 sintetiza os resultados de simulação computacional para 15 execuções do algoritmo, onde as características do melhor indivíduo para cada caso citado acima estão armazenados em cada linha da Tabela. Os resultados destacados em negrito indicam o melhor valor encontrado para aquela característica nos 5 casos avaliados. Um valor sublinhado indica que o mesmo não atende à restrição correspondente.

Figura 7.2 – Destaque do alimentador 23 do SDR-SC.

Tabela 7.2 – Média de 15 execuções do AE com RNP com modelagem mono-objetivo para falta única.

População	Perdas de Potência [kW]	Queda Tensão [%]	Carregamento da Rede [%]	Carregamento das Subestações [%]	Número de Chaveamentos
1º caso	284,14	3,18	72,23	58,79	93,80
2º caso	610,39	3,17	<u>169,86</u>	57,02	76,20
3º caso	469,49	3,89	72,76	54,03	69,80
4º caso	6678,40	<u>16,58</u>	<u>717,77</u>	44,75	106,60
5º caso	295,28	3,25	76,93	54,12	23,00

A média de tempo de computação para gerar os 30.000 indivíduos nos 5 casos (ver Tabela 7.2) foi de 2,46s e para encontrar a melhor solução (iteração 11.597) foi de 0,95s. Note que no 3º caso, devido a utilização de uma função com um único objetivo, em várias execuções, o algoritmo convergiu para um ótimo local, uma vez que a média do carregamento da rede encontrada no 1º caso é menor que a do 3º caso. O 4º caso produz a configuração com o menor carregamento da subestação, porém com os piores valores para os demais aspectos avaliados. Observa-se também que o 1º caso (objetivo de redução de perdas) apresenta resultados próximos aos melhores resultados encontrados para quase todas as características, com exceção do número de chaveamentos. Isso indica que o objetivo de redução de perdas pode direcionar a busca para soluções que possuem um consenso entre a maioria dos objetivos e restrições. Esse é um dos fatores que motivam o uso do objetivo de redução de perdas, mesmo no problema de restabelecimento de energia, apesar desse objetivo não ser, em geral considerado para esse problema na literatura.

7.3 Modelagem mono-objetivo para múltiplas faltas

Resultados de testes computacionais com um objetivo e múltiplas faltas para o SDR-SC são apresentados nesta Seção. Além do setor 504 (ver Figura 7.2) que isola o alimentador 23 (o maior do sistema), os setores 182 e 486 dos alimentadores 6 e 22 (ver Figura 7.3 e Figura 7.4) respectivamente, foram isolados do sistema após a ocorrência de faltas. Os valores médios das características das 20 topologias de rede (população inicial) geradas após isolar os setores defeituosos e conectar as áreas a jusante a um alimentador vizinho são apresentadas a seguir:

- Somatória de perdas de potência igual a 425,83kW;

- Maior queda de tensão igual a 4,72 %;
- Maior carregamento na rede igual a 111,38%;
- Maior carregamento de subestações igual a 56,99%.

Figura 7.3 – Alimentador 6 do SDR-SC.

Figura 7.4 – Alimentador 22 do SDR-SC.

A Tabela 7.3 sintetiza os resultados obtidos em 15 execuções do algoritmo. As características do melhor indivíduo para cada característica avaliada estão destacadas em **negrito** e os resultados que violem restrições operacionais estão sublinhados. A média de tempo de computação para gerar 30.000 indivíduos para os

5 casos apresentados foi de 2,40s, enquanto o tempo de computação para atingir a convergência média (iteração 10.584) foi de 0,85s.

Tabela 7.3 – Média de 15 execuções do AE com RNP com modelagem mono-objetivo para múltiplas faltas.

População	Perdas de Potência [kW]	Queda Tensão [%]	Carregamento da Rede [%]	Carregamento de Subestações [%]	Número de Chaveamentos
1º caso	287,59	3,19	72,69	60,33	96,87
2º caso	612,78	3,18	<u>157,13</u>	57,45	71,00
3º caso	433,92	3,50	64,85	58,87	56,60
4º caso	6372,56	<u>15,80</u>	<u>693,88</u>	45,01	108,60
5º caso	300,87	3,25	76,45	59,18	29,27

Analisando os valores médios em negrito, podemos destacar que os melhores valores encontrados pelo AE utilizando modelagem mono-objetivo para múltiplas faltas correspondem exatamente às características minimizadas naquele caso, diferentemente do que ocorreu para o estudo com o modelo mono-objetivo para uma única falta (ver Seção 7.2). No 2º e 4º casos, o carregamento da rede é violado. Os valores apresentados no 4º caso são os piores encontrados em todas as características, com exceção do carregamento da subestação, que corresponde ao objetivo a ser minimizado nesse caso.

7.4 Modelagem multi-objetivo para falta única

Conforme apresentado na Seção 6.1, o AEMT trabalha com tabela de subpopulações em paralelo e não com uma população única, como fazem os AEs convencionais. Cada subpopulação armazena os 5 melhores indivíduos para cada uma das seguintes características: (i) perdas de potência; (ii) queda de tensão; (iii) carregamento da rede; (iv) carregamento das subestações; (v) função agregação.

Figuras 7.5 a 7.10 apresentam as características do AEMT para a melhor solução na subpopulação da função agregação a cada iteração. A Figura 7.5 mostra a redução total de perdas de potência para o melhor indivíduo da função agregação a cada iteração. A Figura 7.6 mostra os valores correspondentes de queda de tensão. Os valores correspondentes de carregamento na rede e carregamento das subestações são apresentados na Figura 7.7 e Figura 7.8 respectivamente. A Figura 7.9 mostra os valores correspondentes de redução do número de chaveamentos. Por último, a Figura 7.10 mostra o perfil de minimização da função agregação ao longo das iterações.

Figura 7.5 – Falta única, perdas totais.

Figura 7.6 – Falta única, menor tensão.

Figura 7.7 – Falta única, carregamento da rede.

Figura 7.8 – Falta única, carregamento das subestações.

Figura 7.9 – Falta única, número de chaveamentos.

Figura 7.10 – Falta única, função agregação.

A partir da Figura 7.5 a Figura 7.10, é possível observar o problema dos objetivos conflitantes. Note que por volta da iteração de número 5.000 houve alterações bruscas nas características do melhor indivíduo da função agregação. Melhorias consideráveis aconteceram na somatória das perdas, menor tensão e carregamento da rede; por outro lado, houve um aumento no carregamento das subestações e nas operações de chaveamentos. O número de manobras passou de aproximadamente 25 para quase 65 nessa etapa da evolução das soluções, porém houve redução considerável na função agregação. Por fim, nas iterações seguintes, o carregamento das subestações e o número de chaveamentos são significativamente reduzidos.

A Tabela 7.4 mostra o valor médio (obtido após 15 execuções do AEMT) do melhor indivíduo para cada subpopulação ao final do processo. A primeira linha da Tabela 7.4 apresenta o resultado obtido, em média, para o melhor indivíduo na subpopulação de redução de perdas. As perdas encontradas para esse indivíduo foram menores que as perdas de qualquer outro indivíduo de outras subpopulações. O mesmo ocorre para o melhor indivíduo das outras subpopulações, que possuem o menor valor para a característica que sua subpopulação representa, o que é destacado pelos números em negrito. O tempo de processamento para gerar os 30.000 indivíduos avaliados foi em média de 2,19s e o tempo médio para encontrar a melhor solução (indivíduo da iteração 26.531) em termos de função agregação foi de 1,90s.

A modelagem multi-objetivo para falta única possibilitou encontrar resultados melhores que os encontrados quando a modelagem mono-objetivo foi utilizada. Essa afirmação pode ser verificada comparando a Tabela 7.2 com a Tabela 7.4.

Tabela 7.4 – Média de 15 execuções do AEMT para falta única.

Subpopulação	Perdas de Potência [kW]	Queda de Tensão [%]	Carregamento Da Rede [%]	Carregamento da subestação [%]	Número de Chaveamentos
Somatória das Perdas de Potência	282,82	3,18	72,12	57,34	95,67
Queda de Tensão	330,86	3,13	82,28	55,15	73,27
Carregamento da Rede	315,65	3,22	60,25	55,66	73,27
Carregamento das Subestações	8097,65	<u>18,94</u>	<u>701,99</u>	44,52	107,93
Função Agregação	294,08	3,24	76,69	54,31	24,87

Comparando-se os resultados médios apresentados na Tabela 7.4 com os da Tabela 7.2, nota-se que os resultados em negrito da modelagem multi-objetivo, com exceção dos resultados da Função Agregação, são melhores que os resultados em negrito da modelagem mono-objetivo. Os melhores resultados encontrados com o AEMT demonstram a importância da divisão da população em subpopulações, aumentando a diversidade entre os indivíduos que reproduzem, vencendo ótimos locais e possibilitando a obtenção de melhores soluções. Embora do ponto de vista prático não faça tanta diferença, em termos computacionais há a desvantagem de um maior tempo médio de convergência do algoritmo: o modelo mono-objetivo requer 0,95s; enquanto o modelo multi-objetivo precisa de 1,90s.

7.5 Modelagem multi-objetivo para múltiplas faltas

Outros testes realizados com o AEMT envolveram múltiplas faltas. Três faltas simultâneas foram simuladas nos setores 182, 486 e 504 (ver Figura 7.2, Figura 7.3 e Figura 7.4).

A metodologia mostrou-se adequada para resolver problemas com múltiplas faltas. Resultados para esse caso são ilustrados pelas Figura 7.11 a Figura 7.16 que mostram a evolução de cada característica de avaliação pelo AEMT.

