

**UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO**

Restauração de Redes Elétricas Inteligentes
com Algoritmo de Otimização por Cultura de
Bactérias

Jonas Lopes de Vilas Boas

Itajubá, 14 de dezembro de 2018

**UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO**

Jonas Lopes de Vilas Boas

**Restauração de Redes Elétricas Inteligentes
com Algoritmo de Otimização por Cultura de
Bactérias**

Dissertação submetida ao Programa de Pós-Graduação em Ciência e Tecnologia da Computação como parte dos requisitos para obtenção do Título de Mestre em Ciências em Ciência e Tecnologia da Computação.

Área de Concentração: Matemática da Computação

Orientador: Dr. Carlos Henrique Valério Moraes

**14 de dezembro de 2018
Itajubá**

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO

Restauração de Redes Elétricas Inteligentes
com Algoritmo de Otimização por Cultura de
Bactérias

Jonas Lopes de Vilas Boas

Dissertação aprovada por banca examinadora em
14 de Dezembro de 2018, conferindo ao autor o
título de **Mestre em Ciências em Ciência e
Tecnologia da Computação.**

Banca Examinadora:

Dr. Carlos Henrique Valério Moraes (Orientador)

Dr. Alexandre Rasi Aoki

Dr. Zulmar Soares Machado Júnior

Itajubá

2018

Jonas Lopes de Vilas Boas

Restauração de Redes Elétricas Inteligentes com Algoritmo de Otimização por Cultura de Bactérias/ Jonas Lopes de Vilas Boas. – Itajubá, 14 de dezembro de 2018-

73 p. : il. (algumas color.) ; 30 cm.

Orientador: Dr. Carlos Henrique Valério Moraes

Dissertação (Mestrado)

Universidade Federal de Itajubá

Programa de Pós-Graduação em Ciência e Tecnologia da Computação, 14 de dezembro de 2018.

1. Algoritmos Bioinspirados.
2. Restauração de redes de distribuição elétricas.
3. Otimização multiobjetivo.
4. Algoritmo de otimização por cultura de bactérias.

CDU 07:181:009.3

Jonas Lopes de Vilas Boas

Restauração de Redes Elétricas Inteligentes com Algoritmo de Otimização por Cultura de Bactérias

Dissertação submetida ao Programa de Pós-Graduação em Ciência e Tecnologia da Computação como parte dos requisitos para obtenção do Título de Mestre em Ciências em Ciência e Tecnologia da Computação.

Trabalho aprovado. Itajubá, 14 de Dezembro de 2018:

Dr. Carlos Henrique Valério Moraes
Orientador

Dr. Alexandre Rasi Aoki

Dr. Zulmar Soares Machado Júnior

Itajubá
14 de dezembro de 2018

Agradecimentos

Agradeço à minha família pelo incentivo e conselhos em todos os momentos.

Aos meus colegas de mestrado pelo companheirismo e pelos momentos de distração.

Ao meu orientador Carlos Henrique Valério de Moraes pelo apoio na realização do trabalho e pelo conhecimento compartilhado.

À Universidade Federal de Itajubá por toda estrutura fornecida e à equipe da Pós Graduação em Ciência e Tecnologia da Computação, coordenação e professores, pelos ensinamentos e suporte.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, pela concessão da bolsa de mestrado.

“It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change.”
(Charles Darwin)

Resumo

Os sistemas responsáveis pela geração e fornecimento da energia elétrica devem operar respeitando os níveis mínimos nos índices de qualidade, como a continuidade no atendimento aos consumidores. Danos na infraestrutura das redes elétricas, causados por diversos fatores, podem causar interrupções no fornecimento de eletricidade. Visando diminuir os impactos causados, técnicas de restauração dessas redes vêm sendo desenvolvidas. O problema de restauração apresenta características combinatoriais, com múltiplos objetivos e restrições, o que dificulta o uso de técnicas de otimização determinísticas. Algoritmos meta-heurísticos, especialmente os algoritmos bioinspirados, têm apresentado bons resultados na resolução desse problema. Esse trabalho visa desenvolver e avaliar o desempenho de uma solução utilizando uma versão multiobjetivo do Algoritmo de Otimização por Cultura de Bactérias para encontrar a menor manipulação das chaves que maximiza a quantidade de barramentos atendidos, mantendo a configuração radial do sistema e respeitando os limites de corrente nos condutores e de tensão nos barramentos. Para isso foi criado um modelo do sistema elétrico e foram implementadas rotinas para a verificação da rede e cálculo de fluxo de potência por meio da varredura direta-inversa. O método foi aplicado sobre dois sistemas de 70 e 201 barramentos em condições de contingência com uma e duas falhas simultâneas. Os resultados obtidos foram comparados com outro trabalho de restauração de redes elétricas e a metodologia proposta neste trabalho encontra praticamente todas as soluções ótimas esperadas para todos os casos testados.

Palavras-chaves: Algoritmos Bioinspirados, Otimização Multiobjetivo, Reconfiguração de Redes Elétricas, Sistema de Distribuição.

Abstract

The systems responsible for generating and supplying electric energy should operate in accordance with minimum levels of quality indices, such as the continuity of service to consumers. Damage to the infrastructure of power grids, caused by many factors, can cause interruptions in the electricity supply. In order to reduce the impacts caused, restoration techniques have been developed. The problem of restoration presents combinatorial characteristics, with multiple objectives and restrictions, which makes difficult the use of deterministic optimization techniques. Metaheuristic algorithms, especially the bio-inspired algorithms, have presented good results in solving this problem. This work aims to develop and evaluate the performance of a solution using a multiobjective version of the Bacterial Foraging Optimization Algorithm to find the least key manipulation that maximizes the number of buses served, maintaining the radial configuration of the system and respecting the current limits conductors and busbar voltage. For this, a model of the electrical system was created and routines were implemented for the network verification and calculation of power flow through the direct-inverse sweep. The method was applied on two systems of 70 and 201 buses in conditions of contingency with one and two simultaneous faults. The results obtained were compared with other electrical network restoration technic and the methodology proposed in this work finds practically all the optimum solutions expected for all cases tested.

Key-words: Bio-inspired algorithms. Multi-Objective Optimization. Network Reconfiguration. Distribution System.

Lista de ilustrações

Figura 1 – Subdivisões de um SEP (SANCHES, 2013)	25
Figura 2 – Subdivisões do sistema de distribuição (SANCHES, 2013)	26
Figura 3 – Modelo de rede radial (MAZZETTO, 2016)	28
Figura 4 – Sequência de varredura para os métodos de busca em largura (a) e em profundidade (b)	29
Figura 5 – Estratégias de busca por alimentos (PASSINO, 2002)	35
Figura 6 – Fluxograma das etapas do BFOA	36
Figura 7 – Comportamento da <i>E. coli</i> no processo de quimiotaxia (PASSINO, 2002)	37
Figura 8 – Comparação entre configurações factíveis na montagem de computadores	40
Figura 9 – Fluxograma do método proposto	44
Figura 10 – Modelagem de uma rede de distribuição em grafo (MAZZETTO, 2016)	45
Figura 11 – Fluxograma das etapas do MOBFOA	49
Figura 12 – Processo de seleção da nova população (KAUR; KADAM, 2018)	50
Figura 13 – Configuração do sistema de 70 barramentos. Adaptado de (MAZZETTO, 2016)	56
Figura 14 – Sistema de 70 barramentos com falta em 25-26.	57
Figura 15 – Sistema de 70 barramentos após isolamento da falta em 25-26.	57
Figura 16 – Fronteira de Pareto para o sistema de 70 barramentos com falta em 25-26.	58
Figura 17 – Sistema de 70 barramentos com falta em 3-4.	59
Figura 18 – Fronteira de Pareto para o sistema de 70 barramentos com falta em 3-4.	60
Figura 19 – Sistema de 70 barramentos com faltas em 3-4 e 25-26.	60
Figura 20 – Fronteira de Pareto para a restauração do sistema de 70 barramentos com faltas em 3-4 e 25-26.	61
Figura 21 – Configuração ótima para o restabelecimento do sistema de 70 barramentos com faltas em 3-4 e 25-26.	62
Figura 22 – Sistema de 201 barramentos.	63
Figura 23 – Sistema de 201 barramentos com falta em 72-79. Adaptado de (LAMBERT-TORRES et al., 2009)	63
Figura 24 – Fronteira de Pareto para a restauração do sistema de 201 barramentos com falta em 72-79.	64
Figura 25 – Sistema de 201 barramentos com faltas em 9-10 e 20-49.	65
Figura 26 – Fronteira de Pareto para o sistema de 201 barramentos com faltas em 9-10 e 20-49.	66
Figura 27 – Configuração ótima para o sistema de 201 barramentos com faltas em 9-10 e 20-49.	66

Lista de tabelas

Tabela 1 – Parâmetros do MOBFOA	55
Tabela 2 – Soluções Pareto-ótimas para sistema de 70 barramentos com falta em 25-26	58
Tabela 3 – Soluções Pareto-ótimas para sistema de 70 barramentos com falta em 3-4	59
Tabela 4 – Soluções Pareto-ótimas para a restauração do sistema de 70 barramentos com faltas em 3-4 e 25-26	61
Tabela 5 – Soluções Pareto-ótimas para o sistema de 201 barramentos com falta em 72-79.	64
Tabela 6 – Soluções Pareto-ótimas para o sistema de 201 barramentos com faltas em 9-10 e 20-49.	65
Tabela 7 – Comparação dos resultados dos métodos MOBFOA e PSO - Única solução	67
Tabela 8 – Comparação dos resultados dos métodos MOBFOA e PSO - Várias soluções	68

Lista de abreviaturas e siglas

ACO	<i>Ant Colony Optmization</i>
BFOA	<i>Bacterial Foraging Optimization Algorithm</i>
CD	<i>Crowding Distance</i>
Cemig	Companhia Energética de Minas Gerais
GD	<i>Generation Distance</i>
GRASP	<i>Greed Randomized Adaptative Search Procedure</i>
HV	<i>Hypervolume</i>
IGD	<i>Inverted Generation Distance</i>
MOBFOA	<i>Multiobjective Bacterial Foraging Optimization Algorithm</i>
MOOP	<i>Multi-Objective Optimization Problem</i>
MOPSO	<i>Multiobjective Particle Swarm Optimization</i>
NA	Normalmente Aberta
NDS	<i>Non-Dominated Sorting</i>
NF	Normalmente Fechada
NSGA-II	<i>Non-Dominated Sorting Genetic Algorithm-II</i>
PCCS	<i>Pararel Cell Coordinate System</i>
PSO	<i>Particle Swarm Optimization</i>
RDEE	Rede de Distribuição de Energia Elétrica
RNP	Representação Nó-Profundidade
SEP	Sistema Elétrico de Potência
SOOP	<i>Single Objective Optimization Problem</i>
SPEA2	<i>Strenght Pareto Evolutionary Algorithm</i>

Lista de símbolos

ϵ	Limiar de erro de tensão para o fluxo de potência
Θ^i	Posição da i -ésima bactéria
ϕ	Direção do salto de uma bactéria
A	Repositório das soluções Pareto-ótimas
C	Tamanho do salto de uma bactéria
$Cell_{k,m}$	Quantidade de soluções na geração t com o <i>label</i> $L_{k,m}$
d_{atrac}	Profundidade da região de atratividade.
$D(P_i)$	Densidade de uma solução P_i
$DCP(P_i, P_j)$	Distância entre as células paralelas P_i e P_j
$E(t)$	Entropia das soluções na geração t
$f_m(x)$	m -ésima função objetivo para um problema de otimização multi-objetivo para a solução x
f_m^{max}	Maior valor para o objetivo m
f_m^{min}	Menor valor para o objetivo m
$gBest$	Conjunto de soluções ou solução ótima global para o algoritmo de otimização por enxame de partículas
h_{repel}	Altura da região de repulsividade.
I_{ij}	Corrente no ramo que liga os barramentos i e j
id	Indicador para uma chave de manobra em uma rede de distribuição elétrica
J_{cc}^i	Valor resultante da interação entre dois indivíduos para o indivíduo i
J_{saude}^i	Saúde acumulada da bactéria i
K	Tamanho do repositório das soluções Pareto-ótimas
$L_{k,m}$	Rótulo da solução k para o objetivo m no sistemas de coordenadas paralelas

$maxI_i$	Limite de corrente no ramo i
N_{barras}	Número de barramentos em uma rede de distribuição elétrica
N_{ch}	Número de etapas de quimiotaxia
N_{chaves}	Número de chaves de manobra em uma rede de distribuição elétrica
N_{ed}	Número de etapas de eliminação e dispersão
N_{obj}	Número de objetivos para um problema de otimização multi-objetivo
N_{pop}	Número de indivíduos em um população
NR_{des}	Número de restrições de desigualdade
NR_{igu}	Número de restrições de igualdade
N_{ramos}	Número de conexões em uma rede de distribuição elétrica
N_{rep}	Número de etapas de reprodução
N_s	Número máximo de passos numa mesma direção durante a varredura de um gradiente
N_{var}	Número de variáveis de projeto para um problema de otimização multi-objetivo
P	População de bactérias
P'	População de bactérias após a quimiotaxia
p_{ed}	Probabilidade de eliminação de dispersão
pu	Sistema por unidade, razão entre um o valor medido e o valor base
$pBest$	Conjunto de soluções ou solução ótima individual para o algoritmo de otimização por enxame de partículas
R	População de bactérias antes da seleção e reprodução
S_i	Potência demandada pelo barramento i
x	Uma das soluções de um problema de otimização multi-objetivo
$x_i^{(inf)}$	Limite inferior para a variável de projeto i
$x_i^{(sup)}$	Limite superior para a variável de projeto i
x	Uma das soluções de um problema de otimização multi-objetivo

V_{base}	Tensão de um alimentador usada para normalização das tensões
w_{atrac}	Largura da região de atratividade.
w_{repel}	Largura da região de repulsividade.
Z_{ij}	Impedância do condutor que liga os barramentos i e j

Sumário

1	INTRODUÇÃO	18
1.1	Objetivo	19
1.2	Organização do trabalho	20
2	REVISÃO TEÓRICA	21
2.1	Métodos heurísticos de busca	21
2.2	Métodos meta-heurísticos	22
2.3	Considerações Finais	23
3	RESTAURAÇÃO DE REDES ELÉTRICAS	25
3.1	Representação baseada em Grafos	27
3.2	Fluxo de Potência	29
4	ALGORITMOS BIO-INSPIRADOS PARA OTIMIZAÇÃO MULTI- OBJETIVO	31
4.1	Problemas de Otimização	31
4.2	Algoritmos de Otimização Bio-inspirados	33
4.2.1	Forrageamento	34
4.2.2	BFOA	35
4.2.2.1	Quimiotaxia	36
4.2.2.2	Enxame	37
4.2.2.3	Reprodução	38
4.2.2.4	Eliminação e dispersão	38
4.3	Problemas de Otimização Multi-Objetivo	39
4.3.1	Objetivos conflitantes	40
4.3.2	Dominância	40
4.3.3	Fronteira de Pareto	41
4.4	MOOP e algoritmos bio-inspirados	41
4.5	Considerações Finais	43
5	METODOLOGIA PROPOSTA	44
5.1	Modelagem computacional do sistema	44
5.1.1	Isolamento de Falhas	46
5.1.2	Fluxo de Potência	46
5.2	Objetivos e Restrições	47
5.3	MOBFOA	48

5.3.1	Etapas do MOBFOA	50
5.4	Aplicação do Algoritmo	53
6	RESULTADOS	55
6.1	Ajuste de Parâmetros	55
6.2	Sistema com 70 barramentos	56
6.2.1	Única falta	56
6.2.2	Duas faltas	59
6.3	Sistema 201b	62
6.3.1	Única falta	62
6.3.2	Duas faltas	64
6.3.3	Comparação entre métodos	64
7	CONCLUSÃO	69
	REFERÊNCIAS	71

1 Introdução

É indiscutível a importância da energia elétrica na sociedade atual. Praticamente toda a atividade socioeconômica depende da eletricidade para seu perfeito funcionamento, desde das atividades agrícolas, passando pela indústria e comércio, até as atividades cotidianas de toda população. Com a digitalização de processos e a internet, e com o consequente aumento no uso de dispositivos eletrônicos, como computadores e *smartphones*, e de infraestrutura de armazenamento e transmissão de dados digitais, como servidores e roteadores, a eletricidade passou a ter uma importância ainda maior.

Para produzir e entregar a energia elétrica para seus diversos tipos de consumidores, existem os sistemas elétricos de potência, os quais são sistemas complexos que movimentam estudos em diferentes áreas motivados a tornar esses sistemas mais eficientes e solucionar os seus problemas operacionais. Tecnologias de telecomunicações e ferramentas computacionais vêm sendo implementadas em diferentes aplicações, como é o caso das redes elétricas inteligentes, conhecidas como *smart grids*, que emergiram da aproximação dessas áreas e objetivam a melhoria dos sistemas elétricos atuais (BROWN, 2008). Equipadas com sistemas de monitoramento e atuação remota em equipamentos, as *smart grids* permitem o fornecimento de energia de uma forma controlada e inteligente, analisando, entre outros fatores, os padrões de consumo dos usuários, e podem integrar maneiras de geração de energia elétrica por meio de fontes alternativas e renováveis (FARHANGI, 2010; SIANO, 2014).

A RDEE (Rede de Distribuição de Energia Elétrica) é uma das partes dos sistemas elétricos de potências. Essas redes são o ponto de conexão de praticamente todos os consumidores e recebem cerca de 40% dos investimentos do setor de energia elétrica (SILVA, 2014). As empresas distribuidoras de eletricidade são responsáveis pela implantação e manutenção dessas redes e devem possuir redes eficientes e confiáveis para atender todas as demandas dos consumidores e se restabelecer depois que possíveis falhas ocorram, respeitando todas as normas regulamentares.

Os danos à infraestrutura das redes elétricas, causados por tempestades, acidentes de trânsito, vandalismo entre outros fatores, são muito comuns e podem causar interrupções no fornecimento de eletricidade. Em 2012 cerca de oito milhões de pessoas ficaram sem eletricidade durante a passagem do Furacão Sandy nos Estados Unidos. Estima-se que a economia dos Estados Unidos perde mais de 150 bilhões de dólares por ano devido à falhas no fornecimento de energia elétrica (GOLSHANI et al., 2017). Cerca de 80% dessas interrupções acontecem nas RDEE (SILVA, 2014).

Para simplificar a operação e a proteção e reduzir custos, as redes de distribuição

primárias operam geralmente em configurações radiais, sem a formação de malhas. Porém, tais configurações diminuem a confiabilidade do sistema quanto à continuidade no fornecimento de energia (SANTOS, 2009). Como todos os barramentos do sistema são alimentados apenas por uma única subestação, sempre que ocorre um defeito em alguma parte da rede o fornecimento é interrompido para todos os consumidores à jusante da área da contingência.

Para diminuir os impactos causados por essas interrupções foram desenvolvidas técnicas de recuperação do sistema de distribuição de energia, visando minimizar o tempo necessário para a restauração do fornecimento e minimizar o número de pessoas afetadas. Entre elas, as técnicas de otimização de inteligência computacional, como as técnicas bio-inspiradas, vêm sendo amplamente aplicadas para projetar plantas ótimas, encontrar o melhor configuração de chaves de restauração e selecionar a sequência ótima de operação dessas chaves.

Com a ampliação do uso das novas tecnologias das *smart grids* para monitoramento e atuação remota nas redes, técnicas computacionais eficientes serão cada vez mais importantes. O processo conhecido como *self-healing* vem sendo estudado para tornar as redes elétricas inteligentes capazes de se recuperarem de faltas de forma totalmente autônoma (BROWN, 2008; FARHANGI, 2010).

1.1 Objetivo

Esse projeto tem como objetivo geral desenvolver e avaliar o desempenho de uma metodologia para a resolução do problema de restauração de RDEE, visando encontrar soluções ótimas que minimizem a quantidade de chaveamentos realizados para restabelecer o máximo possível de consumidores, usando o Algoritmo de Otimização por Cultura de Bactérias.

Como objetivos específicos, foram considerados:

- Implementar e avaliar o desempenho de uma versão multiobjetivo do Algoritmo de Otimização por Cultura de Bactérias.
- Criar um modelo computacional para um Sistema Elétrico de Potência, considerando os barramentos alimentadores e consumidores, os condutores entre eles e as chaves de manobra.
- Implementar uma rotina para isolar as faltas presentes no sistema.
- Implementar uma rotina de cálculo de fluxo de potência por meio de varredura direta/inversa do sistema.

1.2 Organização do trabalho

O presente trabalho foi organizado da seguinte forma:

- No capítulo 1 é feita uma introdução sobre o problema de restauração de redes de distribuição elétrica, contendo as principais justificativas do desenvolvimento da proposta, e são apresentados os objetivos e a organização do trabalho.
- No capítulo 2 é feita uma revisão teórica de algumas metodologias aplicadas para o problema de restauração de RDEE.
- No capítulo 3 é feita uma breve apresentação sobre os sistemas elétricos de potência, enfatizando sobre as redes de distribuição e suas principais características, além de definir o problema de restauração de RDEE e apresentar algumas teorias usadas durante o processo de resolução computacional do mesmo.
- No capítulo 4 é feita uma introdução aos problemas de otimização e às técnicas de otimização bioinspiradas.
- No capítulo 5 são apresentados os detalhes do desenvolvimento da solução, incluindo informações sobre as técnicas e algoritmos utilizados.
- No capítulo 6 são demonstrados os resultados de alguns experimentos realizados e comparando com uma metodologia presente na literatura.
- Por fim, no capítulo 7 é feita a conclusão sobre os experimentos e possíveis melhorias são propostas como trabalhos futuros.

2 Revisão teórica

Devido à natureza combinatorial do problema da restauração de redes elétricas, as buscas por soluções ótimas para o problema de restauração em redes de grande porte por meios matemáticos determinísticos geralmente consomem muitos recursos computacionais e demandam muito tempo de execução. Dessa forma, as técnicas heurísticas e meta-heurísticas vêm sendo utilizadas em vários trabalhos desenvolvidos, como os citados nesse capítulo, e os algoritmos evolutivos e bio-inspirados têm um destaque especial em vários deles. Os trabalhos que serviram de inspiração para o desenvolvimento da presente proposta e que ilustram a aplicação das técnicas heurísticas e meta-heurísticas foram selecionados e são apresentados brevemente nas próximas seções.

2.1 Métodos heurísticos de busca

Os primeiros trabalhos na área de restabelecimento de redes elétricas utilizam métodos heurísticos. A metodologia proposta em (CIVANLAR et al., 1988), conhecida como “*branch exchange*”, é usada para o problema de redução de perdas de potência na etapa de planejamento e também para o restabelecimento de cargas após uma contingência. A técnica parte de uma configuração de chaves que mantém o sistema radial e que seja factível. Em seguida, uma chave é escolhida de maneira heurística, de acordo com fórmulas de redução de perdas, e é fechada, desfazendo a radialidade. Por fim, outro ramo é escolhido para ser aberto, de maneira que retorne o sistema à configuração radial.

Outra abordagem interessante é proposta em (SHIRMOHAMMADI, 1991). Nela, inicialmente todas as chaves são fechadas e uma rede malhada é criada. É executado um procedimento de fluxo de carga específico para esse tipo de rede e a chave com o menor fluxo é aberta. O procedimento se repete até que uma configuração radial do sistema seja encontrada e a área desconectada pela falta seja restabelecida.

Em (DAS, 2006) um algoritmo usando regras heurísticas e uma abordagem *fuzzy* multiobjetivo é proposta para a resolução do problema de reconfiguração de redes de distribuição elétrica. O método tem como objetivo balancear a carga entre os alimentadores, além de minimizar a perda de potência, o desvio de tensão nos barramentos e a violação de corrente nos ramos. Os objetivos são modelados como conjuntos *fuzzy* (*fuzzy sets*) e submetidos às heurísticas propostas, onde a radialidade do sistema deve ser mantida. O algoritmo foi testado em um sistema com 70 barramentos e obteve bons resultados.

2.2 Métodos meta-heurísticos

Os métodos meta-heurísticos, especialmente os algoritmos evolutivos e bio-inspirados que são a base da metodologia proposta nesse trabalho, são muito explorados para a resolução do problema de reestabelecimento de sistemas distribuição de energia. Em (TOUNE et al., 2002) as técnicas de Busca Tabu, Busca Tabu Reativa e *Simulated Annealing* são comparadas para buscar planos de restabelecimento que minimizem a quantidade de carga não atendida, respeitando as restrições de radialidade do sistema, queda de tensão e limite de carregamento nos ramos e transformadores. O estudo envolve comparações quantitativas e qualitativas da performance das técnicas apresentadas quando são executadas em sistema de 60 barras.

A proposta feita em (DELBEM; CARVALHO; BRETAS, 2005) é uma nova codificação da estrutura de dados baseada em grafos usada para modelar os sistemas de distribuição. Usando dois operadores genéticos o desempenho do algoritmo evolutivo proposto é melhorado, permitindo o seu uso para sistemas de larga escala. A técnica considera apenas uma falha por vez e pode avaliar múltiplos objetivos simultaneamente por meio de subpopulações. Os objetivos da proposta são minimizar o número de cargas desconectadas, o número de chaveamentos, a perda de potência e o custo total de equipamentos instalados. As soluções são restringidas pela queda de tensão e pelo carregamento nas linhas e transformadores. Os operadores genéticos utilizados geram apenas configurações factíveis, garantindo a radialidade do sistema.

Em (LAMBERT-TORRES et al., 2009) o método PSO (*Particle Swarm Optimization*) foi utilizado para solucionar o problema de restauração de redes elétricas. Os objetivos são minimizar a quantidade de chaveamentos e maximizar a quantidade de cargas atendidas, respeitando o limite de potência que pode ser transmitida em um ramo e mantendo a radialidade do sistema. Foi utilizada uma função de adaptação ponderando cada um dos objetivos. O conjunto de chaves foi dividido em chaves NA (Normalmente Aberta) e chaves NF (Normalmente Fechada). O método foi executado inicialmente apenas com as chaves NA, buscando alguma configuração que atenda todos os barramentos sem causar sobrecarga em algum ramo. Caso não encontre uma solução factível, o método é executado para as chaves NF, buscando eliminar as sobrecargas, e novamente para as chaves NA, buscando atender as cargas eliminadas com a abertura das chaves NF.

Em (MANSOUR et al., 2009) é proposto o uso da estrutura de dados RNP (Representação Nó-Profundidade) e seus operadores aliada ao método NSGA-II (*Non-Dominated Sorting Genetic Algorithm-II*) para explorar uma grande variedade de configurações de chaves que mantêm a radialidade do sistema e não deixa áreas desconectadas. Os objetivos são minimizar a quantidade de cargas não atendidas, operações de chaves e perda de potência. A proposta foi testada em um sistema real de grande porte e apresentou resultados satisfatórios em um curto intervalo de tempo.

A utilização da técnica de otimização meta-heurística GRASP (*Greedy Randomized Adaptive Search Procedure*) considerando a inserção de geradores distribuídos é proposta em (MATHIAS-NETO, 2011). Os objetivos da proposta são minimizar os consumidores sem fornecimento minimizando as operações de chaves e devem respeitar as restrições relacionadas com os limites operacionais do sistema, como corrente máxima nos equipamentos e condutores, fluxo de potência máximo, faixa aceitável de tensão, faixa de potência ativa fornecida pelos geradores distribuídos e seu fator de potência, equações de fluxo de potência e radialidade do sistema. A proposta apresenta resultados satisfatórios em um tempo aceitável para o problema de restauração.

A metodologia proposta em (SILVA, 2014) é testada para a restauração de sistema reais de grande porte disponibilizados pela Cemig (Companhia Energética de Minas Gerais). Utilizando a técnica evolutiva multiobjetivo SPEA2 (*Strength Pareto Evolutionary Algorithm*), a proposta entrega ao fim de cada iteração uma solução factível para reestabelecer um conjunto de cargas e no final da execução apresenta a solução que atender a maior quantidade de carga possível. O algoritmo leva em conta também o tempo necessário para a realização dos chaveamentos e permite que o operador tome a decisão mais conveniente sobre o conjunto de soluções encontradas.

Em (MATHIAS-NETO; MANTOVANI, 2016) é proposta uma abordagem utilizando a estrutura RNP com um novo operador para gerar vizinhanças para o método meta-heurístico de Busca Tabu. A proposta também leva em conta a presença da geração distribuída e cargas controladas remotamente. O objetivo da técnica é minimizar a função dos custos referentes à falta de abastecimento, respeitando as restrições operacionais do sistema, com a diferença de que o uso dos operadores RNP garante a operação radial do sistema, não sendo necessária uma restrição nesse sentido. O uso da estrutura de dados RNP melhora consideravelmente o tempo execução da técnica.

2.3 Considerações Finais

Os trabalhos destacados nesse capítulo apresentam características interessantes para o desenvolvimento de estratégias para o problema de restauração. A escolha da estrutura de dados adequada para a representação do sistema elétrico, por exemplo, impacta na eficiência do método tanto quanto a escolha dos métodos de otimização (DELBEM; CARVALHO; BRETAS, 2005; MANSOUR et al., 2009; MATHIAS-NETO; MANTOVANI, 2016). Uma boa combinação de estrutura de dados e método de otimização possibilita a resolução dos problemas de reconfiguração e restauração de redes elétricas para redes de grande porte. Usando uma mesma estrutura de dados, pode-se comparar o desempenho de diferentes técnicas de otimização durante o processo de busca por soluções para o problema de restauração (TOUNE et al., 2002). Além disso, o sistema e os casos

de contingência escolhidos para os teste podem ajudar na compreensão do problema e na verificação dos resultados (DAS, 2006; LAMBERT-TORRES et al., 2009).

Durante a busca pela melhores configurações do sistema, pode-se executar o método considerando inicialmente apenas manipulações no conjunto das chaves NA, fazendo com que as soluções sejam encontradas mais rapidamente. Caso essa primeira execução não seja suficiente, manipulações nas demais chaves do sistema podem ser feitas já considerando as respostas obtidas na execução anterior. Esse procedimento pode ajudar na abrangência da varredura do espaço de busca, mesmo que coloque o sistema temporariamente em estado que viola alguma restrição (CIVANLAR et al., 1988; SHIRMOHAMMADI, 1991; LAMBERT-TORRES et al., 2009). É importante considerar o custo que cada operação de chave pode causar para a concessionária e o tempo necessário para a aplicação da solução (SILVA, 2014), pois permite uma melhor tomada de decisão. A operação do restabelecimento para sistemas com geração distribuída deve considerar a capacidade dos geradores e as possíveis configurações de topologia que podem ser formadas com as manipulações de chaves (MATHIAS-NETO, 2011; MATHIAS-NETO; MANTOVANI, 2016).

3 Restauração de Redes Elétricas

O SEP (Sistema Elétrico de Potência) é composto por toda a infraestrutura destinada à geração da energia elétrica até a entrega para os consumidores finais. Ele pode ser subdividido em três sistemas, como mostra a figura 1 (NBR,). São eles:

- Sistema de Geração: composto pelas usinas de geração de energia.
- Sistema de Transmissão: composto pelas linhas de transmissão e transformadores. Conecta as unidades de geração às subestações de distribuição.
- Sistema de Distribuição: composto por subestações abaixadoras e circuitos elétricos, chamados de alimentadores.

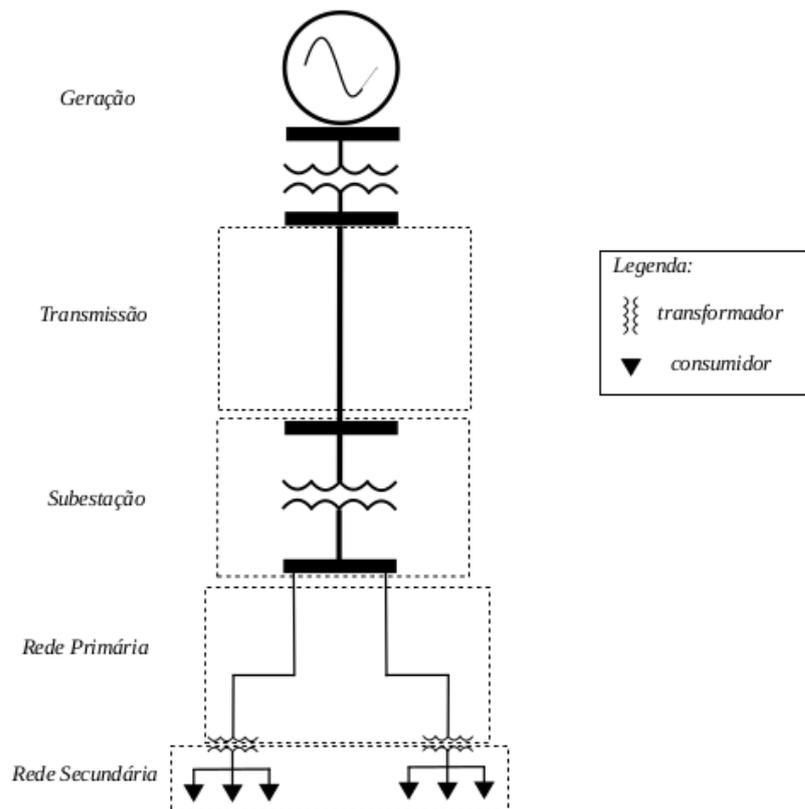


Figura 1 – Subdivisões de um SEP (SANCHES, 2013)

O sistema de distribuição é o ponto de acesso dos consumidores finais ao SEP. Ele é responsável pela conversão das altas tensões provenientes do sistema de transmissão em média ou baixa tensão para atender os diferentes perfis de consumo. Essa parte do sistema pode envolver sistemas de geração distribuída e seus respectivos sistemas de controle. A figura 2 ilustra um sistema de distribuição.