Figura 7.11 – Múltiplas faltas, perdas totais.

Figura 7.12 – Múltiplas faltas, menor tensão.

Figura 7.13 – Múltiplas faltas, carregamento da rede.

Figura 7.14 – Múltiplas faltas, carregamento das subestações.

Figura 7.15 – Múltiplas faltas, número de chaveamentos.

Figura 7.16 – Múltiplas faltas, função agregação.

O perfil de convergência do melhor indivíduo ao longo das iterações em cada subpopulação é apresentado nas Figuras 7.17 a 7.20. Para a subpopulação de minimização da função agregação, o perfil do melhor indivíduo corresponde ao apresentado na Figura 7.16.

Figura 7.17 – Subpopulação, perdas totais.

Figura 7.18 – Subpopulação, menor tensão.

Figura 7.19. – Subpopulação, carregamento da rede.

Figura 7.20 – Subpopulação, carregamento das subestações.

A Tabela 7.5 destaca os valores das características do melhor indivíduo encontrado em cada subpopulação. O tempo de processamento para gerar os 30.000 indivíduos, nesse caso, foi de 2,24s e o tempo médio para atingir a melhor solução encontrada (iteração 27.457) em termo de função agregação foi de 2,02s.

Tabela 7.5 – Média de 15 execuções do AEMT para múltiplas faltas.

Subpopulação	Perdas de Potência [kW]	Queda de Tensão [%]	Carregamento Da Rede [%]	Carregamento de Subestação [%]	Número de Chaveamentos
Somatória das Perdas de Potência	286,90	3,17	71,47	60,07	96,33
Queda de Tensão	351,50	3,14	92,34	58,29	56,87
Carregamento da Rede	347,56	3,30	62,78	57,83	61,53
Carregamento das Subestações	6971,99	<u>17,18</u>	<u>686,55</u>	44,89	108,60
Função Agregação	302,46	3,25	77,71	59,33	28,20

Entre os resultados em negrito encontrados pelas modelagens mono-objetivo e multi-objetivo para múltiplas faltas, apresentados na Tabela 7.3 e Tabela 7.5, respectivamente, os melhores foram encontrados na modelagem multi-objetivo, onde o AEMT foi utilizado. Porém, o tempo médio de processamento para gerar os 30.000 indivíduos pelo algoritmo mono-objetivo foi de 0,85s, enquanto para o multi-objetivo foi de 2,02s.

Em síntese, pode-se dizer que resultados apresentados nas Seções 7.2 a 7.5 mostram que AEs utilizando a RNP são capazes de resolver problemas de restabelecimento para um SDR relativamente grande (SDR-SC, com 3860 barras e 632 chaves) em casos de uma ou múltiplas faltas. Em geral, as melhores soluções encontradas são obtidas em menos iterações quando a modelagem mono-objetivo é utilizada, em comparação com a modelagem multi-objetivo. Por outro lado, a possibilidade do algoritmo convergir para ótimos locais é maior naquela modelagem. Dessa forma, o AEMT, através de suas tabelas, proporciona ferramentas para escapar desses ótimos locais, e possibilita encontrar soluções levemente melhores, por meio de uma melhor exploração do espaço de busca.

Dessa forma, mostrou-se que o AEMT é uma ferramenta capaz de lidar com redes de tamanho real em tempo de processamento reduzido, possibilitando sua aplicação em problemas que requerem soluções em tempo real para casos de uma ou de múltiplas faltas envolvendo grandes áreas desenergizadas.

7.6 Resultados com sistemas construídos a partir do SDR-SC

Nos testes a seguir, o tamanho do SDR-SC foi duplicado, quadruplicado e octuplicado a fim de verificar o desempenho do AEMT com o aumento do tamanho da rede. Os testes são relativos a reconfiguração do SDR para melhorar a qualidade do serviço de fornecimento de energia (redução de perdas, queda de tensão, carregamento da rede e das subestações), sem a ocorrência de faltas.

O Sistema 2 originou-se do SDR-SC (chamado nesta Seção de Sistema 1), que consiste do Sistema 1 duplicado e a adição de 13 chaves NA conectando o sistema original com sua parte duplicada. Da mesma forma, o Sistema 2 foi duplicado e 23 chaves NA foram adicionadas, originando o Sistema 3. O Sistema 4 foi gerado a partir do Sistema 3 através do mesmo procedimento e 12 chaves NA foram adicionadas. Os dados de cada SDR descrito anteriormente são apresentados na Tabela 7.6.

Tabela 7.6 – Dados dos Sistemas construídos a partir do SDR-SC (Sistema 1).

SDR	Barras	Setores	Chaves NF	Chaves NA	Subestações	Alimentadores
Sistema 1	3.860	533	509	123	3	23
Sistema 2	7.720	1.066	1.018	259	6	46
Sistema 3	15.440	2.132	2.036	541	12	92
Sistema 4	30.880	4.264	4.072	1.094	24	184

A Figura 7.21 mostra o tempo necessário para o AEMT gerar 10.000 indivíduos para os Sistemas 1, 2, 3 e 4. A curva em vermelho destaca o tempo de processamento do AEMT para a construção de novas configurações. Em verde, mostra-se a curva limitante superior: $f(n)=0,019\sqrt{n}$. Note que o tempo de processamento requerido pelo AEMT é sublinear com o número de chaves do sistema, limitada pela função $f(n)=k\sqrt{n}$, onde n é o número de chaves e k é uma constante positiva. Com base nesses resultados, é esperado que o algoritmo não incorra em problemas de explosão combinatorial com o aumento da rede. O Sistema 4, com 30.880 barras e 5.166 chaves aproxima-se de um sistema real de uma cidade com cerca de 1,5 milhões de habitantes, indicando que o AEMT pode ser adequado para lidar com uma rede real de larga escala sem simplificações na modelagem da mesma.

Figura 7.21 – Resultados indicando tempo de processamento sublinear para o AEMT.

A Tabela 7.7 mostra a média de resultados obtidos a partir dos Sistemas 1, 2, 3 e 4 para 10.000, 100.000, 400.000, 800.000 e 1.000.000 indivíduos gerados em 30 execuções do algoritmo. Como apenas 1 indivíduo é gerado em cada iteração, quanto maior o tamanho do sistema, maior o número de iterações necessárias para encontrar soluções adequadas.

Tabela 7.7 – Média de Resultados de 30 execuções do AEMT.

Sistema		Número de Iterações				
		10.000	100.000	400.000	800.000	1.000.000
1	Redução de Perdas[%]	14,41	15,88	15,89	15,89	15,89
	Tempo de Processamento [s]	0,34	3,56	14,57	29,18	36,92
2	Redução de Perdas[%]	11,73	15,58	15,66	15,66	15,66
	Tempo de Processamento [s]	0,51	5,95	24,90	50,58	63,98
3	Redução de Perdas[%]	10,59	23,33	24,08	24,16	24,17
	Tempo de Processamento [s]	0,96	11,87	52,46	108,11	137,10
4	Redução de Perdas[%]	6,76	24,03	26,90	27,43	27,52
	Tempo de Processamento [s]	1,22	14,58	67,51	149,87	192,37

Para 10.000 iterações, a porcentagem de redução de perdas diminui com o tamanho do Sistema. Como o número de possíveis configurações aumenta exponencialmente com o número de chaves, para os sistemas maiores existem muitas configurações para serem investigadas, dificultando a busca por soluções adequadas em apenas 10.000 iterações. Para 100.000 iterações, a porcentagem de redução de perdas em geral aumenta com o tamanho da rede. Esse comportamento mantém-se para 400.000, 800.000 e 1.000.000 de iterações. Pode-se observar também uma tendência de convergência da porcentagem de redução de perdas com o aumento do número de iterações, praticamente atingindo o máximo de redução com 1.000.000 de iterações. De uma forma geral, para Sistemas com maior número de chaves, há maior possibilidade de existirem configurações de melhor qualidade.

Porém, encontrar as configurações mais adequadas para redes de grande escala é difícil, pois o espaço de busca das configurações é grande. Em outras palavras, um modelo de rede considerando todas as chaves de um SDR real, proporciona um espaço de busca que pode ter melhores configurações que o proporcionado por modelos simplificados, embora seja difícil encontrar configurações mais adequadas em grandes espaços de busca. Mesmo assim, o AEMT demonstrou ser capaz de encontrar essas configurações, para grandes SDRs.

7.7 Testes com o SDR da Taiwan Power Company

Poucos trabalhos na literatura utilizam SDRs reais e/ou de grande porte nas análises de desempenho. Em geral, somente os trabalhos com SDRs pequenos disponibilizam todos os dados necessários para sua reutilização. Um sistema que tem sido utilizado na literatura em trabalhos de reconfiguração de redes para redução de perdas, é o SDR da TPC, cujos dados estão disponíveis em [94-96, 135].

Esse sistema consiste de uma rede trifásica equilibrada de 11,4kV com 11 alimentadores, 83 chaves NF, 13 chaves NA, com 28,35MW de carga ativa total e 20,70MVAR de carga reativa total. A Figura 7.22 mostra a topologia inicial do sistema, onde os círculos representam uma barra ou centro de carga, linhas cheias são chaves NF, linhas tracejadas são chaves NA e as letras de A a K representam barras em uma subestação. Os dados da rede são mostrados no Apêndice A.

Diversas técnicas foram aplicadas em reconfiguração de redes para redução de perdas de potência no sistema da TPC, das quais se destacam: um algoritmo baseado em *plant growth simulation* (PGSA) [94]; Evolução Diferencial Híbrida Inteiro-mista (EDHI) [95]; Algoritmo de busca baseado em Colônias de Formigas (ACF) e Algoritmos Genéticos [96]; além de *Simulated Annealing* [135].

Figura 7.22 – Topologia Inicial do SDR da Taiwan Power Company.

Especificações do computador utilizado nos testes e da linguagem computacional de implementação dos algoritmos, quando disponíveis no trabalho original, são descritos a seguir:

- PGSA – processador de 2,2GHz e 256MB de memória RAM desenvolvido em Visual C++;
- EDHI – Pentium II, processador de 266MHz, desenvolvido em MATLAB;

- SA, AG e ACF – Pentium IV, processador de 1,3GHz, desenvolvido em MATLAB;
- AEMT – Pentium IV, processador de 2,01GHz, 4GB de memória RAM, desenvolvido em C.

Os trabalhos citados acima utilizaram a mesma configuração inicial (ver Figura 7.22) e encontraram a mesma solução com redução de 11,68% nas perdas de potência, diferenciando no tempo de processamento. A perda total na configuração inicial é de 531,99kW e na configuração final é de 469,88kW. As chaves abertas na topologia das configurações inicial e final correspondem a:

- Inicial – 5-55, 7-60, 11-43, 12-72, 13-76, 14-18, 16-26, 20-83, 28-32, 29-39, 34-46, 40-42 e 53-64;
- Final – 54-55, 6-7, 11-43, 71-72, 12-13, 14-18, 16-26, 82-83, 28-32, 38-39, 33-34, 41-42 e 61-62.

Tabela 7.8 apresenta o tempo de processamento requerido por cada método aplicado para redução de perdas do SDR da TPC. Não é possível a comparação precisa dos métodos somente pelo tempo de processamento, pois computadores com diferentes processadores foram utilizados. No entanto, devido os processadores serem da mesma família (Pentium), é possível fazer uma estimativa do número de operações necessárias na obtenção das soluções em cada método. Assim, a Tabela 7.8 apresenta também, o modelo do processador, quando disponibilizado nas respectivas fontes, a capacidade de operações por segundo (OPS) do processador e o número de operações (NO) necessárias no desenvolvimento da simulação.

Tabela 7.8 – Tempo de processamento dos métodos que utilizam o SDR da TPC.

	Métodos					
	AEMT	PGSA	EDHI	SA	AG	ACF
Tempo de Proc. [s]	0,01	113,25	36,15	257,43	303,66	241,51
Processador	P. IV	N.E. ^(*)	P. II	P. IV	P. IV	P. IV
OPS ^(**)	$2,01 \cdot 10^9$	$2,20 \cdot 10^9$	$0,27 \cdot 10^9$	$1,30 \cdot 10^9$	$1,30 \cdot 10^9$	$1,30 \cdot 10^9$
NO ^(***)	$0,02 \cdot 10^9$	$249,15 \cdot 10^9$	$9,62 \cdot 10^9$	$334,66 \cdot 10^9$	$394,76 \cdot 10^9$	$313,96 \cdot 10^9$

(*) N.E. – Família de processador não especificada. Nesse caso supõe-se um Pentium.

(**) OPS – Número de Operações por Segundo;

(***) NO – Número de Operações.

O tempo médio de processamento relativo ao AEMT foi de 0,0165s para gerar 1.500 indivíduos e de 0,0132s para atingir a solução encontrada na literatura (indivíduo da iteração 1.202) em um Pentium IV de 2,01 GHz, isto é, com a capacidade de realizar $2,01 \times 10^9$ OPS. Assim, o número esperado de operações requeridas pelo AEMT para atingir a solução do indivíduo 1.202 é de $2,01 \times 10^9$ OPS x 0,01s, isto é, $0,02 \times 10^9$. Mesmo que o método de soma de potência para o fluxo de carga tivesse sido utilizado no AEMT, sendo 3,46 vezes mais lento que o de soma de corrente (ver Seção 7.8.1), o número de operações ($0,07 \times 10^9$) teria sido muito menor que o número de operações necessárias pelos outros métodos.

7.8 AEMT com modelos de corrente constante e potência constante para o fluxo de carga

Neste trabalho o Fluxo de carga de varredura direta inversa com o Modelo de Corrente Constante (FMCC) para a soma de corrente foi utilizado. No FMCC, para cada configuração gerada, executam-se apenas duas varreduras: uma varredura para determinar o fluxo de corrente em cada ramo (inversa), e outra para determinar a tensão nas barras do SDR (direta) (ver Seção 5.1). Por outro lado, o Fluxo de carga com o Modelo de Potência Constante (FMPC) para soma de potência, exige um número maior de iterações, dependendo da precisão desejada.

Conseqüentemente, maior esforço computacional é requerido, aumentando-se o tempo de processamento.

Em problemas de reconfiguração de SDRs, a necessidade de execução de um fluxo de carga, para avaliar cada configuração em potencial, é requerida muitas vezes durante o processo de otimização. Assim, na maioria dos algoritmos para essas aplicações, grande parte do tempo de processamento é para execução de fluxo de carga. Particularmente, em problemas de restabelecimento de energia, o objetivo é encontrar topologias do SDR que atendam restrições operacionais do sistema e que restabeleçam a energia ao maior número de consumidores possíveis em um curto intervalo de tempo. Assim, dada a necessidade de trabalhar em tempo real, o FMCC é a princípio mais vantajoso.

Vários trabalhos de reconfiguração de redes não utilizam o FMCC, e sim o FMPC. Por essa razão, resultados da aplicação do AEMT utilizando ambos os modelos para redução de perdas no SDR-SC são comparados a seguir. Os métodos de soma de potência e soma de correntes foram apresentados na Seção 5.1.

7.8.1 Desempenho do AEMT com FMCC e FMPC

Em cada teste, 20.000 indivíduos foram avaliados. A Tabela 7.9 apresenta resultados médios e o desvio padrão para 30 execuções do AEMT utilizando o FMCC e o FMPC para o SDR-SC para redução de perdas sem a ocorrência de faltas. Note que o tempo de processamento com o FMCC é 3,46 vezes menor que o tempo de processamento utilizando o FMPC; enquanto que as perdas totais e o número de chaveamentos de cada modelo são próximos.

A Tabela 7.10 apresenta a menor tensão encontrada em cada um dos 23 alimentadores na configuração inicial, de acordo com o modelo de carga utilizado.

Nota-se que os ângulos encontrados pelos dois modelos de fluxo de carga são iguais para todas as barras analisadas e as amplitudes da tensão, em p.u., muito próximas. Isso significa que, para efeitos de avaliação das restrições de operação do sistema, a vantagem do uso do FMPC são pequenas, com a desvantagem do aumento significativo do tempo de computação.

Tabela 7.9 – Resultados médios de 30 execuções utilizando o FMCC e o FMPC.

Modelo do Fluxo de Carga		Perdas Finais (kW)	Tempo de Processamento (s)	Número de Chaveamentos	Iteração do Melhor Indivíduo
FMCC	Média	237,43	0,79	87	11409
	Desvio Padrão	2,88	0,01	6	4234
FMPC	Média	236,09	2,73	89	10068
	Desvio Padrão	2,74	0,07	5	3469

Tabela 7.10 – Menor tensão no alimentador de acordo com o modelo de carga.

Alimentador	Barra com menor tensão	FMCC		FMPC	
		V (pu)	Graus	V (pu)	Graus
1	937	0,995116	-0,73	0,995128	-0,73
2	621	0,994029	-0,74	0,994036	-0,74
3	814	0,985929	-1,65	0,985952	-1,65
4	884	0,998518	-0,20	0,998520	-0,20
5	903	0,999370	-0,09	0,999371	-0,09
6	1245	0,992436	-0,97	0,992459	-0,97
7	1313	0,995389	-0,25	0,995394	-0,25
8	1396	0,994298	-1,03	0,994409	-1,03
9	1618	0,993790	-0,95	0,993801	-0,95
10	1811	0,991044	-1,04	0,991053	-1,04
11	1867	0,998192	-0,32	0,998194	-0,32
12	1951	0,996401	-0,65	0,996418	-0,65
13	2087	0,996232	-0,54	0,996234	-0,54
14	2408	0,967477	-2,11	0,967571	-2,11
15	2516	0,997469	-0,25	0,997471	-0,25
16	2613	0,998132	-0,27	0,998134	-0,27
17	2670	0,994663	-0,57	0,994669	-0,57
18	2711	0,997539	-0,45	0,997543	-0,45
19	2770	0,997427	-0,37	0,997430	-0,37
20	1623	0,991276	-0,74	0,991285	-0,74
21	3259	0,993568	-0,78	0,993579	-0,78
22	3436	0,987014	-1,00	0,987034	-1,00
23	3777	0,988665	-0,41	0,988674	-0,41

7.9 Comparação do AEMT com outros AEs Multiobjetivos

No Capítulo 6, além do método proposto (AEMT), os métodos NSGA-II [85] e SPEA2 [130] também foram apresentados por serem dois AEs que têm obtido resultados relevantes em diversas aplicações. Esta Seção compara os resultados encontrados pelo AEMT, NSGA-II e SPEA2 para o problema de reconfiguração de redes para redução de perdas utilizando o SDR-SC. O NSGA-II e o SPEA2 originais foram adaptados para utilizar a RNP. Como consequência, esses métodos foram implementados computacionalmente com pequenas modificações em relação aos métodos originais para utilização dos Operadores específicos da RNP. Como consequência, os Operadores de mutação e cruzamento do NSGA-II e SPEA2 foram substituídos pelos Operadores 1 e 2 da RNP (ver Capítulo 4). Para comparar o desempenho dos 3 métodos, as perdas totais no sistema e o número de chaveamentos foram os 2 objetivos considerados para serem minimizados.