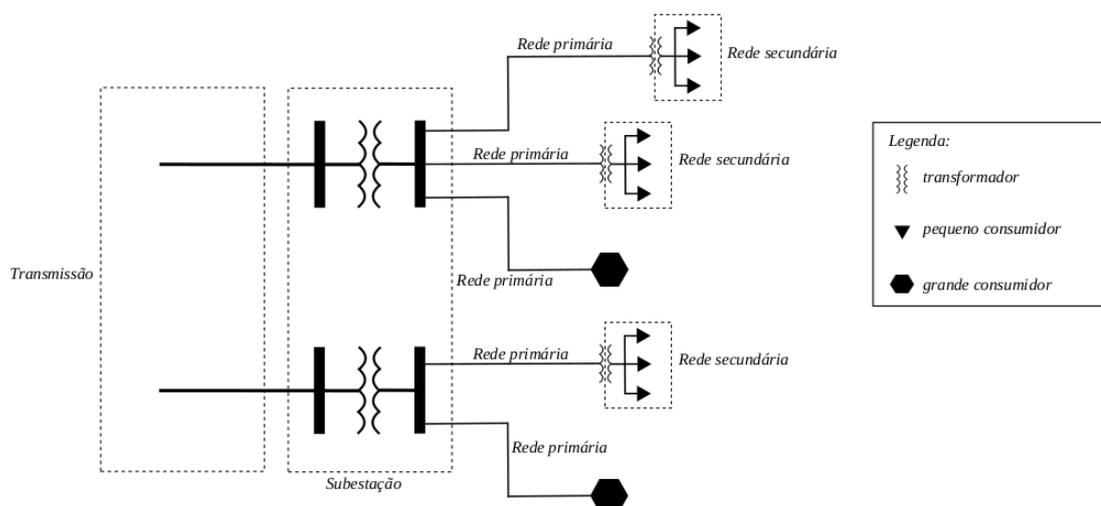


Figura 2 – Subdivisões do sistema de distribuição (SANCHES, 2013)

Cada parte do sistema de distribuição tem características e objetivos diferentes. A Rede de distribuição primária é a parte da rede mais próxima do sistema de transmissão. Ela pode operar com um nível alto de tensão, de $69kV$ até $138kV$, geralmente utilizada para interligação de subestações de distribuição, e não faz parte do escopo desse trabalho. Também opera em nível médio de tensão, de $13,8kV$ ou $34,5kV$, podendo atender consumidores primários como indústrias de médio porte e hospitais. Por meio de manobras de chaves seccionadoras, blocos de carga podem ser transferidos entre partes da rede durante uma contingência ou para realização de uma manutenção preventiva. Já a Rede de distribuição secundária opera com baixa tensão que pode ser $220/127V$ ou $380/220V$ e seus consumidores são os pequenos comércios e indústrias e, na maioria, consumidores domésticos.

Devido à configuração radial das redes primárias, após a ocorrência de uma falha ou desligamento para manutenção de uma parte do sistema, todos os ramais subsequentes a este ponto são afetados. Dessa forma, algumas regiões do sistema são conectadas por chaves de manobra. Esses dispositivos possibilitam uma reconfiguração do sistema, modificando a topologia da rede e restabelecendo o fornecimento para os consumidores desconectados.

O problema de restauração ou restabelecimento de RDEE consiste em encontrar uma configuração de chaves de manobra que restabeleça a maior quantidade de consumidores não atendidos após uma falha. Diferente do problema de reconfiguração de RDEE, que busca uma configuração mais otimizada para operar no sistema permanentemente, o problema de restauração busca colocar o sistema em um estado restaurativo, até que a falha seja corrigida (SANCHES, 2013).

Entre as diversas propostas desenvolvidas para esse problema, várias consideram

um ou mais dos objetivos e levam em consideração uma ou mais restrições. Entre os objetivos e restrições considerados durante a busca por melhores planos de restauração estão:

- Encontrar as melhores soluções no menor tempo possível;
- Realizar o menor número de chaveamentos;
- Restabelecer a maior quantidade de consumidores;
- Evitar sobrecarregar alguma parte do sistema;
- Manter os níveis operacionais de tensão nos barramentos;
- Balancear as cargas entre os alimentadores;
- Manter a topologia radial;
- Reduzir as perdas resistivas nos condutores;
- Reduzir quedas de tensão.

3.1 Representação baseada em Grafos

A estrutura de dados usada para modelar computacionalmente uma RDEE é um dos fatores de maior impacto na eficiência de uma proposta para a solução do problema de restauração (SILVA, 2014). As informações sobre os barramentos e sobre as conexões são constantemente consultadas e uma estrutura ineficiente de armazenamento dessas informações pode causar muita lentidão na busca por soluções.

Geralmente a modelagem de uma RDEE é feita por meio da teoria dos grafos. Um grafo é um conjunto $G = (V, A)$, onde V é um conjunto de vértices ou nós e A é um conjunto de arestas que conectam pares de elementos $u, v \in V$. No caso, um barramento é modelado como um nó e um condutor que conecta dois barramentos é modelado como uma aresta. Algumas definições da teoria de grafos são importantes para a modelagem de um sistema elétrico. São elas:

- Cadeia: sequência de arestas que liga um nó em outro qualquer;
- Ciclo: cadeia que começa e termina no mesmo nó e só passa uma vez por cada aresta;
- Grafo Acíclico: grafo onde nenhum ciclo é formado;
- Grafo Conexo: grafo onde há pelo menos uma cadeia ligando cada par de nós;
- Grafo Desconexo: grafo onde pelo menos um par de nós não é ligado por uma cadeia;

- Árvore: grafo conexo e acíclico;
- Floresta: grafo desconexo formado por subgrafos que são árvores.

Dessa forma, uma RDEE pode ser modelada como um floresta, onde cada árvore é conectada a uma subestação ou alimentador diferente, que fornece energia aos demais barramentos. Algumas das arestas podem ser chaves normalmente abertas ou fechadas. Elas podem ter seu estado alterado de acordo com a necessidade atual do sistema. A combinação de chaves abertas e fechadas deve manter a rede como uma floresta, além de respeitar as demais restrições.

A figura 3 mostra o modelo de um sistema onde os nós são 24 barramentos e 3 subestações conectados por 34 arestas, sendo 13 delas chaves, que podem ser abertas ou fechadas para a otimização do sistema.

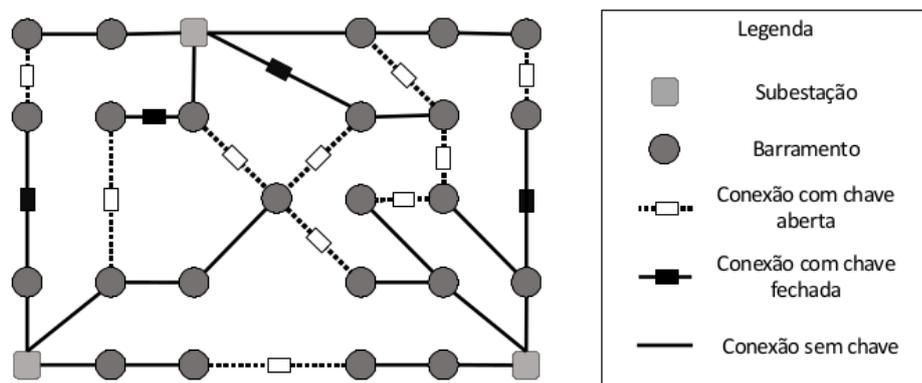


Figura 3 – Modelo de rede radial (MAZZETTO, 2016)

Algumas rotinas de caminhamento em grafos podem ser implementadas para auxiliar nos cálculos necessários para verificação das características da rede, como a busca em largura e a busca em profundidade. Partindo de um nó qualquer e considerando que cada nó de uma estrutura é visitado apenas uma vez na varredura:

- Busca em largura: cada um dos nós adjacente ao nó atual que ainda não foi marcado ou visitado é marcado e o nó atual é dado como visitado. Em seguida, o procedimento é repetido para cada um dos nós marcados até que o objetivo seja encontrado ou que todos os nós sejam visitados. Dessa forma, a cada passo dado na busca, os nós mais próximos do primeiro são visitados antes. A figura 4(a) mostra a sequência de visitas realizadas para a busca em largura.
- Busca em profundidade: um dos nós adjacentes ao nó atual que ainda não foi marcado ou visitado é marcado e o nó atual é dado como visitado. Em seguida, o procedimento

é repetido para o nó marcado. Caso não exista nenhum nó adjacente ao atual que ainda não foi marcado ou visitado, o procedimento retorna para o nó anterior ao atual, até que o objetivo seja encontrado ou que todos os nós sejam visitados. Dessa forma a busca se “aprofunda” no grafo a cada passo dado. A figura 4(b) mostra a sequência de visitas realizadas para a busca em profundidade.

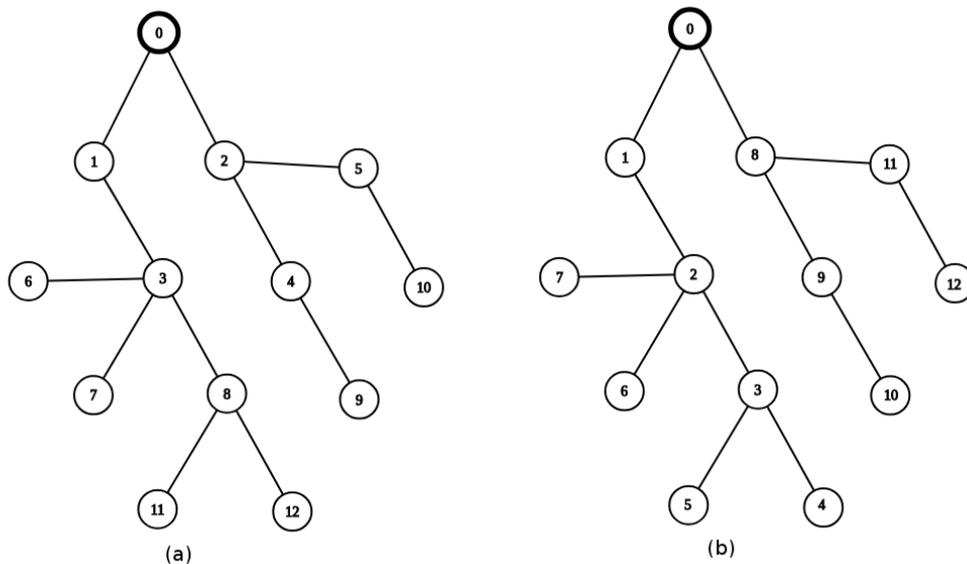


Figura 4 – Sequência de varredura para os métodos de busca em largura (a) e em profundidade (b)

De acordo com a aplicação, um tipo de busca é mais indicado. A busca em largura, por exemplo, sempre encontra o caminho mais curto entre dois vértice. A busca em profundidade, no entanto, tem uma implementação mais simples e pode ser utilizada para encontrar ciclos em um grafo, por exemplo.

3.2 Fluxo de Potência

O estudo de fluxo de potência é realizado para a obtenção das condições de operação do sistema. Pode-se usar esse método para saber a injeção de corrente e a tensão em cada barramento, as quedas de tensão nos condutores, por exemplo.

Entre os diversos métodos presentes na literatura, os mais utilizados para análise de sistemas de distribuição são os métodos de varredura direta/inversa. Esses métodos percorrem as redes de maneira direta, ou à jusante, ou seja, do alimentador até as cargas mais extremas, e da maneira inversa, ou à montante, e obtêm os valores em cada barramento e condutor.

Dois desses métodos são o da soma de correntes e o da soma de potência. Ambos utilizam as leis de Ohm iterativamente e ajustam os valores buscados por meio de

limiares ajustado inicialmente. Nesse trabalho o fluxo de potência é calculado pelo método da soma de correntes e uma rotina é apresentada na seção 5.1.2 (SHIRMOHAMMADI et al., 1988).

De maneira geral, o método da soma das correntes é iniciado ajustando a tensão de todos os barramentos como a tensão base. Em seguida é calculada a corrente no nós mais extremos, levando em conta a potência demandada por eles e a tensão já configurada. Tendo o valor para essa corrente, é calculada a corrente no nó à montante somando o valor de corrente obtido anteriormente. Esse processo é feito até o nó alimentador. O próximo passo é calcular as tensões em cada barramento dadas as quedas nos condutores, fazendo a varredura à jusante da rede. Com todas as tensões calculadas é feito a checagem do erro, subtraindo a tensão anterior com a atual. O procedimento é repetido até que o erro seja menor que um valor preestabelecido.

4 Algoritmos Bio-Inspirados para Otimização Multi-Objetivo

Problemas de otimização que consideram mais de um objetivo, conhecidos como MOOP (*Multi-Objective Optimization Problem*), são muito comuns em aplicações reais. Diferente do problema de otimização simples, ou SOOP (*Single-Objective Optimization Problem*), que considera maximizar ou minimizar uma única função objetivo, o MOOP visa encontrar combinações de variáveis, contínuas ou discretas, que resultem nas melhores soluções possíveis simultaneamente para vários objetivos que podem ser conflitantes. Nesse caso, existe um conjunto de soluções ótimas, chamado de Fronteira de Pareto (DEB, 2011).

Assim como os problemas de otimização, os métodos de solução são diversos e cada tipo de método é mais adequado para um certo tipo problema. Alguns problemas de otimização que envolvem variáveis discretas, chamados de problemas de otimização combinatória, como é o caso do problema de restauração de RDEE, são inviáveis de serem resolvidos por métodos de programação matemática ou puramente combinatórios (SANCHES, 2013).

A busca por métodos eficientes para a resolução de problemas de otimização combinatória complexos motiva inúmeros estudos, principalmente sobre técnicas heurísticas e meta-heurísticas. Nesse sentido, a Computação Natural é uma área de estudo que vem crescendo e vários métodos para a resolução de SOOP ou MOOP foram desenvolvidos nas últimas décadas (DEB, 2011).

Na próxima seção desse capítulo são apresentados alguns conceitos importantes sobre os problemas de otimização. Em seguida são introduzidos os algoritmos de otimização bio-inspirados e é feita uma breve apresentação do algoritmo usado como base na proposta desse trabalho. Na última seção são apresentados os conceitos de otimização multiobjetivo, algumas técnicas usadas por algoritmos evolutivos multiobjetivo e algumas métricas de avaliação de desempenho desses algoritmos.

4.1 Problemas de Otimização

Um problema de otimização tem como objetivo encontrar o valor ótimo, máximo ou mínimo, que uma função objetivo pode assumir em um dado intervalo. É um problema muito comum de engenharia, presente em diversas áreas. (BASTOS, 2004) apresenta alguns conceitos importantes em relação aos problemas de otimização, tais quais:

- Variáveis de estado: São as características do sistema a ser otimizado que se alteram

durante o processo, podendo assumir valores reais ou inteiros;

- **Função Objetivo:** É a função de uma ou mais variáveis de estado que se quer otimizar, minimizando-a ou maximizando-a;
- **Restrições:** São condições de igualdade ou desigualdade que descrevem situações de projeto consideradas não desejáveis;
- **Solução viável:** É o par formado pelo ponto indicado pela combinação das variáveis de projeto que respeita todas as restrições e maximiza ou minimiza o valor da função objetivo nesse ponto;
- **Espaço de busca:** É o conjunto, espaço ou região que compreende as soluções viáveis do problema a ser otimizado, delimitado pelas restrições;
- **Ponto Ótimo:** É o ponto formado pelas variáveis de estado que satisfazem as restrições e apresentam o melhor valor para as função objetivo;
- **Valor Ótimo:** É o valor da função objetivo no ponto ótimo;
- **Solução Ótima:** É o par formado pelo ponto ótimo e o valor ótimo, podendo ser uma solução ótima local ou global.

O método de resolução do problema de otimização está relacionado com as características da função objetivo e restrições. Em um problema de otimização contínua, por exemplo, existem infinitas posições no espaço de busca, gerando infinitas soluções factíveis. Esse tipo de problema geralmente apresenta gradientes positivos e negativos nos valores da função objetivo e os valores ótimos podem ser encontrados com métodos matemáticos derivativos nestes casos. Existem também problemas de otimização que envolvem variáveis de decisão inteiras, podendo ser binárias, e apresentam um conjunto de soluções discretas, chamados de problemas de otimização combinatória, como é o caso do problema de restauração de redes elétricas. Nesse caso, são utilizados métodos combinatórios que podem entregar soluções exatas, como os Algoritmos Gulosos e da Programação Dinâmica, ou aproximadas, como os Algoritmos Genéticos.

Os métodos de Programação Matemática podem ser classificados em dois tipos: Programação Linear, se a função objetivo e as restrições são funções lineares das variáveis de projeto e Programação Não-Linear, quando a função objetivo, ou pelo menos uma das restrições, é uma função não-linear. O Método Simplex, o Método do Gradiente Conjugado, o Método de Newton e o Método do Lagrangeano são exemplos de métodos matemáticos clássicos de otimização.

Apesar das garantias fornecidas pelos métodos clássicos, eles apresentam uma série de limitações, como é apresentado em (BASTOS, 2004):

- Dificuldade em encontrar ótimos globais dependendo do ponto de partida da execução do método, podendo resultar em soluções ótimas locais;
- Necessidade da função objetivo ser contínua e diferenciável, tornando necessária uma aproximação das variáveis de estado quando o método é aplicado a um problema de domínio discreto;
- Aplicação em domínio restrito para cada tipo de método;
- Necessidade de ponderação dos valores ótimos quando são envolvidos múltiplos objetivos, reduzindo a eficácia.

Dessa forma os métodos aproximativos, foram desenvolvidos e vêm sendo melhorados para resolver esses problemas de maneira viável. Entre essas técnicas estão os algoritmos bio-inspirados.

4.2 Algoritmos de Otimização Bio-inspirados

Algoritmos que modelam o comportamento de seres vivos ou algum fenômeno biológico, chamados de algoritmos bio-inspirados, partem do princípio de que vários processos da natureza sempre encontram estratégias ótimas. Eles são parte dos estudos da inteligência computacional, mais precisamente na área da computação natural. Usando condições iniciais e regras relativamente simples, são amplamente aplicados para aprendizado de máquina, classificação de elementos e problemas de otimização, entre outros.

Grande parte desses métodos utilizam populações de indivíduos que se espalham pelo espaço de busca, podendo procurar simultaneamente pelas melhor solução. Cada indivíduo representa uma solução factível. A maneira que esses indivíduos se movimentam no espaço de busca e são selecionados são específicos de cada método.

Alguns modelos usam como base o método evolutivo dos seres vivos, proposto por Darwin. Eles são chamados algoritmos evolucionários e modelam processos relacionados à evolução como transferência de genes entre gerações, cruzamento, mutação e seleção natural. Os algoritmos genéticos, proposto em (HOLLAND, 1973), é um exemplo de técnica que utiliza esses conceitos.

Outro exemplo de técnica inspirada na biologia é a inteligência de enxames. Essas técnicas tratam de modelar o comportamento social de algumas espécies de seres vivos, enquanto buscam por alimento ou se protegem de predadores. A técnica proposta em (EBERHART; KENNEDY, 1995), por exemplo, é baseada no movimento dos pássaros que voam em bandos, enquanto a proposta em (DORIGO; MANIEZZO; COLONI, 1996) usa como base a estratégia de busca de alimento das formigas.

A busca dos seres vivos por alimentos, ou forrageamento, tem inspirado o desenvolvimento de várias técnicas meta-heurísticas de otimização. Entre elas, está o algoritmo BFOA (*Bacterial Foraging Optimization Algorithm*), que serve como base para o desenvolvimento da proposta deste trabalho.

4.2.1 Forrageamento

Algumas das técnicas da computação natural são inspiradas nas estratégias dos animais na busca por alimento. Segundo os pesquisadores do campo da teoria de obtenção de alimentos (*foraging theory*), os animais e outros seres heterótrofos, como as bactérias, sempre tendem a tomar ações que maximizam a energia obtida durante o tempo gasto na busca por alimentos, respeitando suas limitações (cognitivas e sensoriais, por exemplo) e as limitações do ambiente (características físicas da região, risco de predadores e a presença de alimentos, por exemplo) (PASSINO, 2002).

Melhores estratégias na busca e obtenção de alimento aumentam as chances de sobrevivência e reprodução. Espécies carnívoras ou herbívoras mais eficientes em encontrar e ingerir boas concentrações de alimento, são mais eficientes em acumular energia para sobreviver e realizar outras atividades importantes, como lutar, dormir e reproduzir, garantindo a continuidade da espécie.

Essas estratégias podem estar relacionadas com a distância percorrida pelo indivíduo por intervalo de tempo. Alguns animais usam a estratégia de “cruzeiro” e estão sempre buscando alguma presa através do ambiente, outros usam a estratégia de “emboscada”, passando muito tempo esperando pela aproximação da presa. A maioria dos animais têm estratégias classificadas entre esses dois extremos, chamadas estratégias do tipo “saltatório”, que buscam pequenas concentrações de alimento e permanecem na localidade por um tempo antes de “saltar” para outro ambiente. A figura 5 mostra a relação entre o deslocamento e o tempo para cada tipo de estratégia.

O comportamento social entre os indivíduos de uma espécie também é muito importante na forma em que eles obtêm alimentos. Por meio da comunicação entre os indivíduos, eles podem aumentar sua chance de encontrar alimento, caçar presas maiores e se proteger de predadores. A associação entre indivíduos é muito comum no reino animal e a inteligência desses coletivos são amplamente estudadas e usadas como modelos em diversas aplicações, como é o caso dos bandos de pássaros, dos cardumes de peixes, das colônias de formigas e das alcateias de lobos, por exemplo.

Com base nos conceitos da *foraging theory*, que incluem alguns dos conceitos de reprodução e seleção natural abordados pelos algoritmos evolutivos e dos conceitos propostos pelos algoritmos de inteligência de enxame, em (PASSINO, 2002) é apresentado um novo método de otimização chamado de BFOA.

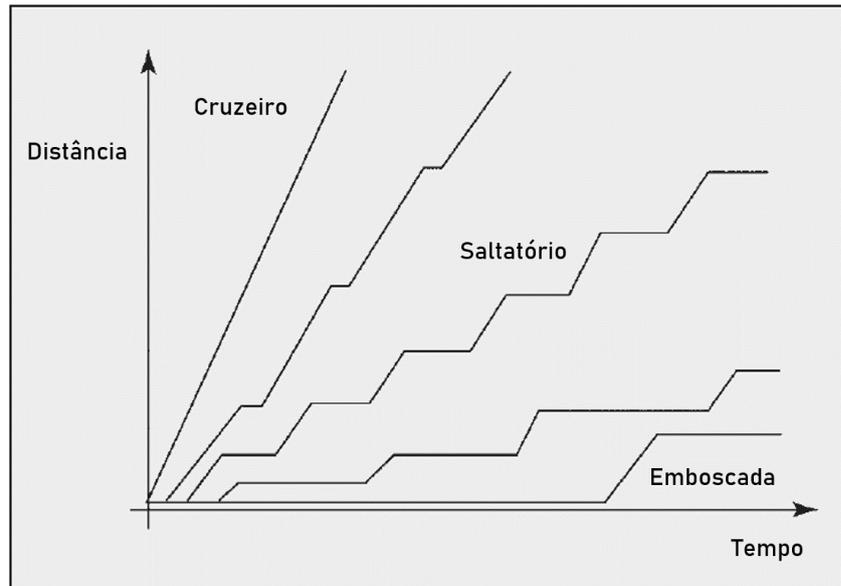


Figura 5 – Estratégias de busca por alimentos (PASSINO, 2002)

4.2.2 BFOA

O BFOA, proposto por (PASSINO, 2002), é um algoritmo de otimização baseado na ideia de inteligência de enxames e modela o comportamento da bactéria *Escherichia coli*. Encontrada no intestino humano, a *E. coli* usa flagelos para se movimentar e é provavelmente o microrganismo melhor compreendido pela ciência.

O algoritmo representa a busca dessas bactérias por nutrientes ou a distanciação das toxinas, a interação entre as bactérias e a sua reprodução. Enquanto muda de posição no espaço de busca, que pode ter várias dimensões, cada indivíduo de uma população de bactérias encontra uma certa combinação de nutrientes e toxinas. No procedimento de otimização, essa combinação é o valor da função objetivo que se pretende maximizar ou minimizar quando as variáveis de estado são as coordenadas da posição da bactéria em uma determinada iteração.

Quatro processos são implementados para modelar o comportamento da população de bactérias enquanto realiza a busca. São eles: *Chemo taxis* (ou quimiotaxia), *Swarming* (a interação entre indivíduos em um enxame), *Reproduction* (ou reprodução) e *Elimination and Dispersion* (ou eliminação e dispersão).

A figura 6 mostra um fluxograma simplificado do método. Antes de realizar essas etapas, uma população P é criada com um número de N_{pop} indivíduos posicionados aleatoriamente pelo espaço de decisão. Por padrão, os cálculos mostrados a seguir, buscam a minimização da função objetivo.

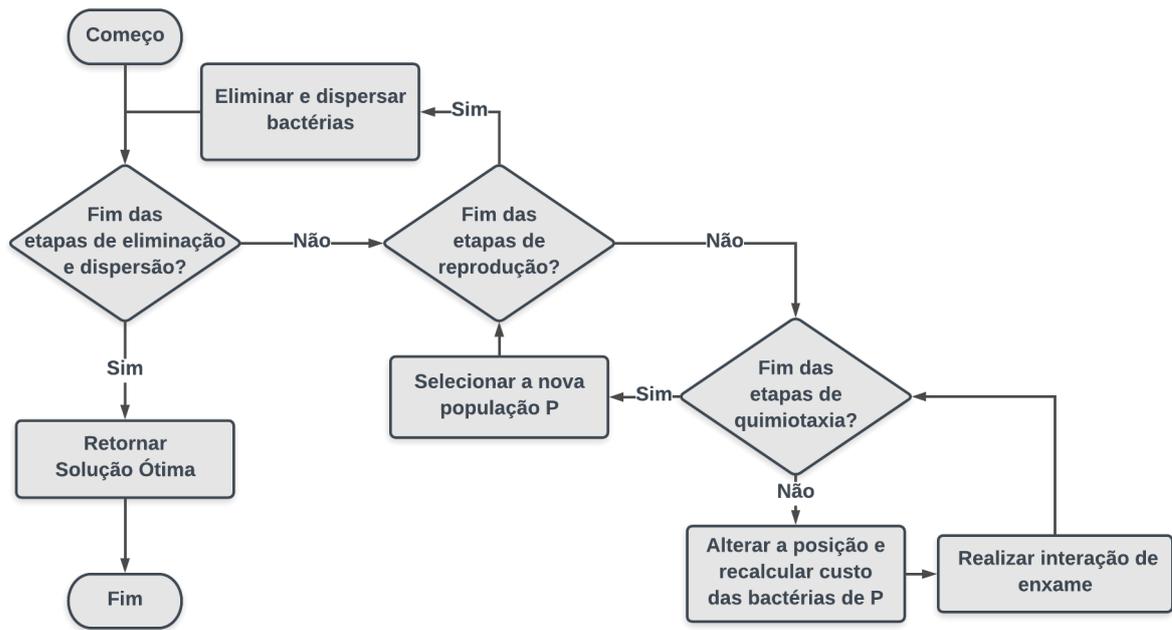


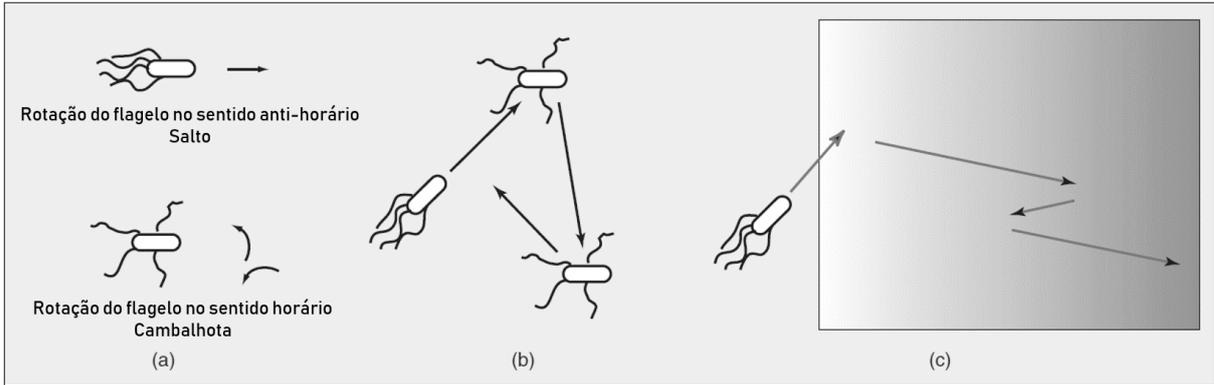
Figura 6 – Fluxograma das etapas do BFOA

4.2.2.1 Quimiotaxia

A estratégia de busca de alimentos das bactérias é do tipo saltatório e está diretamente relacionada com sua mobilidade. Entre os tipos de bactérias, existem as que independem de flagelo para se moverem e as que se movem por meio flagelos, como a *Escherichia coli*. Biólogos descobriram em meados da década de 70 que o movimento das bactérias não era aleatório, mas sim atraído para uma direção ou repellido dela por meio de estímulos e chamaram esse comportamento de quimiotaxia (XING; GAO, 2014).

A *E. coli* pode se mover de duas formas diferentes: avançando em uma direção ou mudando de direção. Como é mostrado na figura 7(a), a bactéria gira seus flagelos no sentido anti-horário para realizar um avanço (ou *swim*), e no sentido horário para realizar uma mudança de direção (ou *tumble*). Ela alterna entre esses dois estados durante seu período de vida, mudando aleatoriamente de direção, o que permite a varredura do espaço em busca de nutrientes, como pode ser visto na figura 7(b). Se a bactéria detecta um gradiente de nutrientes, como na figura 7(c), e avança em direção à maior concentração, ela tende a passar mais tempo avançando do que mudando de direção.

Nessa etapa do algoritmo, a alteração da posição das bactérias é feita pela equação 4.1, onde Θ^i é a posição da i -ésima bactéria da população, $\phi(j)$ é a direção aleatória tomada depois da cambalhota, $C(i)$ é o tamanho do avanço tomado na direção $\phi(j)$, j é a etapa do processo de quimiotaxia, k é a etapa do processo de reprodução, l é a etapa


 Figura 7 – Comportamento da *E. coli* no processo de quimiotaxia (PASSINO, 2002)

do processo de eliminação e dispersão.

$$\Theta^i(j + 1, k, l) = \Theta^i(j, k, l) + C(i) \times \phi(j) \quad (4.1)$$

É calculado o custo J , dado pela função objetivo para a bactéria i na nova posição e a interação de enxame com as demais bactérias é realizada. Se o custo da posição $\Theta^i(j + 1, k, l)$ é melhor que o custo da posição $\Theta^i(j, k, l)$, a bactéria realiza mais um avanço de tamanho $C(i)$ nessa mesma direção $\phi(j)$. Esse processo define a varredura da bactéria em um gradiente e é repetido enquanto o custo da nova posição for melhor que o anterior ou se a quantidade de passos tomados for menor que N_s , ajustado inicialmente.

O processo de quimiotaxia é feito para cada uma das N_{pop} bactérias da população e é repetido N_{ch} vezes.

4.2.2.2 Enxame

Vários tipos de bactéria apresentam comportamento de enxame. No caso da *E. coli*, quando submetida à uma condição de estresse, ela libera um sinal na direção das outras bactérias. Esse sinal pode atrair ou repelir as demais, de acordo com a situação em que está submetida. No processo de otimização, essa interação faz com que o valor da função objetivo para uma bactéria, dado pelo custo J , receba um decréscimo (considerando a minimização como padrão) caso ela esteja atraindo as outras ou um acréscimo caso ela esteja repelindo.

A equação 4.2 mostra a alteração causada no valor do custo, onde J_{cc} é o fator de atração/repelimento que a bactéria i causa na população P . Uma nova interação entre os indivíduos é feita a cada mudança de posição.

$$J(i, j + 1, k, l) = J(i, j + 1, k, l) + J_{cc}(\Theta^i(j + 1, k, l), P(j + 1, k, l)) \quad (4.2)$$

Essa alteração dos valores da função objetivo cria uma região no espaço de busca em torno de cada bactéria. As demais bactérias são atraídas ou relidas de acordo com essa região. O valor J_{cc} é dada pela equação 4.3, onde p é a quantidade de dimensões do espaço de decisão, d_{atrac} é a profundidade e w_{atrac} é a largura da região de atratividade, $h_{repel} = d_{atrac}$ é a altura e w_{repel} é a largura da região de repulsividade.

$$\begin{aligned}
J_{cc}(\Theta, P(j, k, l)) &= \sum_{i=1}^{N_{pop}} J_{cc}^i(\Theta, \Theta^i(j, k, l)) \\
&= \sum_{i=1}^{N_{pop}} (-d_{atrac} e^{(-w_{atrac} \sum_{m=1}^p (\Theta_m - \Theta_m^i)^2)}) \\
&\quad + \sum_{i=1}^{N_{pop}} (h_{repel} e^{(-w_{repel} \sum_{m=1}^p (\Theta_m - \Theta_m^i)^2)})
\end{aligned} \tag{4.3}$$

O valor de saúde J_{saude} de uma bactéria i , dada pela equação 4.4, indica quanto nutriente ela acumula durante o processo de quimiotaxia. Esse valor é usado para selecionar as bactérias na etapa de reprodução.

$$J_{saude}^i = \sum_{j=1}^{N_{ch}+1} J(i, j, k, l) \tag{4.4}$$

4.2.2.3 Reprodução

As bactérias mais saudáveis e que acumularam mais nutrientes tem maior chance de se reproduzir. As bactérias *E. coli* são bactérias procariontes e se reproduzem de forma assexuadas, realizando a cópia do cromossomo e dividindo a célula em duas, sem mutações. De maneira contrária, as bactérias que absorveram menos nutrientes ou que não conseguiram evitar as toxinas são eliminadas.