Os parâmetros utilizados nos testes em cada método são mostrados a seguir:

NSGA-II –	tamanho da população pai $ P $ igual a 100;
tamanho da população de descendentes $ Q $ igual a 100;
número de iterações igual a 150;
SPEA2 –	tamanho da população $ P $ igual a 100;
tamanho da população externa $ P' $ igual a 100;
número de iterações igual a 150;
AEMT –	tamanho das subpopulações (tabelas) S_{P_i} igual a 5, para $i = 1, \dots, 5$;
número de iterações igual a 15.000.

Deve-se ressaltar que, para o AEMT, apenas 1 indivíduo foi gerado em cada iteração, enquanto que para os métodos NSGA-II e SPEA2 100 indivíduos foram

gerados por iteração. Ao final do processo, todos os métodos geraram 15.000 indivíduos.

Para fins de comparação, a seleção do Operador 1 ou Operador 2 para gerar um novo indivíduo, para os 3 métodos, foi realizada aleatoriamente, isto é, sem a utilização de ajuste dinâmico. A seleção do indivíduo para reprodução, no NSGA-II e SPEA2, é realizada pelo método de torneio; enquanto que para o AEMT, seleciona-se aleatoriamente uma subpopulação e, em seguida, seleciona-se aleatoriamente um indivíduo dessa subpopulação.

Todos os algoritmos partem da rede inicial com as seguintes características:

- Somatória de perdas de potência igual a 281,34kW;
- Maior queda de tensão igual a 3,25 %;
- Maior carregamento na rede igual a 67,12%;
- Maior carregamento nas Subestações igual a 53,34%.

O tempo de processamento foi obtido da média de 30 execuções dos métodos AEMT, NSGA-II e SPEA2 sendo 557ms, 642ms e 649ms respectivamente. As fronteiras de Pareto obtidas pelo método NSGA-II nas iterações 1 e 150 são apresentadas na Figura 7.23. As fronteiras de Pareto geradas a partir do método SPEA2 nas iterações 1 e 150 são apresentadas na Figura 7.24. As soluções encontradas pelo AEMT são apresentadas na Figura 7.25. Porém, para a subpopulação de carregamento das subestações há indivíduos com perdas de potência muito elevadas (acima de 3.660kW), assim, esses indivíduos não aparecem nessa Figura. A fronteira de Pareto é apresentada na Figura 7.26 para cada método em análise.

Figura 7.23 – Fronteiras de Pareto encontradas pelo NSGA-II nas iterações 1 e 150.

Figura 7.24 – Fronteiras de Pareto encontradas pelo SPEA2 nas iterações 1 e 150.

Figura 7.25 – Soluções encontradas pelo AEMT.

Figura 7.26 – Comparação das melhores fronteiras obtidas pelo AEMT, NSGA-II e SPEA2.

As soluções, em azul, apresentadas na parte superior da Figura 7.26 são relativas às melhores soluções na subpopulação de perdas de potência do AEMT, enquanto as outras soluções (em azul) são as melhores soluções finais encontradas nas demais subpopulações. Pela Figura 7.26 nota-se que o AEMT é capaz de encontrar melhores soluções, principalmente nos extremos da fronteira. No entanto, a diversidade de soluções é baixa.

A vantagem do NSGA-II e SPEA2 está na maior diversidade das soluções encontradas, pois conseguem mapear uma grande parte da fronteira de Pareto com soluções factíveis, variando desde a primeira configuração (com 0 manobras) até configurações com 40 manobras de chaves (NSGA-II) e 54 manobras de chaves (SPEA2). As soluções da fronteira encontrada pelo SPEA2 dominam, em grande parte, as soluções obtidas pelo NSGA-II.

Deve-se observar que todas as soluções apresentadas na Figura 7.26 atendem as restrições operacionais do sistema, bem como as restrições de radialidade e conectividade que são atendidas naturalmente quando os Operadores da RNP são utilizados.

8 Conclusões

Este Capítulo sintetiza os principais resultados obtidos e apresenta algumas considerações sobre perspectivas de pesquisas futuras relativas a este trabalho. O restabelecimento de energia em SDRs geralmente envolve reconfiguração de redes. Um AE associado a uma estrutura de dados eficiente foi proposto, o AEMT. Existem várias aplicações para o AEMT, sendo algumas citadas como sugestões de trabalhos futuros. Este trabalho explorou com maior ênfase essa técnica para o problema de restabelecimento de energia em um SDR real, no qual se busca o restabelecimento de energia ao maior número possível de consumidores após as faltas terem sido identificadas e isoladas, sem violar os limites operacionais do sistema como sobrecarga nas linhas e subestações e queda de tensão estabelecidos em normas técnicas.

Sistemas de Distribuição do mundo real possuem grande quantidade de barras e chaves. Assim, técnicas que consigam lidar com modelagens complexas envolvendo a maior quantidade possível de componentes de um SDR podem ser interessantes para obtenção de planos de restabelecimento de melhor qualidade. O uso da estrutura de dados RNP é base de desenvolvimento do AEMT, uma vez que essa codificação garante a rápida geração de novas configurações de rede sempre radiais e conexas. Com isso, evitam-se rotinas para identificar e corrigir essas infactibilidades, que aumentam consideravelmente o tempo de processamento e estão presentes em várias abordagens encontradas na literatura.

O AEMT foi avaliado utilizando uma rede de um SDR real com 3.860 barras e 632 chaves (509 chaves NF e 123 chaves NA) sem simplificações em relação à rede real, ou seja, foram consideradas todas as linhas, barras e chaves da mesma. Além disso, foram realizados testes utilizando outras três redes sintetizadas a partir desse SDR real. As redes sintetizadas possuem tamanhos (número de barras e chaves) que são aproximadamente duas, quatro e oito vezes o tamanho do SDR real. Com isso, o AEMT foi avaliado inclusive para uma rede com 30.880 barras, 4.264 setores, 5.166 chaves (4.072 chaves NF e 1.094 chaves NA), 24 subestações e 184 alimentadores. Considerando que o SDR real alimentava uma cidade de quase 200.000 habitantes, a maior rede sintetizada corresponderia a uma SDR para uma cidade de cerca de 1,5 milhões de habitantes.

Quanto maior o tamanho da rede, maior o número de configurações possíveis. Em consequência, aumenta-se a possibilidade de existirem um número maior de soluções que atendam as restrições do problema. Por outro lado, obter resultados adequados para esse tipo de problema é difícil, pois o número de configurações infactíveis também aumenta com o tamanho da rede. Outros aspectos como funções não-lineares, descontínuas, com vários ótimos locais e envolvendo múltiplos objetivos conflitantes tornam o problema ainda mais complexo.

Os resultados obtidos nos testes com o AEMT para o SDR-SC e para as redes sintetizadas mostram que o AEMT é capaz de vencer tais dificuldades e explorar adequadamente o espaço das soluções factíveis, encontrando configurações de alta qualidade em curto intervalo de tempo. Em média, o AEMT requer tempo de processamento de 2,20s para gerar 30.000 configurações.

Para lidar adequadamente com múltiplos objetivos conflitantes, o AEMT utiliza de subpopulações organizadas em uma tabela, a qual corresponde a uma

estratégia simples e de baixo custo computacional para evitar ótimos locais e encontrar soluções na fronteira de Pareto.

Resultados do AEMT para redução de perdas de potência foram comparados com resultados encontrados pelo NSGA-II e SPEA2 adaptados com a RNP para o mesmo problema (ver Seção 7.9). Estes dois algoritmos encontram fronteiras de Pareto com um número maior de soluções. Por outro lado, o AEMT conseguiu estender significativamente os extremos da fronteira. Vale ressaltar também que o tempo de computação requerido pelo AEMT é menor que das outras duas técnicas multiobjetivo, mas não de forma expressiva.

As principais contribuições deste trabalho podem ser sintetizadas como segue:

1. Desenvolvimento de um AE Multi-objetivo em Tabela – o AEMT, capaz de lidar com problemas com objetivos conflitantes de uma forma simples (ver Seção 6.4);
2. Utilização de uma estrutura de dados, a RNP, que possibilita investigar em reduzido tempo de processamento uma grande quantidade de configurações factíveis (ver Capítulo 4);
3. Proposição de um fluxo de carga não convencional, baseado na RNP, com modelo de corrente constante utilizando o método de varredura direta/inversa para soma de corrente, possibilitando uma rápida avaliações de redes com milhares de barras (ver Seção 5.1);
4. Desenvolvimento de uma metodologia eficiente para calcular o número de chaveamentos de uma configuração para outra (ver Seção 5.2);

5. Apresentação de resultados que garantem que o AEMT encontra soluções adequadas e que pode encontrar soluções em regiões do espaço de busca que o NSGA-II e o SPEA2, nos testes realizados, não conseguiram atingir (ver Seção 7.9);
6. Aplicação do AEMT ao SDR da *Taiwan Power Company*, SDR disponível na literatura encontrando os mesmos resultados que outros métodos para redução de perdas, porém com tempo de processamento muito menor (ver Seção 7.7).

Como sugestões de trabalhos futuros destacam-se:

1. Verificar o comportamento do AEMT para outros SDRs reais de larga-escala;
2. Trabalhar com um conjunto de funções agregação na Tabela de subpopulações do AEMT, aumentando assim, o número de subpopulações e a diversidade de soluções na fronteira de Pareto obtida pelo AEMT;
3. Explorar o AEMT para outros problemas de reconfiguração de redes: redução de perdas de potência, alocação ótima de capacitores, planejamento da expansão do sistema e balanceamento de cargas;
4. Adaptar o AEMT para trabalhar com a RNPG (Representação Nó-Profundidade Grau), uma nova codificação em desenvolvimento no grupo de pesquisa de Sistemas Embarcados, Evolutivos e Robóticos do ICMC-USP [136] que estende a RNP e possui garantia teórica de complexidade de tempo sublinear.