Para modelar esse processo, após todas as etapas de quimiotaxia, os valores de saúde de todas as bactérias, dado pelo valor J_{saude} , são ordenados de maneira crescente. O conjunto é dividido em dois e a metade com os melhores valores de saúde sobrevive, enquanto a outra metade é substituída pela réplica da metade melhor. A quantidade de iterações de reprodução é definida pela constante N_{rep} ajustada inicialmente.

4.2.2.4 Eliminação e dispersão

O ambiente em que as bactérias estão buscando por alimento pode mudar gradualmente ou instantaneamente, fazendo com que alguns indivíduos se dispersem para outras regiões durante essa mudança.

Nas iterações desse processo, cada bactéria é eliminada com uma probabilidade p_{ed} e outra é posicionada aleatoriamente no espaço de busca. Esse processo é realizado

por N_{ed} iterações. Um alto valor N_{ed} aumenta a complexidade da solução, mas também permitem uma melhor varredura do espaço de busca, diminuindo a chance da solução encontrada ser um ótimo local. O valor de p_{ed} não deve ser grande, pois levaria a uma busca exaustiva.

4.3 Problemas de Otimização Multi-Objetivo

Os problemas de otimização multi-objetivo, ou MOOP, envolvem a otimização de dois ou mais objetivos simultaneamente, os quais podem ser conflitantes, ou seja, a melhoria de um dos objetivos causa a degradação de outro. Além disso esses problemas podem possuir restrições de igualdade e de desigualdade que devem ser satisfeitas para que uma solução seja factível. De maneira geral, um MOOP é enunciado conforme equação 4.5 (SILVA, 2014).

$$\begin{aligned}
 & \text{Maximizar/Minimizar } f_m(x), & m = 1, 2, \dots, N_{obj} \\
 & \text{sujeito a : } g_j(x) \geq 0, & j = 1, 2, \dots, NR_{des} \\
 & h_k(x) = 0, & k = 1, 2, \dots, NR_{igu} \\
 & x_i^{(inf)} \leq x_i \leq x_i^{(sup)}, & i = 1, 2, \dots, N_{var}
 \end{aligned} \tag{4.5}$$

onde x é o conjunto de N_{var} variáveis de projeto, também denominado de solução. Os valores máximo e mínimo para cada elemento x_i são dados por $x_i^{(sup)}$ e $x_i^{(inf)}$, respectivamente, que delimitam o espaço de decisão. As NR_{des} desigualdades e as NR_{igu} igualdades são as restrições. Uma solução x que satisfaz todas as restrições é chamada de solução factível. O conjunto de todas as soluções factíveis é chamado de espaço de busca.

Em uma aplicação real, cada uma das N_{obj} funções objetivo $f_m(x)$ podem ser maximizadas ou minimizadas independentemente. Porém os algoritmos de otimização exigem que as funções sejam apenas de maximização ou minimização. Para isso, pode ser feita a negação da função, alterando seu objetivo para se adequar ao método. Uma técnica bastante usada nas técnicas de otimização é a definição de um valor de normalização para facilitar a comparação dos resultados. Assim, pode-se facilmente padronizar as funções objetivo como maximização ou minimização.

Um exemplo de MOOP com dois objetivos conflitantes é o processo de tomada de decisão na montagem de um computador, considerando o desempenho e o preço. Nesse caso, os objetivos são maximizar o desempenho e minimizar o preço. Para tornar ambos os objetivos como maximização, foi definido que o valor máximo a ser gasto na máquina é de 10 mil reais e que os valores dados são a porcentagem desse total economizada por cada máquina.

Adicionando melhores componentes de *hardware*, pode-se melhorar o desempenho da máquina, o que também altera seu preço. Então, as variáveis do projeto são os componentes a serem adicionados ou removidos. Como restrições desse problema, estão as combinações entre componentes, ou seja, alguns componentes não podem funcionar corretamente se combinados com outros. Uma solução factível, portanto, é uma combinação de componentes que funcione perfeitamente em conjunto. As melhores soluções são as combinações que têm melhores desempenhos com menores preços. A figura 8 mostra algumas soluções, relacionando seu desempenho com seu preço.

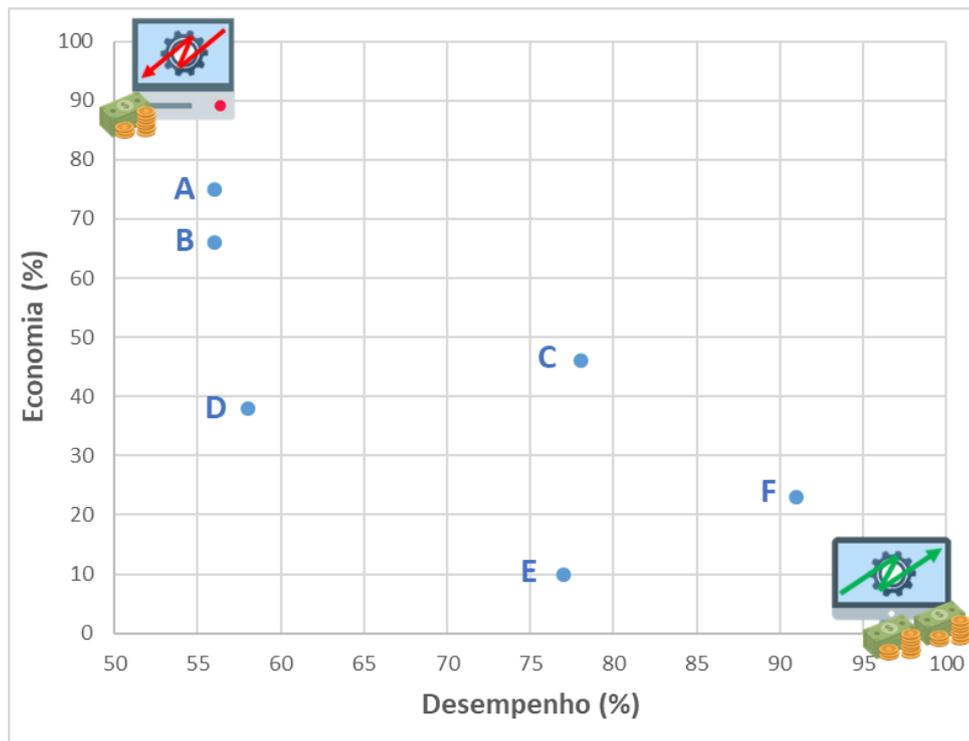


Figura 8 – Comparação entre configurações factíveis na montagem de computadores

4.3.1 Objetivos conflitantes

Os objetivos em um MOOP geralmente têm uma relação de conflito entre si. Se, sempre que o valor de uma função objetivo f_1 é melhorado, conseqüentemente, o valor de outra função f_2 piora, pode-se dizer que f_1 é conflitante com f_2 . No exemplo da figura 8, uma máquina com melhores componentes de *hardware* tem melhor desempenho. Porém a adição desses componentes aumenta o preço. Desta forma, o aumento do desempenho é conflitante com a redução do preço.

4.3.2 Dominância

Um MOOP pode ter mais de uma solução ótima. Uma solução não pode ser avaliada como melhor que outra analisando apenas uma das funções objetivos. Observando

a figura 8 pode-se dizer que a máquina A é melhor que a máquina B, pois ela consegue atingir o mesmo desempenho com o um custo menor. Porém a combinação A tem desempenho pior que C e ao mesmo tempo tem o preço melhor. Nesse caso, não pode-se afirmar que a solução A é melhor que a solução C ou o inverso. Por isso, utiliza-se o conceito de dominância de Pareto para comparar duas soluções factíveis de um MOOP.

Dadas duas soluções factíveis x e y , é dito que x domina y , ou $x \prec y$, se:

1. A solução x não é pior que y em todos os objetivos;
2. A solução x é melhor que y em pelo menos um objetivo.

4.3.3 Fronteira de Pareto

Levando em conta o conceito de dominância, sempre existe um conjunto de soluções factíveis de um MOOP que são não dominadas entre si, ou seja, que apresentam melhores resultados em um objetivo e piores em outro. Esse conjunto é chamado de Pareto-ótimo e os valores das funções objetivo das soluções desse conjunto formam a fronteira de Pareto. Na figura 8, as soluções A, C e F formam o conjunto Pareto-ótimo.

4.4 MOOP e algoritmos bio-inspirados

Como já discutido no início desse capítulo, os algoritmos bio-inspirados apresentam uma série de vantagens em relação aos métodos clássicos de otimização. Entre elas, valem ser destacadas a capacidade de evitar ótimos locais e a possibilidade desses algoritmos de trabalhar com múltiplos objetivos. Apesar das implementações bio-inspiradas serem originalmente feitas para otimizar uma única função objetivo, as adaptações necessárias para resolver um MOOP são simples e vêm sendo realizadas por diversos pesquisadores (ZITZLER; LAUMANN; THIELE, 2001; DEB et al., 2002; NIU et al., 2013; HU; YEN, 2015; YI et al., 2016; KAUR; KADAM, 2018).

Os já citados algoritmos genéticos são amplamente utilizados para otimização multi-objetivo. Devido a sua simplicidade de implementação e flexibilidade para adaptações, alguns pesquisadores propuseram técnicas para atingir melhores resultados na resolução de um MOOP. Duas dessas técnicas, – que serão apresentadas em detalhes a seguir, são a Ordenação por Não-Dominância ou NDS (*Non-Dominated Sorting*) e ordenação por Distância de Multidão ou CD (*Crowding Distance*). Essas técnicas compõem a metodologia proposta em (DEB et al., 2002) chamada de NSGA-II, muito citada na literatura, inclusive usada em (MANSOUR et al., 2009) e em (MARQUES et al., 2017) para a resolução do problema de restauração de RDEE.

Outros algoritmos evolutivos também são amplamente aplicados na otimização multiobjetivo. Muitas versões dessas técnicas foram desenvolvidas e muitas vezes elas são combinadas com outras metodologias para melhorar os resultados em aplicações específicas. Um método genérico para MOOP é apresentado em (ZITZLER; LAUMANN; THIELE, 2001), chamado de SPEA2 (*Strength Pareto Evolutionary Algorithm*). Nesse método é atribuída a cada solução uma "força", que representa a quantidade de soluções que ela domina de acordo com a dominância de Pareto. Um exemplo da aplicação do SPEA2 é apresentada em (SILVA, 2014) e em (SANCHES, 2013) duas versões de algoritmos evolutivos são estudadas e outras duas são propostas, em todos os casos para o problema de restauração e/ou reconfiguração de RDEE.

A técnica PSO, também já citada, foi adaptada para resolver um MOOP por diversos autores. Chamada de MOPSO (*Multiobjective Particle Swarm Optimization*), essa adaptação avalia um conjunto de melhores soluções globais e locais, por meio dos repositórios *gBest* e *pBest*, diferente do método com um único objetivo. Isso faz com que cada indivíduo da população seja guiado por soluções diferentes e afeta diretamente na convergência do método e na diversidade das soluções. Pensando nisso, em (HU; YEN, 2015) é proposto um MOPSO que usa um Sistema de Coordenadas de Células Paralelas, ou PCCS (*Parallel Cell Coordinate System*), para avaliar a qualidade das soluções a cada geração e adaptar os parâmetros do algoritmo com base neste *feedback*, melhorando a convergência e preservando a diversidade das soluções, ou seja, um garantindo um melhor espalhamento na fronteira de Pareto.

Alguns trabalhos também têm proposto versões do BFOA para otimização multi-objetivo. A principal mudança na técnica original é a questão da reprodução das bactérias. Em (NIU et al., 2013) é desenvolvido uma técnica de seleção das bactérias que vão se reproduzir, baseada na saúde e na dominância de Pareto dos indivíduos. O proposto em (YI et al., 2016) é a mesma estratégia de seleção porém com a utilização de um repositório de melhores soluções e uma adaptação do tamanho do salto das bactérias a cada geração, semelhante ao proposto em (HU; YEN, 2015). Usando os conceitos do NSGA-II, em (KAUR; KADAM, 2018) é proposta a seleção dos indivíduos para reprodução através da NDS, seguida de uma etapa de ordenação por CD e uma ordenação por força de Pareto adaptada, que leva em conta as soluções dominadas e dominantes de cada bactéria. Esse último método ainda exclui a interação de enxame das bactérias e a repetição de quimiotaxia, fazendo a cada mudança de posição e avaliação uma nova seleção e reprodução.

Nesse trabalho é proposta uma nova versão do MOBFOA (*Multiobjective Bacterial Foraging Optimization Algorithm*) que aproveita os melhores aspectos de cada um dos trabalhos estudados, e será abordado em detalhes no próximo capítulo.

4.5 Considerações Finais

Os algoritmos bioinspirados são indicados para a resolução de um MOOP e vários métodos, como o NSGA-II e o SPEA2, foram propostos e vêm sendo estudados e melhorados nos últimos anos. Algumas propostas de versões multiobjetivo do BFOA também foram propostas, como em (NIU et al., 2013) e em (KAUR; KADAM, 2018). Como foi discutido nesse capítulo, para o desenvolvimento de um método bioinspirado de otimização para múltiplos objetivos conflitantes alguns fatores têm que ser levados em consideração:

- Uma solução é considerada melhor que outra respeitando a relação de dominância de Pareto;
- As soluções Pareto-ótimas são as soluções que não dominantes entre si e dominantes em relação a todas as outras soluções;
- A ordenação por não-dominância é uma maneira de ranquear as soluções em uma população de acordo com a dominância de Pareto, fazendo a separação em frentes de diferentes *ranks*;
- A ordenação por distância de multidão pode ser usada para ranquear as melhores soluções em uma frente específica e é usada para manter o espalhamento e a diversidade das soluções;
- O mapeamento do conjunto Pareto-ótimo no sistema PCCS e o cálculo da entropia ajuda na definição do estado da população em uma determinada geração como sendo de convergência, diversidade ou estagnação, e pode ser usado como fator de adaptação de parâmetros dos métodos de otimização.

5 Metodologia Proposta

A metodologia proposta envolve a modelagem do sistema a ser restaurado, o isolamento das regiões envolvidas nas faltas, o cálculo do fluxo de potência para uma determinada configuração do sistema e a busca por combinações de chaveamentos factíveis e ótimas que restabelecem a maior parte das cargas do sistema. A figura 9 mostra um fluxograma do método proposto. Todas as rotinas do protótipo foram implementadas na linguagem C.

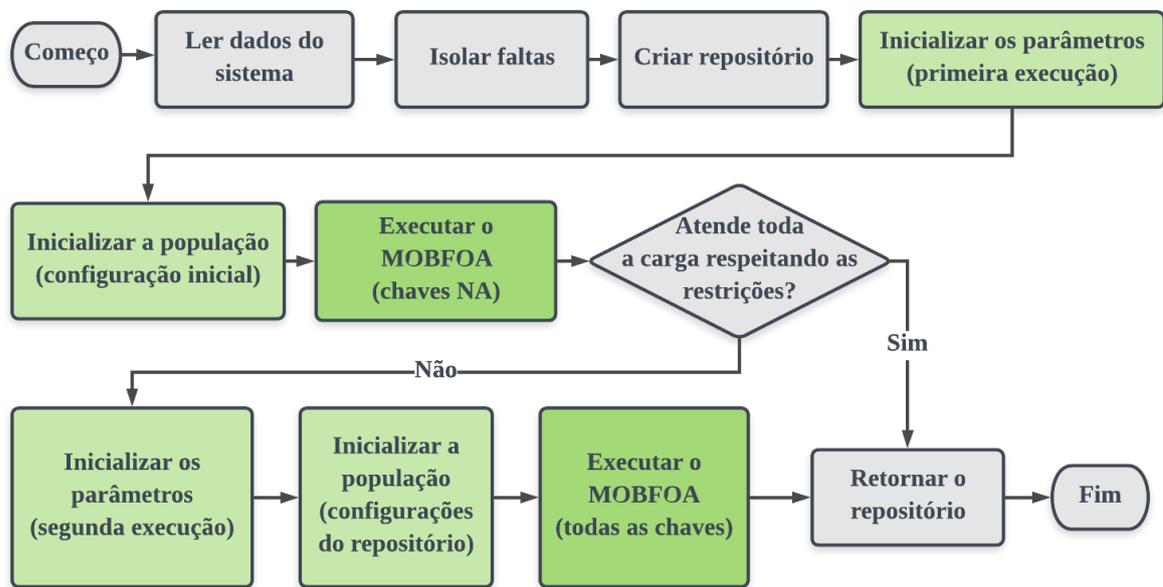


Figura 9 – Fluxograma do método proposto

Na próxima seção é apresentado o modelo da rede elétrica implementado com base em dados de entrada, como potência demandada pelos consumidores e resistência dos condutores. Em seguida são detalhadas as rotinas para a busca e isolamento das falhas e para o cálculo do fluxo de potência no modelo. É definido ainda o modelo matemático utilizado para o problema de restauração de RDEE estudado e, por fim, é abordada em detalhes a versão proposta para o algoritmo MOBFOA e a sua forma de aplicação para a resolução do problema.

5.1 Modelagem computacional do sistema

O grafo que modela o sistema é representado por uma matriz de adjacências $N_{barras} \times N_{barras}$, onde N_{barras} é o número de barramentos. Cada elemento da matriz armazena as informações sobre a conexão entre o barramento i e o barramento j , sendo

$i, j = 0, 1, \dots, N_{barras} - 1$. Também é implementado um vetor de N_{barras} posições para armazenar as informações de cada um dos barramentos e um vetor de N_{chaves} para armazenar o estado de cada chave.

O modelo representa um sistema monofásico e com cargas de potência constante. A figura 10 mostra o modelo gráfico relacionando as informações de entrada sobre cada uma das partes do sistema.

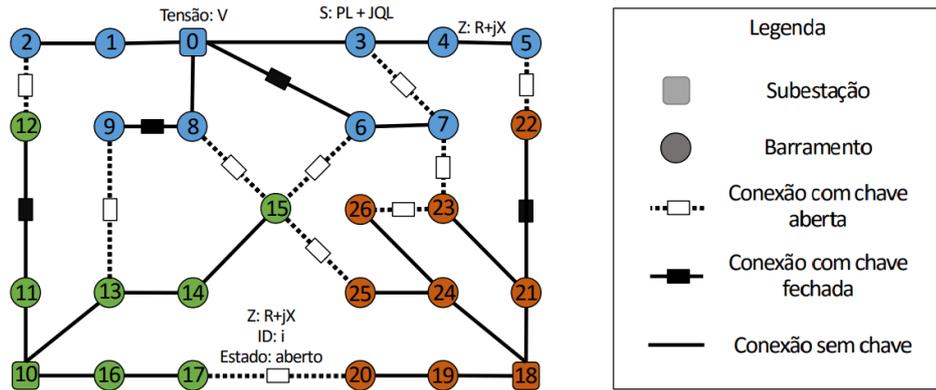


Figura 10 – Modelagem de uma rede de distribuição em grafo (MAZZETTO, 2016)

As informações referente aos barramentos são dados em valores de potência complexa demandada. Os dados referente às conexões entre os barramentos são a impedância do condutor, máximo de corrente suportada e um indicador id que indica caso o condutor seja uma conexão normal, ou um valor inteiro na faixa $id = 0, 1, 2, \dots, N_{chaves}$ que representa o índice da chave, caso ela exista.

As informações do sistema que são relevantes para o procedimento de recuperação são dispostas em um arquivo de entrada. Os dados são formatados de maneira que o programa possa lê-los e organizá-los nas estrutura de dados. Na primeira linha do arquivo estão as informações referentes aos limites inferior $V^{(min)}$ e superior $V^{(max)}$ de tensão para as barras. Na segunda linha estão as quantidades de barramentos N_{bar} e chaves N_{chaves} no sistema. Esses valores são definidos como constantes no programa e são utilizados para alocar inicialmente os espaços de memória e filtrar os resultados obtidos nos cálculos do fluxo da rede. Também são informados, o valor de potência base S_{base} , o valor de tensão base V_{base} e o limiar ϵ para análise de convergência usado durante a rotina de fluxo de potência. As próximas N_{ramos} linhas, onde N_{ramos} é o número de conexões do sistema, contém as seguintes informações:

- Conexão entre duas barras i e j , expressa por um par (i, j)
- Identificação id da chave, caso presente na aresta;
- Estado da chave, caso presente na aresta;

- Impedância Z_{ij} da conexão, expressa em resistência e reatância;
- Potência S do barramento j , expressa em potência ativa e reativa;
- Limite $maxI$ de corrente no condutor.

5.1.1 Isolamento de Falhas

Sempre que uma falha é detectada no sistema, ela deve ser isolada, ou seja, é desconectada do sistema a parte da árvore que foi afetada, com seus barramentos e conexões. Esse processo é importante, pois no primeiro momento não se sabe ao certo o que ocasionou a falta. Por isso as falhas podem atrapalhar o processo de restabelecimento das demais cargas do sistema.

A rotina implementada para o isolamento das falhas é semelhante a proposta em (MAZZETTO, 2016). Uma busca em profundidade na rede é realizada, partindo de todas as raízes, ou subestações, até que o local com falha seja encontrado. Em seguida são modificadas todas as chaves adjacentes mais próximas para aberto. Essas chaves são sinalizadas para que não sejam alteradas no processo de restauração.

Uma rotina auxiliar também foi implementada para realizar alguns testes comparativos. Assim como em (LAMBERT-TORRES et al., 2009), é proposto um método que simplifica o processo de isolamento. Por meio da exclusão do ramo com falhas do sistema, considerando esse ramo como uma conexão aberta, essa estratégia torna o processo de isolamento mais rápido. Porém isso implica que falta tenha ocorrido necessariamente em um dos condutores e que ele não esteja em curto.

5.1.2 Fluxo de Potência

Para saber se obter as informações sobre as cargas atendidas em uma determinada configuração do sistema e se todas as restrições são respeitadas, deve-se fazer o cálculo do fluxo de potência. Como a solução proposta atende apenas redes radiais, um método simples para realizar esse cálculo é a soma das correntes.

O método da soma das correntes foi implementado como uma rotina iterativa que termina quando a diferença entre o valor de tensão calculado na fase i e o valor calculado na fase $i + 1$ é menor que o limiar de convergência ϵ , definido inicialmente.

O procedimento do algoritmo é o seguinte:

1. Considera-se a impedância (Z_{ij}) de cada condutor, composto por um valor de resistência e reatância; a Potência consumida (S_i), composto por potência ativa e reativa; e a Tensão da fonte de alimentação (V_{base}), também usada para normalizar os valores da análise.

2. Assume-se inicialmente a tensão referencial (da fonte de alimentação), e denota-se o seu valor normalizado em $1pu$. Posteriormente é atribuída para cada consumidor j o valor de tensão igual ao da referência, ou seja, $V_j = V_{base}$.
3. É feita uma varredura partindo dos nós finais, determinando os valores das correntes do sistema de acordo com a equação 5.1, onde I_k é cada uma das correntes que sai do nó j para os nós subsequentes. Se alguma corrente I_{ij} ultrapassar o limite $maxI$ no condutor em questão, é feita uma indicação de falta.

$$I_{ij} = \left(\frac{S_j}{V_j} \right)^* + \sum I_k \quad (5.1)$$

4. Partindo da subestação para as barras finais, são determinados os valores das tensões do sistema de acordo com a equação 5.2 e verificando se o valor V_j está dentro da faixa operacional e, caso não esteja, o processo termina e retorna um indicador de falta.

$$V_j = V_i - Z_{ij}I_{ij} \quad (5.2)$$

5. É realizado um teste de erro absoluto entre o valor anterior da tensão V_0 e o valor atual V_1 , para um limiar ϵ previamente especificado, como descrito na equação 5.3.

$$|V_i^0 - V_i^1| < \epsilon \quad (5.3)$$

6. Caso o teste falhar para algum valor de tensão se repete o procedimento a partir da etapa 3. Caso contrário, a rotina termina retornando um indicativo de sucesso.

5.2 Objetivos e Restrições

O modelo proposto para o problema de restauração de RDEE é definido na equação 5.4.

$$\begin{aligned}
 & \text{Minimizar } f_m(x), & m = 1, 2 \\
 & \text{sujeito a : } I_k(x) \leq \max I_k, & k = 1, 2, \dots, N_{ramos} \\
 & & V^{(min)} \leq V_j(x) \leq V^{(max)}, \quad j = 1, 2, \dots, N_{barras} \\
 & & 0 \leq x_i \leq 100, \quad i = 1, 2, \dots, N_{chaves} \\
 & & G(x) \text{ deve ser radial,}
 \end{aligned} \quad (5.4)$$

onde x é o conjunto de estados das chaves de manobra que representa uma configuração do sistema, $f_m(x)$ são as funções objetivos a serem minimizadas, $I_k(x)$ é o fluxo de corrente no ramo k para a configuração x , $\max I_k$ é a corrente máxima suportada pelo

ramo k , $V_j(x)$ é a tensão no barramento j para a configuração x , $V^{(min)}$ e $V^{(max)}$ marcam os limites operacionais de tensão, $G(x)$ é o grafo formado pela configuração x , N_{ramos} é a quantidade de ramos que ligam dois barramentos, N_{barras} é a quantidade de barramentos conectados, N_{chaves} é a quantidade de chaves de manobra disponíveis.

Para permitir a melhor movimentação dos indivíduos no espaço de decisão, cada variável x_i pode assumir valores reais entre 0 e 100. Durante a varredura do sistema, uma chave i é considerada aberta caso $x_i \leq 50$ e fechada caso contrário.

Levando em conta essas definições, uma solução x é factível se garante que a estrutura da rede se mantenha em topologia radial, respeite os limites de operacionais de tensão nos barramentos e mantenha os limites de corrente nas conexões entre os barramentos.

As soluções ótimas devem minimizar as funções objetivo $f_m(x)$, que são:

- $f_1(x)$: Quantidade de carga não restaurada para a configuração x . Indica a porcentagem da potência total demandada que não é atendida. Esse total é a soma das demandas de todos os consumidores fora da região isolada.
- $f_2(x)$: Número de operações das chaves. É calculado pela comparação entre o vetor x de solução de estados das chaves com o vetor x_0 da configuração inicial após o isolamento.

5.3 MOBFOA

A versão do MOBFOA proposta nesse trabalho é um híbrido das metodologias propostas em (NIU et al., 2013) e (KAUR; KADAM, 2018). O foco da implementação foi em alterar as etapas de seleção e reprodução das bactérias de maneira que elas não dependam da saúde dos indivíduos, fazendo com que o método possa ser usado adaptado para a versão multiobjetivo mais facilmente. Além disso, a maneira em que as bactérias caminham pelo espaço foi alterada para uma versão mais adaptativa e focada na interação com o restante da população, como vai ser discutida em mais detalhes nessa seção. A figura 11 mostra os ajustes realizados no método em relação ao algoritmo tradicional.

O primeiro passo foi a implementação de um repositório para armazenar as soluções não-dominadas, ou seja, as soluções do conjunto Pareto-ótimo. A leitura dos elementos contidos nesse repositório a cada geração indicam a qualidade da solução obtida até o momento, e pode ser usada para alterar as características da busca e melhorar a convergência e a diversidade da solução final. Para isso, é utilizado o sistema de coordenadas chamado de PCCS, assim como em (HU; YEN, 2015) e em (NIU et al., 2013), para mapear as soluções e calcular o valor de entropia do conjunto, que é utilizado na adaptação da varredura feita pelas bactérias.

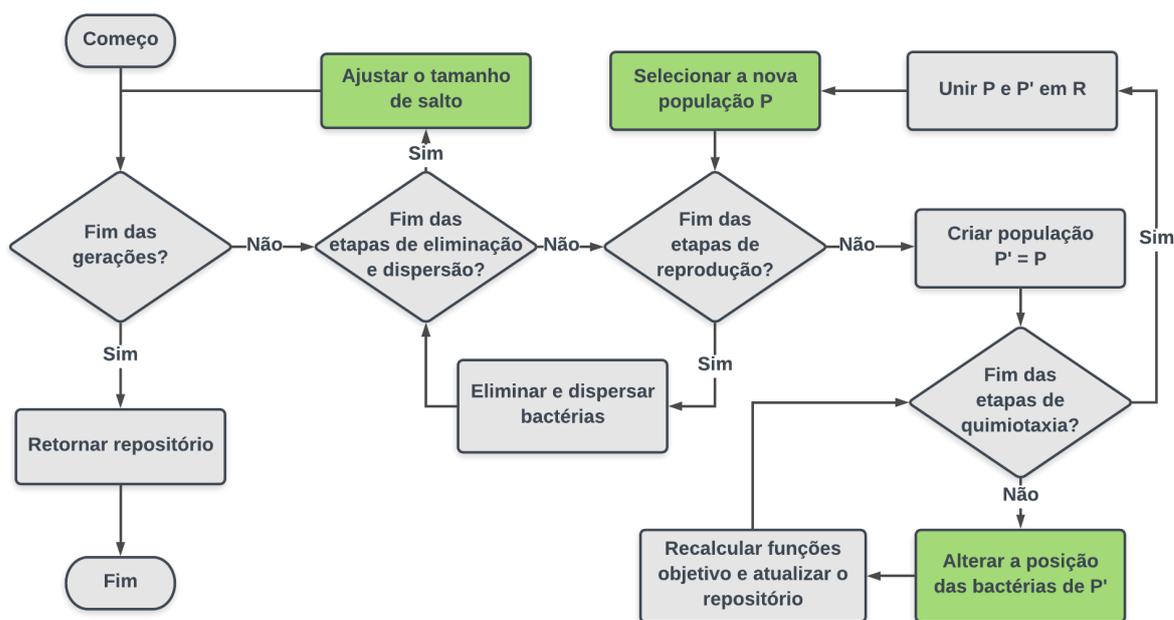


Figura 11 – Fluxograma das etapas do MOBFOA

Como discutido na seção 4.2.2, o tamanho do salto define a movimentação das bactérias durante a varredura do espaço de busca. O ajuste desse tamanho de salto pode impactar muito a convergência do método. Um valor de tamanho de salto C fixo e grande faz com que a convergência seja atingida mais rapidamente, porém pode fazer com que os indivíduos saltem "sobre" os pontos ótimos, trazendo instabilidade na busca. De outra forma, um valor C fixo e pequeno, melhora a estabilidade da busca, mas tem uma convergência mais lenta, podendo fazer com que o método termine antes de encontrar uma solução ótima, ou ainda pode fazer com que a solução convirja para um ótimo local. Realizando o ajuste de C através das gerações pode-se aproveitar os melhores benefícios durante a varredura, mantendo a estabilidade, melhorando a velocidade da convergência e evitando ótimos locais.

O mesmo problema discutido na implementação da versão do MOPSO proposta em (HU; YEN, 2015) foi considerado nesse trabalho. As bactérias não devem ser atraídas para um único ponto, mas sim para a região Pareto-ótima. Dessa forma, as operações de *swarm*, abordadas no capítulo anterior, devem ser adaptadas para que as interações de atração e repelimento respeitem essa característica. Na versão proposta essa etapa foi simplificada e integrada na etapa de quimiotaxia, como o proposto em (PANIGRAHI et al., 2011), onde cada bactéria vai escolher a direção θ para se mover atraída por uma outra bactéria dominante.

A etapa de seleção e reprodução é feita de forma semelhante à proposta em (DEB et al., 2002) e em (KAUR; KADAM, 2018). As melhores bactérias são escolhidas por meio da NDS e, como critério de desempate, é usada a ordenação por CD. A figura 12 mostra

como a construção da nova população é feita. Na seção 5.3.1 esse procedimento é descrito em detalhes.