9 Publicações Originadas desta Pesquisa

1. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "Redução de Perdas em Sistemas de Distribuição de Tamanho Real", *XVI CBA - Congresso Brasileiro de Automática*, Salvador-BA, pp. 1692-1697, Out., 2006.
2. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "Representação Nó-profundidade para AEs aplicados a minimização de perdas resistivas em Sistemas de Distribuição", *VIII SBAI - Simpósio Brasileiro de Automação Inteligente*, Florianópolis-SC, Out., 2007.
3. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "Reconfiguração de Sistemas de Distribuição de Energia Elétrica utilizando Algoritmo Evolutivo com Representação Nó-Profundidade", *Cong. Latino Americano de Ger. e Transm. de Energia Elétrica - CLAGTEE*, Chile, Out., 2007.
4. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "Energy Restoration for Large-Scale Distribution System using EA and a New Data Structure", *2008 IEEE Power & Energy Society General Meeting*, Pittsburgh-PA, USA, july, 2008.
5. **A. C. Santos**, M. Nanni, M. R. Mansour, A. C. B. Delbem, J. B. A. L. Jr., N. G. Bretas, "A Power Flow Method Computationally Efficient for Large-Scale Distribution Systems", *2008 IEEE PES Latin America Transmission and Distribution Conference and Exposition*, Bogota-Colombia, Aug., 2008.

6. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "A Multiobjective Evolutionary Algorithm with Node-depth Encoding for Energy Restoration", *The 4th International Conference on Natural Computation - ICNC'08*, Jinan-China, Oct., 2008.
7. **A. C. Santos**, A. C. B. Delbem, N. G. Bretas, "Restabelecimento de Energia em grandes Redes de Distribuição com uma Estrutura de Dados eficiente para manipulação de grafos", *XVII CBA - Congresso Brasileiro de Automática*, set., 2008.
8. M. R. Mansour, **A. C. Santos**, J. B. A. London Jr., A. C. B. Delbem, N. G. Bretas, "Aplicação de Conjuntos Fuzzy para definição do critério de parada de um Algoritmo Evolutivo desenvolvido para reconfiguração de Redes de Distribuição", *Congresso da Academia Trinacional das Ciências – C3N*, Out., 2008.
9. M. R. Mansour, **A. C. Santos**, A. C. B. Delbem, J. B. A. L. Jr., N. G. Bretas, "Aplicação de conjuntos fuzzy para avaliação das funções multi-objetivo de um Algoritmo Evolutivo para a reconfiguração de sistemas de distribuição de energia elétrica", *In: XL SBPO: Simpósio Brasileiro de Pesquisa Operacional*, João Pessoa, PB, 2008.
10. M. R. Mansour, **A. C. Santos**, A. C. B. Delbem, J. B. A. L. Jr., N. G. Bretas, "Energy Restoration in Distribution Systems using Multi-Objective Evolutionary Algorithm and an Efficient Data Structure". *In: Proceedings of the IEEE Power Tech*, Bucharest – Romania, 2009. (Aceito para publicação).
11. **A. C. Santos**, A. C. B. Delbem, J. B. A. London Jr., N. G. Bretas, "Node-depth Encoding and Evolutionary Algorithm for Restoration of Large-scale Distribution Systems", *IEEE Transactions on Power Systems*. (Submetido).

10 Referências Bibliográficas

- [1] S. H. Strogatz, "Exploring complex networks", *Nature*, vol. 410, pp. 268-276, Março, 2001.
- [2] W. X. Fan, C. Guanrong, "Complex networks: small-world, scale-free and beyond", *IEEE Circuits and Systems Magazine*, vol. 3, pp. 6-20, 2003.
- [3] S. N. Dorogovtsev, A. V. Goltsev, J. F. F. Mendes, "Critical phenomena in complex networks", *Reviews of Modern Physics*, vol. 80, pp. 1275, Outubro, 2008.
- [4] J. A. Cipoli, *Engenharia de Distribuição*, 1ª ed. Rio de Janeiro: Qualitymark Editora Ltda, 1993.
- [5] N. Kagan, C. C. B. Oliveira, E. J. Robba, *Introdução aos sistemas de distribuição de energia elétrica*, vol. 1, 1ª ed. São Paulo: Edgard Blucher, 2005.
- [6] R. J. Sarfi, M. M. A. Salama, A. Y. Chikhani, "Distribution system reconfiguration for loss reduction: an algorithm based on network partitioning theory", *IEEE Power Industry Computer Application Conference*, pp. 503-509, Maio, 1995.
- [7] F. V. Gomes, S. Carneiro, Jr., J. L. R. Pereira, M. P. A. Vinagre, P. A. N. A. Garcia, L. R. A. De Araujo, "A new distribution system reconfiguration approach using optimum power flow and sensitivity analysis for loss reduction", *IEEE Transactions on Power Systems*, vol. 21, pp. 1616-1623, Novembro, 2006.
- [8] C. S. Chen, M. S. Kang, J. C. Hwang, C. W. A. Huang, "Application of binary integer programming for load transfer of distribution systems", *Proceedings:*

International Conference on Power System Technology, PowerCon 2000, pp. 305-310 vol.1, Dezembro, 2000.

- [9] M. W. Siti, D. V. Nicolae, A. A. Jimoh, A. A. Ukil, "Reconfiguration and load balancing in the LV and MV distribution networks for optimal performance", *IEEE Transactions on Power Delivery*, vol. 22, pp. 2534-2540, Outubro, 2007.
- [10] J. R. S. Mantovani, F. Casari, R. A. Romero, "Reconfiguração de sistemas de distribuição radiais utilizando o critério de queda de tensão", *SBA Controle & Automação*, vol. 11, pp. 150-159, 2000.
- [11] P. V. Prasad, S. Sivanagaraju, N. Sreenivasulu, "Network reconfiguration for load balancing in radial distribution systems using genetic algorithm", *Electric Power Components and Systems*, vol. 36, pp. 63-72, Janeiro, 2008.
- [12] J. Inagaki, J. Nakajima, M. Haseyama, "A multiobjective service restoration method for power distribution systems", *Proceedings IEEE International Symposium on Circuits and Systems, ISCAS*, pp. 1784-1787, Maio, 2006.
- [13] Y. Kumar, B. Das, J. Sharma, "Multiobjective, multiconstraint service restoration of electric power distribution system with priority customers", *IEEE Transactions on Power Delivery*, vol. 23, pp. 261-270, Janeiro, 2008.
- [14] M.-S. Tsai, "Development of an object-oriented service restoration expert system with load variations", *IEEE Transactions on Power Systems*, vol. 23, pp. 219-225, Fevereiro, 2008.
- [15] N. Kagan, "Configuração de redes de distribuição através de algoritmos genéticos e tomada de decisão fuzzy", Tese de Livre Docência, USP, São Paulo, 1999.
- [16] D. I. Sun, D. R. Farris, P. J. Cote, R. R. Shoults, M. S. Chen, "Optimal distribution substation and primary feeder planning via the fixed charge network

- formulation", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-101, pp. 602-609, Março, 1982.
- [17] M. Vaziri, K. Tomsovic, A. Bose, "Numerical analysis of a directed graph formulation of the multistage distribution expansion problem", *IEEE Transactions on Power Delivery*, vol. 19, pp. 1348-1354, Julho, 2004a.
- [18] M. Vaziri, K. Tomsovic, A. Bose, "A directed graph formulation of the multistage distribution expansion problem", *IEEE Transactions on Power Delivery*, vol. 19, pp. 1335-1341, Julho, 2004b.
- [19] P. C. Paiva, H. M. Khodr, J. A. Dominguez-Navarro, J. M. A. Yusta, A. J. A. Urdaneta, "Integral planning of primary-secondary distribution systems using mixed integer linear programming", *IEEE Transactions on Power Systems*, vol. 20, pp. 1134-1143, Maio, 2005.
- [20] W. El-Khattam, Y. G. Hegazy, M. M. A. Salama, "An integrated distributed generation optimization model for distribution system planning", *IEEE Transactions on Power Systems*, vol. 20, pp. 1158-1165, Maio, 2005.
- [21] N. G. Boulaxis, M. P. Papadopoulos, "Optimal feeder routing in distribution system planning using dynamic programming technique and GIS facilities", *IEEE Transactions on Power Delivery*, vol. 17, pp. 242-247, Janeiro, 2002.
- [22] E. Diaz-Dorado, J. C. Pidre, "Optimal planning of unbalanced networks using dynamic programming optimization", *IEEE Transactions on Power Systems*, vol. 19, pp. 2077-2085, Novembro, 2004.
- [23] M. A. Farrag, M. M. El-Metwally, M. S. El-Bages, "A new model for distribution system planning", *International Journal of Electrical Power & Energy Systems*, vol. 21, pp. 523-531, Julho, 1999.

- [24] B. Mauro, S. Thomas, P. Luis, V. Klaus, "A racing algorithm for configuring metaheuristics", *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 11-18, Julho, 2002.
- [25] K. Aoki, K. Nara, T. Satoh, M. Kitagawa, K. Yamanaka, "New approximate optimization method for distribution system planning", *IEEE Transactions on Power Systems*, vol. 5, pp. 126-132, Fevereiro, 1990.
- [26] E. Miguez, J. Cidras, E. Diaz-Dorado, J. L. Garcia-Dornelas, "An improved branch-exchange algorithm for large-scale distribution network planning", *IEEE Transactions on Power Systems*, vol. 17, pp. 931-936, Novembro, 2002.
- [27] V. Parada, J. A. Ferland, M. Arias, K. Daniels, "Optimization of electrical distribution feeders using simulated annealing", *IEEE Transactions on Power Delivery*, vol. 19, pp. 1135-1141, Fevereiro, 2004.
- [28] K. Nara, Y. Hayashi, Y. Yamafuji, H. Tanaka, J. Hagihara, S. Muto, S. Takaoka, M. Sakuraoka, "A Tabu Search algorithm for determining distribution tie lines", *Proceedings, ISAP '96, International Conference on Intelligent Systems Applications to Power Systems*, pp. 266-270, Janeiro, 1996.
- [29] A. Augugliaro, L. Dusonchet, E. R. Sanseverino, "An evolutionary parallel tabu search approach for distribution systems reinforcement planning", *Advanced Engineering Informatics*, vol. 16, pp. 205-215, Novembro, 2002.
- [30] R. Ranjan, B. Venkatesh, D. Das, "A new algorithm for power distribution system planning", *Electric Power Systems Research*, vol. 62, pp. 55-65, Janeiro, 2002.
- [31] I. J. Ramirez-Rosado, J. L. Bernal-Agustin, "Reliability and costs optimization for distribution networks expansion using an evolutionary algorithm", *IEEE Power Engineering Review*, vol. 21, pp. 70-70, Abril, 2001.

- [32] A. C. B. Delbem, A. C. P. L. F. Carvalho, N. G. Bretas, "Main chain representation for evolutionary algorithms applied to distribution system reconfiguration", *IEEE Transactions on Power Systems*, vol. 20, pp. 425-436, Fevereiro, 2005.
- [33] F. Mendoza, A. Garcia, J. L. Bernal-Agustin, "Application of the NPGA to the design of power distribution systems", *IEEE/PES Transmission & Distribution Conference and Exposition: Latin America*, pp. 1-5, Agosto, 2006.
- [34] F. Mendoza, J. L. Bernal-Agustin, J. A. Dominguez-Navarro, "NSGA and SPEA applied to multiobjective design of power distribution systems", *IEEE Transactions on Power Systems*, vol. 21, pp. 1938-1945, Novembro, 2006.
- [35] E. M. Carreno, N. Moreira, R. Romero, "Distribution network reconfiguration using an efficient evolutionary algorithm", *IEEE Power Engineering Society General Meeting*, pp. 1-6, Junho, 2007.
- [36] J.-S. Wu, T. E. Lee, Y.-J. Lin, "A rule-based genetic algorithm for the inter-feeder load transfer in the multiple outages of electrical distribution systems", *Second International Conference on Innovative Computing, Information and Control*, pp. 444-444, Setembro, 2007.
- [37] J. F. Gomez, H. M. Khodr, P. M. De Oliveira, L. Ocque, J. M. Yusta, R. Villasana, A. J. Urdaneta, "Ant colony system algorithm for the planning of primary distribution circuits", *IEEE Transactions on Power Systems*, vol. 19, pp. 996-1004, Maio, 2004.
- [38] J. Chia-Feng, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design", *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, pp. 997-1006, Abril, 2004.

- [39] J. Xiaoling, Z. Jianguo, S. Ying, L. Kejun, Z. Boqin, "Distribution network reconfiguration for load balancing using binary particle swarm optimization", *PowerCon 2004. International Conference on Power System Technology*, pp. 507-510, Novembro, 2004.
- [40] P. Hansen, N. Mladenovic, "Variable neighborhood search: principles and applications", *European Journal of Operational Research*, vol. 130, pp. 449-467, Maio, 2001.
- [41] N. Mladenovic, P. Hansen, "Variable neighborhood search", *Computers & Operations Research*, vol. 24, pp. 1097-1100, Novembro, 1997.
- [42] W. G. Zvietcovich, R. A. Romero, "Reconfiguração de sistemas de distribuição de energia elétrica utilizando a metaheurística busca em vizinhança variável", *XVI CBA - Congresso Brasileiro de Automática*, Salvador-BA, pp. 1411-1416, Outubro, 2006.
- [43] P. Festa, M. G. C. Resende, "GRASP: An annotated bibliography", *In Essays and Surveys on Metaheuristics, C. Ribeiro and P. Hansen, Ed. Essays and Surveys on Metaheuristics, Kluwer Academic Publishers*, pp. 325-367, 2002.
- [44] F. Glover, J. P. Kelly, M. Laguna, "New advances for wedding optimization and simulation", *Simulation Conference Proceedings*, pp. 255-260, 1999.
- [45] R. Martí, M. Laguna, F. Glover, "Principles of scatter search", *European Journal of Operational Research*, vol. 169, pp. 359-372, Março, 2006.
- [46] A. N. B. Peter, T. Dirk, "Adaptive variance scaling in continuous multi-objective estimation-of-distribution algorithms," in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. London, England: ACM, 2007.

- [47] T. S. P. Duque, K. Sastry, A. C. B. Delbem, D. E. Goldberg, "Evolutionary algorithm for large scale problems", *ISDA 2007 - Seventh International Conference on Intelligent Systems Design and Applications*, pp. 819-822, Outubro, 2007.
- [48] K. Deb, *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley, 2001.
- [49] H. A. Abbass, R. Sarker, C. Newton, "PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems", *Proceedings of the 2001 Congress on Evolutionary Computation* pp. 971-978, Maio, 2001.
- [50] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182-197, Abril, 2002.
- [51] A. C. B. Delbem, A. C. P. L. F. Carvalho, C. A. Policastro, A. K. O. Pinto, K. Honda, A. C. Garcia, "Node-depth encoding for evolutionary algorithms applied to network design", *6th Genetic and Evolutionary Computation Conference – GECCO*, pp. 678-687, 2004.
- [52] A. C. Santos, A. C. B. Delbem, "Restabelecimento de energia em grandes redes de distribuição com uma estrutura de dados eficiente para manipulação de grafos", *XVII CBA - Congresso Brasileiro de Automática*, Setembro, 2008.
- [53] A. C. Santos, A. C. B. Delbem, N. G. Bretas, "Representação nó-profundidade para algoritmos evolutivos aplicados a minimização de perdas resistivas em sistemas de distribuição", *VIII SBAI - Simpósio Brasileiro de Automação Inteligente*, Florianópolis-SC, Outubro, 2007.

- [54] A. C. Santos, A. C. B. Delbem, N. G. Bretas, "Energy restoration for large-scale distribution system using EA and a new data structure", *2008 IEEE Power & Energy Society General Meeting*, Pittsburgh-PA, USA, Julho, 2008.
- [55] S. Toune, H. Fudo, T. Genji, Y. Fukuyama, Y. Nakanishi, "Comparative study of modern heuristic algorithms to service restoration in distribution systems", *IEEE Transactions on Power Delivery*, vol. 17, pp. 173-181, Janeiro, 2002.
- [56] S. Curcic, C. S. Özveren, L. Crowe, P. K. L. Lo, "Electric power distribution network restoration: a survey of papers and a review of the restoration problem", *Electric Power Systems Research*, vol. 35, pp. 73-86, Novembro, 1996.
- [57] C. Y. Teo, "A computer aided system to automate the restoration of electrical power supply", *Electric Power Systems Research*, vol. 24, pp. 119-125, Agosto, 1992.
- [58] K. Hoyong, K. Yunseok, J. Kyung-Hee, "Algorithm of transferring the load of the faulted substation transformer using the best-first search method", *IEEE Transactions on Power Delivery*, vol. 7, pp. 1434-1442, Julho, 1992.
- [59] C. C. Liu, S. J. Lee, S. S. Venkata, "An expert system operational aid for restoration and loss reduction of distribution systems", *IEEE Transactions on Power Systems*, vol. 3, pp. 619-626, Maio, 1988.
- [60] Y. Fujii, A. Miura, J. Tsukamoto, M. G. Youssef, Y. Noguchi, "On-line expert system for power distribution system control", *International Journal of Electrical Power & Energy Systems*, vol. 14, pp. 45-53, Fevereiro, 1992.
- [61] T. J. Kendrew, J. F. Graham, "Applying expert system technology to automated distribution feeder deployment and sectionalizing", *Proc. American Power Conj.*, United States, pp. 563-568, Janeiro, 1989.

- [62] K. Okuda, H. Watanabe, F. Wang, K. Yamazaki, T. Baba, "An application of knowledge engineering for fault restoration operation in secondary power systems", *Electr. Eng. Jpn*, vol. 108, pp. 51-59, 1988.
- [63] D. Srinivasan, A. C. Liew, C. S. Chang, J. S. P. Chen, "Intelligent operation of distribution network", *IEE Proceedings-Generation, Transmission and Distribution*, vol. 141, pp. 106-116, Março, 1994.
- [64] D. Shirmohammadi, "Service restoration in distribution networks via network reconfiguration", *IEEE Transactions on Power Delivery*, vol. 7, pp. 952-958, Abril, 1992.
- [65] J. S. Wu, K. L. Tomsovic, C. S. Chen, "A heuristic search approach to feeder switching operations for overload, faults, unbalanced flow and maintenance", *IEEE Transactions on Power Delivery*, vol. 6, pp. 1579-1586, Outubro, 1991.
- [66] Y.-Y. Hsu, H. M. Huang, H. C. Kuo, S. K. Peng, C. W. Chang, K. J. Chang, H. S. Yu, C. E. Chow, R. T. Kuo, "Distribution system service restoration using a heuristic search approach", *IEEE Transactions on Power Delivery*, vol. 7, pp. 734-740, Abril, 1992.
- [67] S. Devi, D. P. Sen Gupta, S. Sargunraj, "A search technique for restoring power supply in complex distribution systems," in *power system for the year 2000 and beyond. Proc. 6th Nat. Power Syst. Conf.*: New Delhi: Tata McGraw-Hill, 1990, pp. 122-125.
- [68] S. Devi, D. P. Sen Gupta, S. Sargunraj, "Optimal restoration of supply following a fault on large distribution systems", *APSCOM-91. International Conference on Advances in Power System Control, Operation and Management*, pp. 508-513 vol.2, Novembro, 1991.