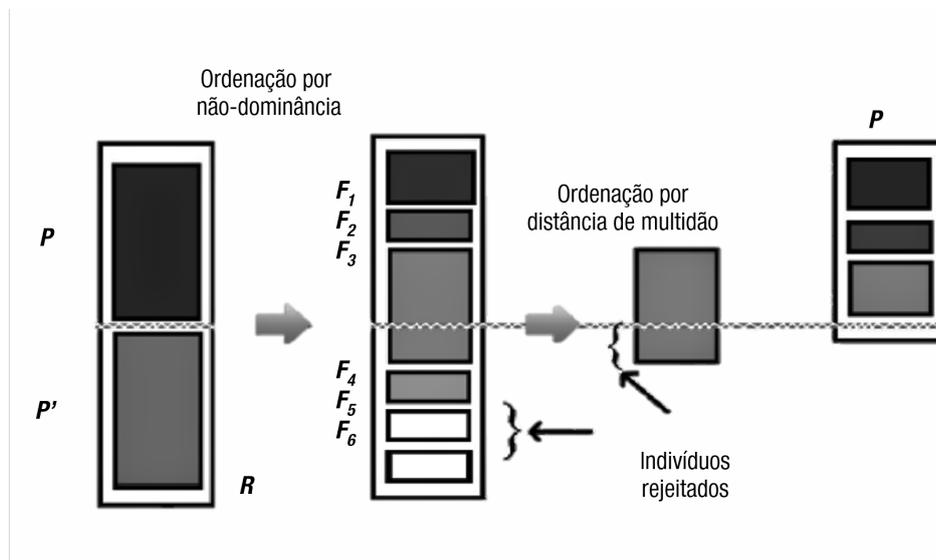


Figura 12 – Processo de seleção da nova população (KAUR; KADAM, 2018)

5.3.1 Etapas do MOBFOA

Dado um MOOP com M objetivos, as etapas do algoritmo e as rotinas citadas anteriormente são apresentadas em detalhes a seguir.

1. O número de etapas N_{ed} e a probabilidade p_{ed} de eliminação e dispersão, o número de etapas de reprodução N_{rep} , o número de etapas de quimiotaxia N_{ch} , o tamanho do salto C , os operadores de ajuste do tamanho do salto λ e μ e o número de gerações $maxgen$ são inicializados. Uma população P de tamanho N_{pop} é criada e os indivíduos são espalhados pelo espaço de busca aleatoriamente ou seguindo uma ordem pré-estabelecida e um repositório A é criado para armazenar as soluções Pareto-ótimas.
2. É realizada a avaliação de cada indivíduo para cada um dos M objetivos e os indivíduos que apresentam soluções não-dominadas são inseridos em A .
3. As soluções em A são mapeado no PCCS. Cada solução recebe um *label* L que representa cada solução na matriz PCCS e é um número inteiro entre 1 e K , calculado de acordo com a equação 5.5, onde K é a quantidade de soluções em A , M é o número de objetivos, $n = 1, 2, \dots, K$ é o indicador de cada solução, $m = 1, 2, \dots, M$ é o indicador de cada função objetivo, $f_{k,m}$ é o valor da solução k para o objetivo m , f_m^{max} é o maior valor encontrado para o objetivo m , f_m^{min} é o menor valor encontrado para o objetivo m e $\lceil x \rceil$ é um operador de arredondamento que retorna o menor

valor inteiro que não é menor que x . O valor é arredondado para cima e quando o valor é mudado para 1, para evitar divisão por zero.

$$L_{k,m} = \left\lceil K \frac{f_{k,m} - f_m^{min}}{f_m^{max} - f_m^{min}} \right\rceil \quad (5.5)$$

4. Caso, durante a atualização de A , o tamanho do repositório ultrapasse o valor máximo S , a solução com maior densidade é eliminada. A densidade de uma solução é a soma das distâncias entre ela e cada uma das demais soluções. A distância DCP (Distância de Célula Paralela) entre duas soluções P_i e P_j no PCCS é dada pela equação 5.6 e a densidade D de uma solução P_i é dada pela equação 5.7.

$$DCP(P_i, P_j) = \begin{cases} \sum_{m=1}^M |L_{i,m} - L_{j,m}|, & \text{se } \exists m \mid L_{i,m} \neq L_{j,m} \\ 0.5, & \text{caso contrário} \end{cases} \quad (5.6)$$

$$D(P_i) = \sum_{j=1}^K \frac{1}{DCP(P_i, P_j)^2} \quad (5.7)$$

5. A cada geração é calculada a entropia da população, que mede a uniformidade da aproximação da fronteira de Pareto. O cálculo da entropia E para a geração t é dado pela equação 5.8, onde $Cell_{k,m}(t)$ é a quantidade de soluções na geração t com o label $L_{k,m}$ na k -ésima linha e m -ésima coluna do PCCS.

$$E(t) = - \sum_{k=1}^K \sum_{m=1}^M \frac{Cell_{k,m}(t)}{KM} \log \frac{Cell_{k,m}(t)}{KM} \quad (5.8)$$

6. O ajuste no tamanho do salto das bactérias é feito com o diferencial $\Delta E(t) = E(t) - E(t-1)$ de acordo com o estado evolucionário da população, que pode ser estado de convergência, estado de diversidade e estado de estagnação. O limiar para o estado de convergência é dado por $\delta_C = \frac{2}{K} \log 2$ e o limiar para o estado de estagnação é dado por $\delta_E = \frac{2}{N_{pop}M} \log 2$, onde N_{pop} é o tamanho máximo do arquivo.
7. Dados os limiares, o ajuste do tamanho do salto C na geração t é dado pela equação 5.9, onde λ e μ são os ajustes e $K(t)$ é a quantidade de soluções em A na geração t .

$$C_t = \begin{cases} C_{t-1} - \lambda(1 + |\Delta E(t)|), & \text{se } |\Delta E(t)| > \delta_C \text{ ou } |K(t) - K(t-1)| > 0 \\ C_{t-1} + \mu|\Delta E(t)|, & \text{se } \delta_C < |\Delta E(t)| < \delta_E \text{ e } K(t) = K(t-1) \\ C_{t-1}, & \text{se } |\Delta E(t)| > \delta_E \text{ e } K(t) = K(t-1) \end{cases} \quad (5.9)$$

8. Para cada uma das N_{ed} etapas de eliminação e dispersão é realizado o processo de reprodução.

9. Para cada uma das N_{rep} etapas de reprodução é realizado o processo de quimiotaxia. É gerada uma nova população P' copiando os valores da população P e para cada uma das N_{ch} etapas de quimiotaxia, a população P' é atualizada recalculando a posição de cada indivíduo.
10. Para cada indivíduo i , sendo $i = 1, 2, \dots, N_{pop}$, um vetor $\Delta(i)$, que representa a direção do salto de cada bactéria, é gerado e um indivíduo aleatório de índice r é selecionado. Se a r -ésima bactéria domina a i -ésima, o valor de $\Delta(i)$ é dado pela equação 5.10, onde $\theta^i(ch, r, ed)$ é a posição da i -ésima bactéria na ch -ésima etapa de quimiotaxia, rp -ésima etapa de reprodução e ed -ésima etapa de eliminação e dispersão. Senão $\Delta(i)$ recebe valores aleatórios no intervalo $[-1, 1]$ para cada uma das N_{var} posições.

$$\Delta(i) = \theta^r(ch, rp, ed) - \theta^i(ch, rp, ed) \quad (5.10)$$

11. Com a direção do salto definida, a posição de cada bactéria é atualizada com a equação 5.11.

$$\theta^i(ch + 1, rp, ed) = \theta^i(ch, rp, ed) + C_t \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (5.11)$$

12. Em seguida todas as bactérias são avaliadas para todos os objetivos e o repositório A é atualizado com o mesmo procedimento feito anteriormente.
13. Terminado o processo de quimiotaxia, uma nova população é gerada por meio da ordenação por não-dominância. Para isso, a população P é mesclada com a população P' em um conjunto R . Todas as soluções que formam a fronteira de Pareto, consideradas soluções de *rank* 1, são selecionadas. Caso o número de soluções encontradas for menor que N_{pop} , elas são adicionadas na nova população P e removidas de R . Uma nova fronteira de Pareto, ou seja, a fronteira de *rank* 2, é selecionada de R . Caso o número de membros dessa fronteira for menor que $N_{pop} - |P|$, onde $|P|$ é o tamanho do conjunto P , eles também são adicionados em P e removidos de R , e assim por diante. A primeira parte da figura 12 ilustra esse procedimento.
14. Se alguma das fronteiras não puder ser adicionada em P , ou seja, for maior que o espaço disponível para que P tenha tamanho N_{pop} , é necessário realizar a ordenação por distância de multidão dos elementos dessa fronteira. A distância de multidão de uma solução é a distância média entre suas duas soluções vizinhas. É necessária a ordenação dos elementos com base em cada um dos objetivos e atribuição de

distância infinita às soluções com valor f_m^{min} e f_m^{max} . Cada elemento k da fronteira F em questão tem a distância $F[k]_{dist}$, dada pela equação equação 5.12.

$$F[k]_{dist} = F[k]_{dist} + \frac{f_{k+1,m} - f_{k-1,m}}{f_m^{max} - f_m^{min}} \quad (5.12)$$

Para garantir a diversidade na fronteira de Pareto, as soluções com os maiores valores são adicionadas em P , até que $|P| = N_{pop}$. A parte final da figura 12 mostra essa operação.

15. Terminadas as etapas de reprodução, os indivíduos são eliminados e dispersos no espaço de busca com um probabilidade p_{ed} .
16. Terminadas as etapas de eliminação e dispersão, é realizada um novo ajuste no tamanho do salto.
17. Terminadas as gerações, tem-se o repositório A com a melhor aproximação da fronteira de Pareto encontrada.

5.4 Aplicação do Algoritmo

De modo a tornar a execução do algoritmo mais rápida, uma estratégia de aplicação semelhante da proposta em (LAMBERT-TORRES et al., 2009) é implementada. O objetivo é tornar a execução do protótipo mais rápida.

O método é aplicado inicialmente ao conjunto de chaves NA (Normalmente Abertas). Caso alguma configuração dessas chaves for suficiente para atender todas as cargas respeitando todas as restrições, o programa finaliza retornando qual ou quais chaves NA devem ser fechadas.

Após a primeira execução é verificado se alguma das soluções ótimas encontradas viola o limite de corrente em algum condutor. Se isso acontece, pode existir uma chave NF (Normalmente Fechada) que elimina a sobrecarga ocorrida. Além disso, pode existir uma outra chave NA que, após a abertura dessa chave NF, pode ser fechada para atender o restante das cargas. Dessa forma, uma segunda execução do algoritmo é feita considerando todas as chaves do sistema.

Mesmo que não tenham ocorridas sobrecargas durante a primeira execução também é verificado se alguma solução restabelece todos os barramentos fora da área de isolamento. Caso isso não aconteça, a nova execução também é realizada para que a busca continue. Vale ressaltar que pode não existir uma solução que restabeleça 100% da carga demandada e a segunda execução só é realizada para garantir que essa possível solução ótima apenas não tenha sido encontrada na primeira.

Para cada uma das execuções, é feita uma inicialização dos indivíduos para otimizar a busca pelas melhores configurações. Na primeira execução, a população é inicializada com o valor da configuração inicial das chaves após o isolamento das faltas. Na segunda execução, caso seja necessária, a população é iniciada com as melhores combinações encontradas na execução anterior.

6 Resultados

A avaliação da eficácia da metodologia proposta para a resolução do problema de restauração de RDEE foi realizada por meio da aplicação do método em dois sistemas com 70 e 201 barramentos. Para cada um dos sistemas alguns casos de contingência com uma e mais falhas foram testados.

Para o sistema de 70 barramentos, por ser um sistema menor, foram ilustrados os casos de falhas, o processo de isolamento e as algumas das melhores configurações de restabelecimento ótimas encontradas. Para esse sistema são analisados três exemplos de conjuntos de falhas.

Para o sistema de 201 barramentos, dois conjuntos de falhas são testados e ilustrados. Em seguida, vários experimentos são feitos e as soluções são apresentadas na forma de tabela e comparadas com as soluções encontradas em (LAMBERT-TORRES et al., 2009).

Na próxima seção os parâmetros escolhidos para a execução da versão implementada do MOBFOA são apresentados. Nas duas seções seguintes são apresentados os resultados obtidos pelos métodos para os sistemas.

6.1 Ajuste de Parâmetros

Os parâmetros usados para os testes foram escolhidos com base em trabalhos presentes na literatura e por meio da observação da convergência do método. Como discutido na seção 5.4, o algoritmo é executado duas vezes. Cada uma das execuções tem uma configuração inicial de parâmetros diferentes, pois envolvem um número diferente de variáveis. A tabela 1 mostra o valor escolhido para cada um dos parâmetros nas duas execuções.

Tabela 1 – Parâmetros do MOBFOA

Parâmetros	Execução 1	Execução 2
N_{ed}	2	2
p_{ed}	0,25	0,25
N_{rep}	4	4
N_{ch}	60	120
C	50	50
λ	2	2
μ	10	10
$maxgen$	5	10

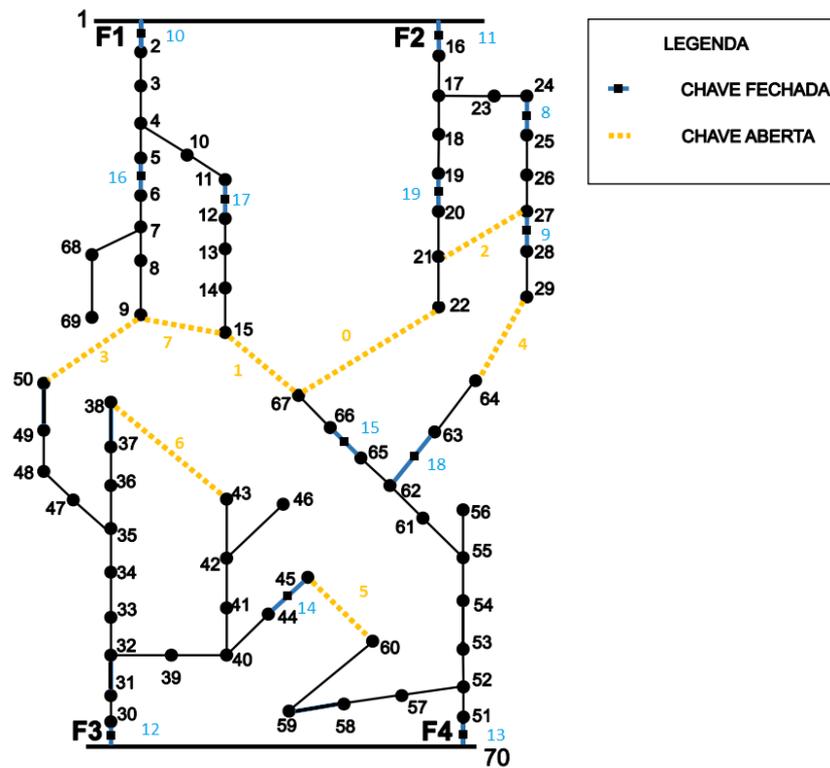


Figura 13 – Configuração do sistema de 70 barramentos. Adaptado de (MAZZETTO, 2016)

6.2 Sistema com 70 barramentos

O primeiro sistema utilizado para os testes foi adaptado de (DAS, 2006), de onde foram retiradas as informações de entrada para esse sistema, e é composto por 70 barramentos, 2 subestações, 9 chaves NA e 11 chaves NF. Os valores de tensão base é configurado como 11 kV e o valor de potência base é de 1 MVA . Os valores de tensão nos barramentos devem estar na faixa de 0,8 pu a 1,05 pu . Os limites de corrente são dados para cada condutor.

O sistema foi escolhido para ilustrar o funcionamento do algoritmo, devido a facilidade na sua representação. A figura 13 mostra a configuração inicial do sistema.

6.2.1 Única falta

Para o primeiro experimento, foi considerada uma falha na conexão que liga os barramentos 25 e 26, como mostra a figura 14. Para o isolamento do sistema as chaves NF 8 e 9 são abertas e marcadas para que não sejam utilizadas na etapa de restauração, assim como a chave NA 2. A figura 15 mostra a configuração do sistema após o isolamento.

Executando uma análise por força bruta de todas as combinações de chaves pos-

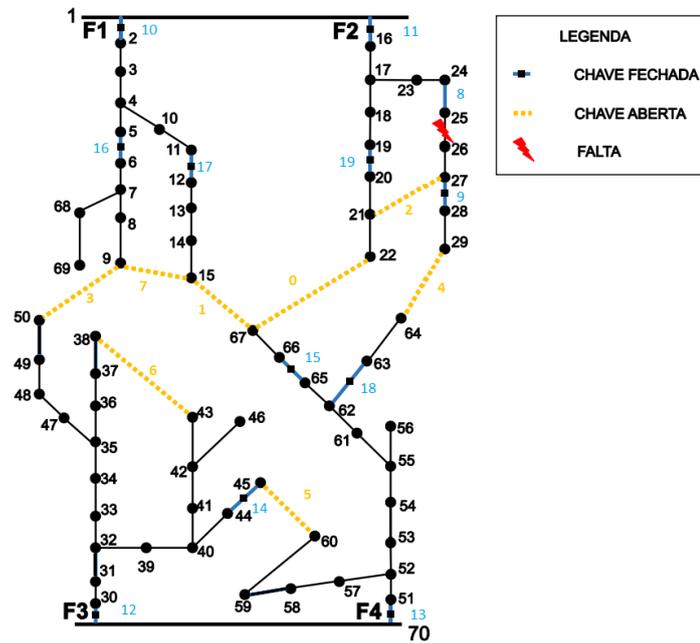


Figura 14 – Sistema de 70 barramentos com falha em 25-26.