- [69] A. L. Morelato, A. J. Monticelli, "Heuristic search approach to distribution system restoration", *IEEE Transactions on Power Delivery*, vol. 4, pp. 2235-2241, Outubro, 1989.
- [70] J. Nahman, G. Strbac, "A new algorithm for service restoration in large-scale urban distribution systems", *Electric Power Systems Research*, vol. 29, pp. 181-192, Maio, 1994.
- [71] K. Aoki, K. Nara, M. Itoh, T. Satoh, H. Kuwabara, "A new algorithm for service restoration in distribution systems", *IEEE Transactions on Power Delivery*, vol. 4, pp. 1832-1839, Julho, 1989.
- [72] K. Aoki, K. Nara, T. Satoh, "New configuration algorithm for distribution system - priority constrained emergency service restoration", *Proceedings of IFAC Conference on Power Systems and Power Plant Control*, pp. 443-448, 1989.
- [73] E. N. Dialynas, D. G. Michos, "Interactive modeling of supply restoration procedures in distribution system operation", *IEEE Transactions on Power Delivery*, vol. 4, pp. 1847-1854, Julho, 1989.
- [74] N. D. R. Sarma, V. C. Prasad, K. S. Prakasa Rao, "Network reconfiguration in distribution networks for service restoration," in *Power Systems for the Year 2000 and Beyond, Proc. 6th Nat. Power Systems Conj., Bombay, India*, N. D. Tata McGraw-Hill, Ed., 1990, pp. 131-135.
- [75] C. S. Chen, J. S. Wu, C. S. Moo, "Fault restoration by optimizing switch configuration in distribution systems", *J. Chin. Inst. Eng.*, vol. 12, pp. 781-789, 1989.
- [76] Y. Fukuyama, H. D. Chiang, K. N. Miu, "Parallel genetic algorithm for service restoration in electric power distribution systems", *International Journal of Electrical Power & Energy Systems*, vol. 18, pp. 111-119, Fevereiro, 1996.

- [77] M. S. Srinivas, "Distribution load flows: a brief review", *IEEE Power Engineering Society Winter Meeting*, pp. 942-945, Janeiro, 2000.
- [78] A. C. Santos, M. Nanni, M. R. Mansour, A. C. B. Delbem, J. B. A. L. Jr., N. G. Bretas, "A power flow method computationally efficient for large-scale distribution systems", *2008 IEEE PES Latin America Transmission and Distribution Conference and Exposition*, Bogota-Colombia, Agosto, 2008.
- [79] D. E. Goldberg, *Genetic algorithms in search, optimization, and machine learning*. Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 1989.
- [80] A. Augugliaro, L. Dusonchet, E. Riva Sanseverino, "Multiobjective service restoration in distribution networks using an evolutionary approach and fuzzy sets", *International Journal of Electrical Power & Energy Systems*, vol. 22, pp. 103-110, Fevereiro, 2000.
- [81] T. Bäck, D. B. Fogel, Z. Michalewicz, *Handbook of evolutionary computation*: New York: Oxford Univ. Press and Institute of Physics, 1997.
- [82] Y. T. Hsiao, C. Y. Chien, "Enhancement of restoration service in distribution systems using a combination fuzzy-GA method", *IEEE Transactions on Power Systems*, vol. 15, pp. 1394-1400, Novembro, 2000.
- [83] D.-J. Shin, J.-O. Kim, T.-K. Kim, J.-B. Choo, C. Singh, "Optimal service restoration and reconfiguration of network using genetic-tabu algorithm", *Electric Power Systems Research*, vol. 71, pp. 145-152, Outubro, 2004.
- [84] R. Benayoun, J. de Montgolfier, J. Tergny, O. Laritchev, "Linear programming with multiple objective functions: step method (stem)", *Mathematical Programming*, vol. 1, pp. 366-375, Dezembro, 1971.

- [85] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II", *KanGAL report 200001, Indian Institute of Technology, Kanpur, India*, 2000.
- [86] T. H. Cormem, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to algorithms*. Cambridge, MA: MIT Press/McGraw Hill, 1990.
- [87] M. T. Jensen, "Reducing the run-time complexity of multiobjective EAs: The NSGA-II and other algorithms", *IEEE Transactions on Evolutionary Computation*, vol. 7, pp. 503-515, Outubro, 2003.
- [88] D. Shirmohammadi, H. W. Hong, A. Semlyen, G. X. A. Luo, "A compensation-based power flow method for weakly meshed distribution and transmission networks", *IEEE Transactions on Power Systems*, vol. 3, pp. 753-762, Maio, 1988.
- [89] B. Grady, *Object-oriented analysis and design with applications (2nd ed.)*: Benjamin-Cummings Publishing Co., Inc., 1994.
- [90] A. Merlin, H. Back, "Search for a minimal-loss operating spanning tree configuration in an urban power distribution system", *Proc. 5th Power System Computation Conference*, vol. 1, pp. 1-18, 1975.
- [91] J. Z. Zhu, "Optimal reconfiguration of electrical distribution network using the refined genetic algorithm", *Electric Power Systems Research*, vol. 62, pp. 37-42, Maio, 2002.
- [92] K. Nara, A. Shiose, M. Kitagawa, T. Ishihara, "Implementation of genetic algorithm for distribution systems loss minimum re-configuration", *IEEE Transactions on Power Systems*, vol. 7, pp. 1044-1051, Agosto, 1992.
- [93] H. Demuth, M. Beale, "Neural network toolbox user's guide version 3.0", *MathWorks, Inc.*, 1997.

- [94] C. Wang, H. Z. Cheng, "Optimization of network configuration in large distribution systems using plant growth simulation algorithm", *IEEE Transactions on Power Systems*, vol. 23, pp. 119-126, Fevereiro, 2008.
- [95] C. T. Su, C. S. Lee, "Network reconfiguration of distribution systems using improved mixed-integer hybrid differential evolution", *IEEE Transactions on Power Delivery*, vol. 18, pp. 1022-1027, Julho, 2003.
- [96] C.-T. Su, C.-F. Chang, J.-P. Chiou, "Distribution network reconfiguration for loss reduction by ant colony search algorithm", *Electric Power Systems Research*, vol. 75, pp. 190-199, Agosto, 2005.
- [97] B. Enacheanu, B. Raison, R. Caire, O. Devaux, W. Bienia, N. HadjSaid, "Radial network reconfiguration using genetic algorithm based on the matroid theory", *IEEE Transactions on Power Systems*, vol. 23, pp. 186-195, Fevereiro, 2008.
- [98] A. Schrijver, *Combinatorial optimization - polyhedra and efficiency*. Berlin, Germany: Springer-Verlag, 2003.
- [99] S. Civanlar, J. J. Grainger, H. Yin, S. S. H. Lee, "Distribution feeder reconfiguration for loss reduction", *IEEE Transactions on Power Delivery*, vol. 3, pp. 1217-1223, Julho, 1988.
- [100] M. E. Baran, F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing", *IEEE Transactions on Power Delivery*, vol. 4, pp. 1401-1407, Abril, 1989.
- [101] D. Das, "A fuzzy multiobjective approach for network reconfiguration of distribution systems", *IEEE Transactions on Power Delivery*, vol. 21, pp. 202-209, Janeiro, 2006.

- [102] D. Zhang, Z. Fu, L. Zhang, "Joint optimization for power loss reduction in distribution systems", *IEEE Transactions on Power Systems*, vol. 23, pp. 161-169, Fevereiro, 2008.
- [103] M. Srinivas, L. M. Patnaik, "Adaptive probabilities of crossover and mutation in genetic algorithms", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, pp. 656-667, Abril, 1994.
- [104] V. N. Gohokar, M. K. Khedkar, G. M. Dhole, "Formulation of distribution reconfiguration problem using network topology: a generalized approach", *Electric Power Systems Research*, vol. 69, pp. 304-310, Maio, 2004.
- [105] C. L. Lucchesi, *Introdução à teoria dos grafos*, vol. Instituto de Matemática Pura e Aplicada. 12º Colóquio Brasileiro de Matemática. Poços de Caldas, 1979.
- [106] I. S. Gradshteyn, I. M. Ryzhik, *Tables of integrals, series, and products*. San Diego, CA: Academic Press, 6th ed, 2000.
- [107] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, *Network flows: theory, algorithms, and applications*. Englewood Cliffs: Prentice Hall, 1993.
- [108] E. G. Goodaire, M. M. Parmenter, *Discrete mathematics with graph theory*. Englewood Cliffs, NJ: Prentice-Hall, 1998.
- [109] W. F. Tinney, C. E. Hart, "Power flow solution by newton's method", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-86, pp. 1449-1460, Novembro, 1967.
- [110] B. Stott, O. Alsac, "Fast decoupled load flow", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-93, pp. 859-869, Maio, 1974.
- [111] S. C. Tripathy, G. D. Prasad, O. P. Malik, G. S. A. Hope, "Load-flow solutions for ill-conditioned power systems by a newton-like method", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-101, pp. 3648-3657, Outubro, 1982.

- [112] S. Iwamoto, Y. Tamura, "A load flow calculation method for ill-conditioned power systems", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-100, pp. 1736-1743, Abril, 1981.
- [113] D. Rajicic, A. Bose, "A modification to the fast decoupled power flow for networks with high R/X ratios", *IEEE Transactions on Power Systems*, vol. 3, pp. 743-746, Maio, 1988.
- [114] D. Das, H. S. Nagi, D. P. Kothari, "Novel method for solving radial distribution networks", *IEE Proceedings-Generation, Transmission and Distribution*, vol. 141, pp. 291-298, Julho, 1994.
- [115] M. H. Haque, "Efficient load flow method for distribution systems with radial or mesh configuration", *IEE Proceedings-Generation, Transmission and Distribution*, vol. 143, pp. 33-38, Janeiro, 1996.
- [116] A. Chipperfield, P. Fleming, "Evolutionary algorithms for control engineering", *13th IFAC World Congress*, San Francisco, CA, USA, pp. 181-186, 1996.
- [117] D. B. Fogel, *Evolutionary computation: toward a new philosophy of machine intelligence*: IEEE Press Piscataway, NJ, USA, 1995.
- [118] C. R. Darwin, *On the origin of species by means of natural selection of the preservation of favoured races in the struggle for life [1859]*. New York: reprinted by Random House, 1993.
- [119] C. A. C. Coello, G. B. Lamont, D. A. V. Veldhuizen, *Evolutionary algorithms for solving multi-objective problems (genetic and evolutionary computation)*: Springer-Verlag New York, Inc., 2006.
- [120] C. M. Fonseca, P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization", *Evolutionary Computation*, vol. 3, pp. 1, 1995.

- [121] H. Tamaki, H. Kita, S. Kobayashi, "Multi-objective optimization by genetic algorithms: a review", *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 517-522, Maio, 1996.
- [122] D. A. V. Veldhuizen, G. B. Lamont, *Multiobjective evolutionary algorithm research: a history and analysis*: Technical Report TR-98-03, Department of Electrical and Computer Engineering, Wright-Patterson AFB, Ohio, 1998.
- [123] W. M. Lin, F. S. Cheng, M. T. Tsay, "Distribution feeder reconfiguration with refined genetic algorithm", *Generation, Transmission and Distribution, IEE Proceedings-*, vol. 147, pp. 349-354, Novembro, 2000.
- [124] K. A. D. Jong, "Evolutionary computation. A unified approach," in *Artificial Life*, vol. 13: MIT Press, 2006, pp. 423-426.
- [125] N. G. Bretas, A. C. B. Delbem, A. Carvalho, "Restabelecimento de energia ótimo para sistemas de distribuição radiais por algoritmos genéticos", *Congresso Latino Americano de Distribuição de Energia Elétrica*, pp. 577-581, Setembro, 1998.
- [126] N. G. Bretas, A. C. B. Delbem, A. Carvalho, "Representação por cadeias de grafo para AG aplicados ao restabelecimento de energia ótimo em sistemas de distribuição radiais", *Revista SBA Controle & Automação*, vol. 12, pp. 42-51, 2001.
- [127] J. H. Holland, *Adaptation in natural and artificial systems*: University of Michigan, MIT Press, 1975.
- [128] K. Warwick, A. Ekwue, R. Aggarwal, *Artificial intelligence techniques in power systems*. London, UK: IEE Press, 1997.

- [129] N. Srinivas, K. Deb, "Multi-Objective function optimization using non-dominated sorting genetic algorithms", *Evolutionary Computation*, vol. 2, pp. 221-248, 1995.
- [130] E. Zitzler, L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm", *Technical Report 103, Computer Engineering and Networks Laboratory (TIK)*, Maio, 2001.
- [131] E. Zitzler, L. Thiele, "An evolutionary algorithm for multiobjective optimization: the strength pareto approach", *Technical Report 43, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Maio*, 1998.
- [132] B. W. Silverman, *Density estimation for statistics and data analysis*. London: Chapman and Hall, 1986.
- [133] C. A. C. Coello, "A comprehensive survey of evolutionary-based multi-objective optimization technique", *Knowledge and Information Systems*, vol. 1(3), pp. 269–308, 1999.
- [134] ANEEL, "Procedimentos de distribuição de energia elétrica no sistema elétrico nacional – PRODIST," vol. 8 - Qualidade de Energia Elétrica: Agência Nacional de Energia Elétrica - ANEEL, atualizada em Dezembro, 2008.
- [135] H. C. Cheng, C. C. Kou, "Network reconfiguration in distribution systems using simulated annealing", *Electric Power Systems Research*, vol. 29, pp. 227-238, Maio, 1994.
- [136] T. W. Lima, A. C. B. Delbem, *Estruturas de dados eficientes para algoritmos evolutivos aplicados a projeto de redes*. São Carlos, ICMC - USP: (Relatório Técnico), 2007.

Apêndice A

Dados de barra e de linha do SDR da *Taiwan Power Company*. Considere B_i - B_f sendo a barra inicial e final respectivamente, R sendo a resistência na linha, X a reatância na linha, P a carga ativa na barra final e Q a carga reativa na barra final. A Figura 7.22, com a topologia inicial do SDR da TPC, é reproduzida a seguir:

B _i -B _f	R (Ω)	X (Ω)	P (kW)	Q (kVAr)
A-1	0,1944	0,6624	0	0
1-2	0,2096	0,4304	100	50
2-3	0,2358	0,4842	300	200
3-4	0,0917	0,1883	350	250
4-5	0,2096	0,4304	220	100
5-6	0,0393	0,0807	1100	800
6-7	0,0405	0,1380	400	320
7-8	0,1048	0,2152	300	200
7-9	0,2358	0,4842	300	230
7-10	0,1048	0,2152	300	260
B-11	0,0786	0,1614	0	0
11-12	0,3406	0,6944	1200	800
12-13	0,0262	0,0538	800	600
12-14	0,0786	0,1614	700	500
C-15	0,1134	0,3864	0	0
15-16	0,0524	0,1076	300	150
16-17	0,0524	0,1076	500	350
17-18	0,1572	0,3228	700	400
18-19	0,0393	0,0807	1200	1000
19-20	0,1703	0,3497	300	300
20-21	0,2358	0,4842	400	350
21-22	0,1572	0,3228	50	20
21-23	0,1965	0,4035	50	20
23-24	0,1310	0,2690	50	10
D-25	0,0567	0,1932	50	30
25-26	0,1048	0,2152	100	60
26-27	0,2489	0,5111	100	70
27-28	0,0486	0,1656	1800	1300
28-29	0,1310	0,2690	200	120
E-30	0,1965	0,3960	0	0
30-31	0,1310	0,2690	1800	1600
31-32	0,1310	0,2690	200	150
32-33	0,0262	0,0538	200	100
33-34	0,1703	0,3497	800	600
34-35	0,0524	0,1076	100	60
35-36	0,4978	10,222	100	60
36-37	0,0393	0,0807	20	10
37-38	0,0393	0,0807	20	10
38-39	0,0786	0,1614	20	10
39-40	0,2096	0,4304	20	10
38-41	0,1965	0,4035	200	160
41-42	0,2096	0,4304	50	30
F-43	0,0486	0,1656	0	0
43-44	0,0393	0,0807	30	20
44-45	0,1310	0,2690	800	700
45-46	0,2358	0,4842	200	150
G-47	0,2430	0,8280	0	0
47-48	0,0655	0,1345	0	0

B _i -B _f	R (Ω)	X (Ω)	P (kW)	Q (kVAr)
48-49	0,0655	0,1345	0	0
49-50	0,0393	0,0807	200	160
50-51	0,0786	0,1614	800	600
51-52	0,0393	0,0807	500	300
52-53	0,0786	0,1614	500	350
53-54	0,0524	0,1076	500	300
54-55	0,1310	0,2690	200	80
H-56	0,2268	0,7728	0	0
56-57	0,5371	11,029	30	20
57-58	0,0524	0,1076	600	420
58-59	0,0405	0,1380	0	0
59-60	0,0393	0,0807	20	10
60-61	0,0262	0,0538	20	10
61-62	0,1048	0,2152	200	130
62-63	0,2358	0,4842	300	240
63-64	0,0243	0,0828	300	200
I-65	0,0486	0,1656	0	0
65-66	0,1703	0,3497	50	30
66-67	0,1215	0,4140	0	0
67-68	0,2187	0,7452	400	360
68-69	0,0486	0,1656	0	0
69-70	0,0729	0,2484	0	0
70-71	0,0567	0,1932	2000	1500
71-72	0,0262	0,0528	200	150
J-73	0,3240	11,040	0	0
73-74	0,0324	0,1104	0	0
74-75	0,0567	0,1932	1200	950
75-76	0,0486	0,1656	300	180
K-77	0,2511	0,8556	0	0
77-78	0,1296	0,4416	400	360
78-79	0,0486	0,1656	2000	1300
79-80	0,1310	0,2640	200	140
80-81	0,1310	0,2640	500	360
81-82	0,0917	0,1883	100	30
82-83	0,3144	0,6456	400	360
5-55	0,1310	0,2690		
7-60	0,1310	0,2690		
11-43	0,1310	0,2690		
12-72	0,3406	0,6994		
13-76	0,4585	0,9415		
14-18	0,5371	1,0824		
16-26	0,0917	0,1883		
20-83	0,0786	0,1614		
28-32	0,0524	0,1076		
29-39	0,0786	0,1614		
34-46	0,0262	0,0538		
40-42	0,1965	0,4035		
53-64	0,0393	0,0807		