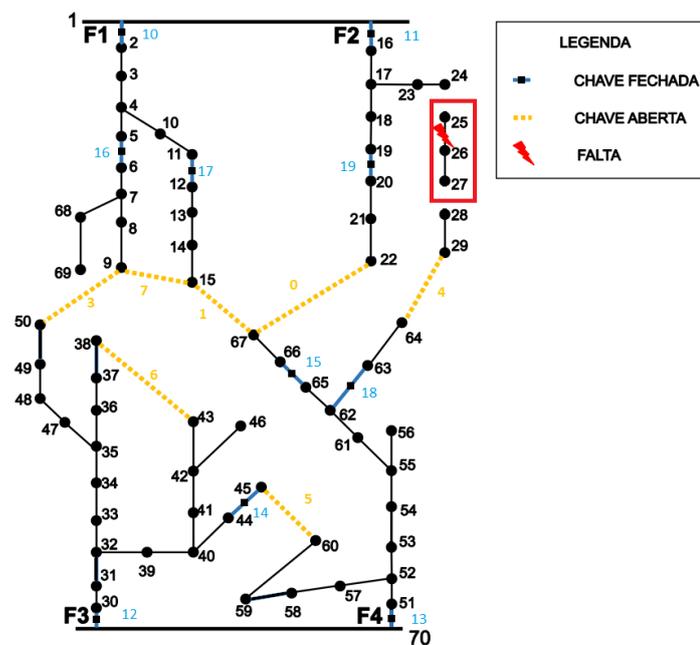


Figura 15 – Sistema de 70 barramentos após isolamento da falha em 25-26.

Tabela 2 – Soluções Pareto-ótimas para sistema de 70 barramentos com falta em 25-26

Linha(s) Defeituosa(s)	NA	NF	P (%)
25 - 26	29 - 64	Nenhuma	100,00
	Nenhuma	Nenhuma	94,83

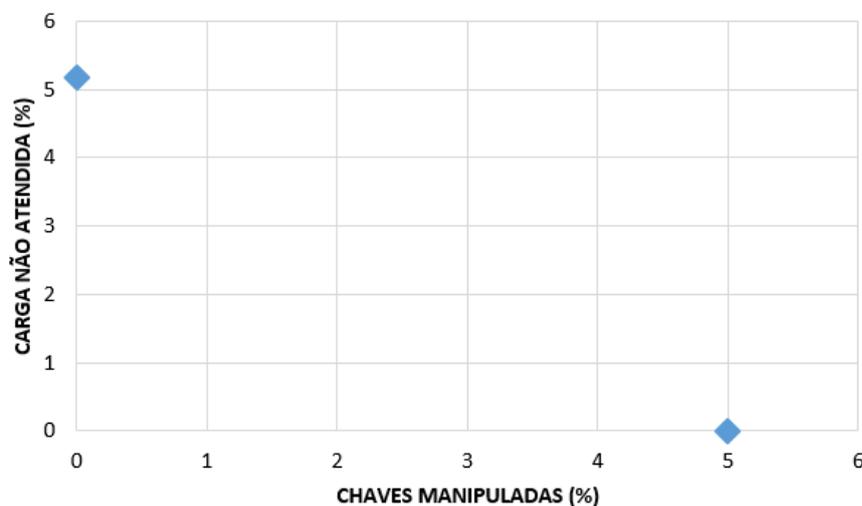


Figura 16 – Fronteira de Pareto para o sistema de 70 barramentos com falta em 25-26.

síveis, foram encontradas as duas soluções Pareto-ótimas, presentes na tabela 2. Como é mostrado na tabela, a primeira solução atende toda a demanda realizando manobras em 5% das chaves, ou seja, fechando a chave NA 4 para atender os barramentos 28 e 29. A segunda não realiza nenhuma manobra nas chaves disponíveis e deixa de atender cerca de 5% da potência demandada, ou seja, as cargas dos barramentos 28 e 29, que foram desconectados do sistema depois do isolamento. Vale ressaltar que a demanda da região da falha é subtraída da demanda total durante o isolamento.

As soluções encontradas pelo algoritmo estudado são mostradas na figura 16 por meio da curva de Pareto. O método encontrou todas as soluções ótimas em 35 segundos, usando 2400 iterações.

No segundo experimento, foi considerada uma falta entre os barramentos 3 e 4. As chaves NF 10, 16 e 17 são utilizadas no isolamento e, por isso, são marcadas para que não sejam aletradas na etapa de otimização. A figura 17 mostra a configuração do sistema após o isolamento.

Novamente, ao executar uma análise por força bruta, as soluções ótimas da tabela 3 foram encontradas. Na primeira solução, fechando as chaves NA 1 e 7, que são 10% das chaves disponíveis, toda demanda é atendida. Na segunda solução, apenas a chave NA 1 é fechada, ou seja, 5% das chaves, e apenas 5% da carga não é atendida. A última solução é a configuração após o isolamento. Nela, sem realizar nenhuma mudança de chave, cerca de 89% da demanda é atendida, ou seja, cerca de 11% da carga demandada está desconectada

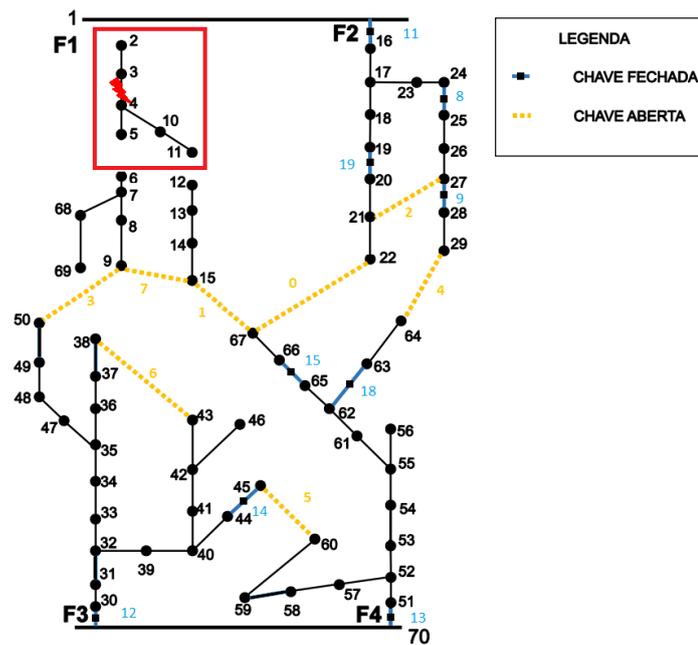


Figura 17 – Sistema de 70 barramentos com falta em 3-4.

Tabela 3 – Soluções Pareto-ótimas para sistema de 70 barramentos com falta em 3-4

Linha(s) Defeituosa(s)	NA	NF	P (%)
3 - 4	67 - 15	Nenhuma	100,00
	9 - 50	Nenhuma	95,00
	Nenhuma	Nenhuma	89,34

do sistema. Vale lembrar que todas as configurações ótimas respeitam todas as restrições operacionais do sistema.

O algoritmo também encontra todas as soluções ótimas para esse caso, levando cerca de 75 segundos e 2400 iterações. As soluções são mostradas na fronteira de Pareto da figura 18.

6.2.2 Duas faltas

A figura 19 mostra o mesmo sistema após a ocorrência e isolamento das duas faltas tomadas como exemplo na seção anterior, dessa vez, simultaneamente. As chaves envolvidas na rotina de isolamento e impossibilitadas de serem utilizadas na restauração são as chaves NF 8, 9, 10, 16 e 17 e a chave NA 2.

Depois da análise por força bruta, foram encontradas as soluções mostradas na tabela 4. Nesse caso, realizando apenas o fechamento de chaves NA não foi possível restabelecer todos os barramentos sem violar o limite de corrente em algum condutor. Por isso, em duas soluções a chave NF 14 precisou ser aberta e em outros dois a chave NF 15

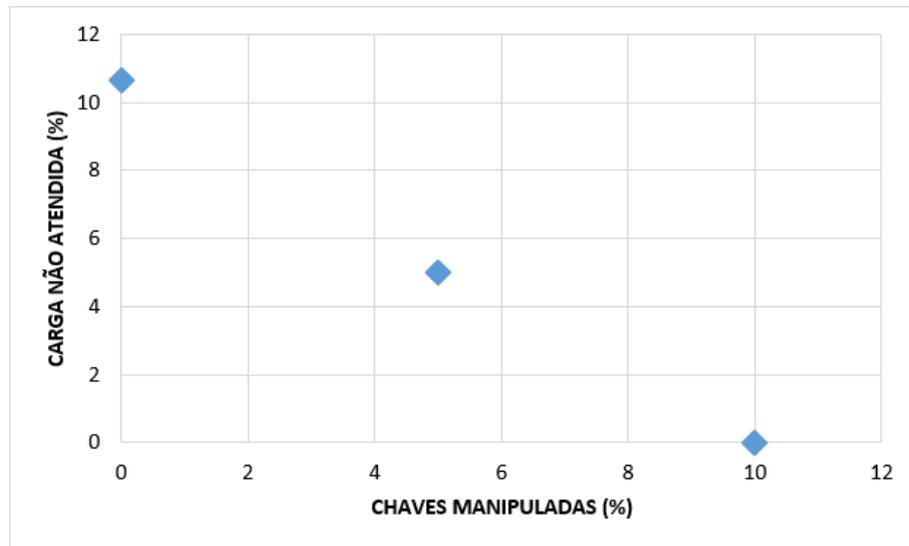


Figura 18 – Fronteira de Pareto para o sistema de 70 barramentos com falta em 3-4.

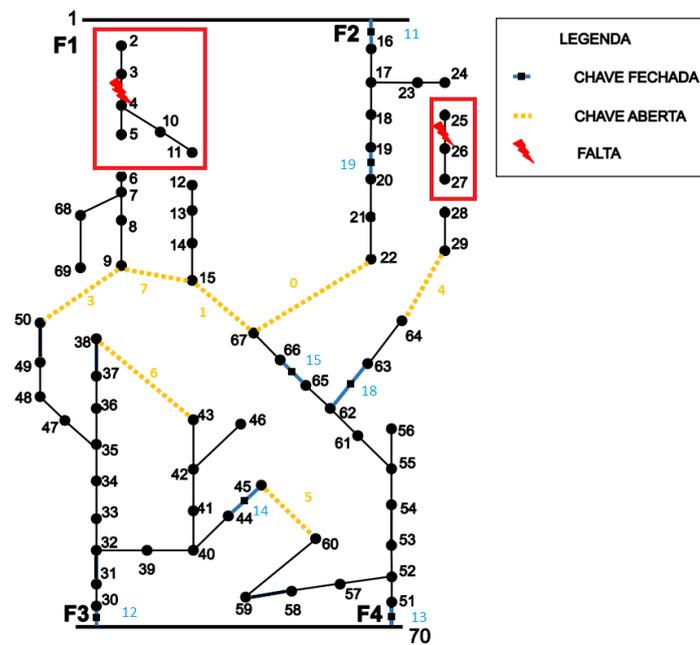


Figura 19 – Sistema de 70 barramentos com faltas em 3-4 e 25-26.

Tabela 4 – Soluções Pareto-ótimas para a restauração do sistema de 70 barramentos com faltas em 3-4 e 25-26

Linha(s) Defeituosa(s)	NA	NF	P (%)
3 - 4 25 - 26	9 - 50 29 - 64 45 - 60	44 - 45	100,00
	9 - 15 22 - 67 67 - 15	65 - 66	100,00
	9 - 50 29 - 64	65 - 66	100,00
	22 - 67 67 - 15 29 - 64 9 - 15	65 - 66	100,00
	9 - 15 29 - 64	44 - 45	97,56
	9 - 50	44 - 45	97,56
	67 - 15 9 - 50	Nenhuma	94,22
	67 - 15	Nenhuma	88,86
	Nenhuma	Nenhuma	82,79

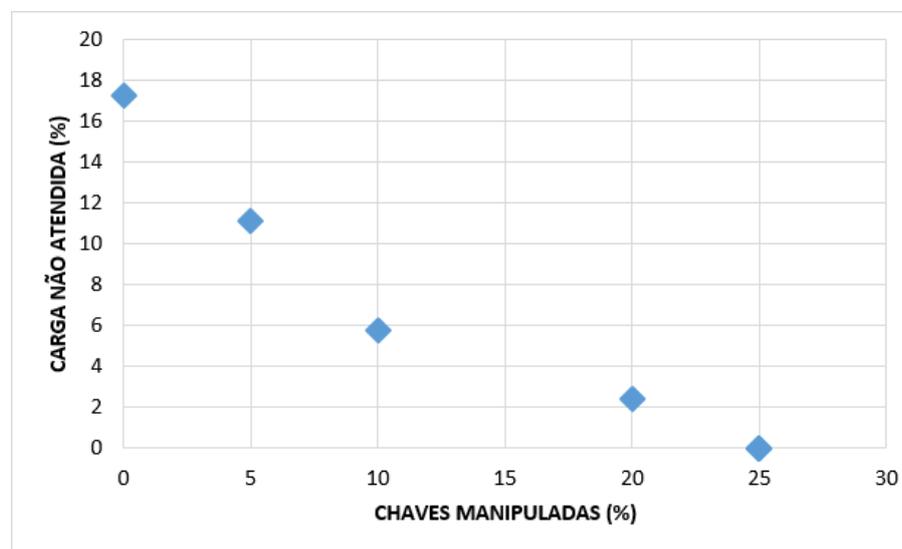


Figura 20 – Fronteira de Pareto para a restauração do sistema de 70 barramentos com faltas em 3-4 e 25-26.

foi alterada.

Na figura 20 a curva de Pareto das soluções encontradas pelo algoritmo proposto é exibida com as melhores soluções possíveis. As soluções foram encontradas em 633 segundos com 12000 iterações.

A figura 21 mostra a segunda das 5 soluções encontradas. Essa solução restabelece

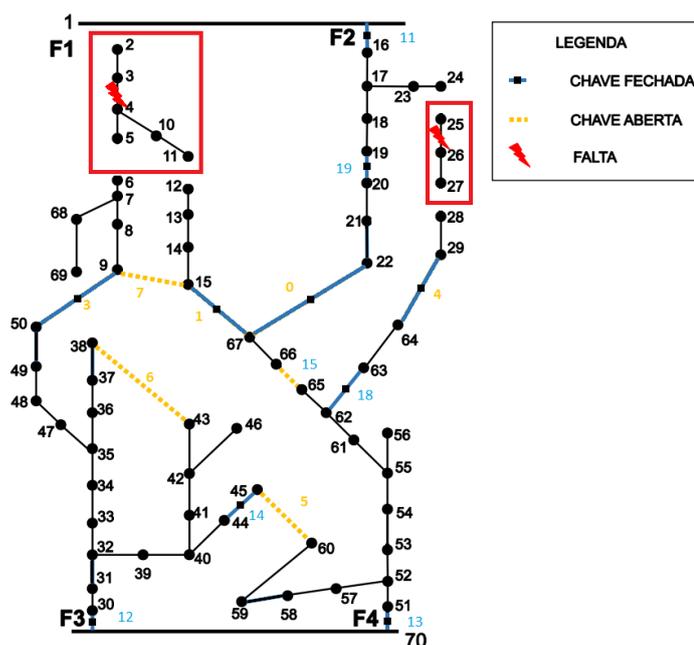


Figura 21 – Configuração ótima para o restabelecimento do sistema de 70 barramentos com faltas em 3-4 e 25-26.

toda a carga demandada.

6.3 Sistema 201b

O segundo sistema utilizado para os experimentos foi adaptado de (LAMBERT-TORRES et al., 2009) e tem 201 barramentos, uma subestação com três alimentadores, 39 chaves NA e 37 chaves NF. Os valor de tensão base é de 10 kV e o valor de potência base é 100 KVA . Esse sistema não apresenta os valores de impedância para os condutores, dessa forma, não é feito o cálculo de queda de tensão e, conseqüentemente, o valor de tensão para cada barramento é fixado em 1 pu . O limite de fluxo nos condutores é dado pela potência de máxima de 7983 kW . Dessa forma, para se adequar ao método foi considerado o limite de corrente de 79,83 A para cada condutor. A figura 22 mostra a configuração inicial do sistema.

6.3.1 Única falta

O primeiro experimentos considera uma falha entre os barramentos 71 e 79. Depois do isolamento a seção entre os barramentos 76 e 89 ficam sem fornecimento. A figura 23 mostra a configuração do sistema após o isolamento.

Após a execução do método foram encontradas as soluções da tabela 5, que também são mostradas na fronteira de Pareto da figura 24. O método usou 2400 iterações e levou

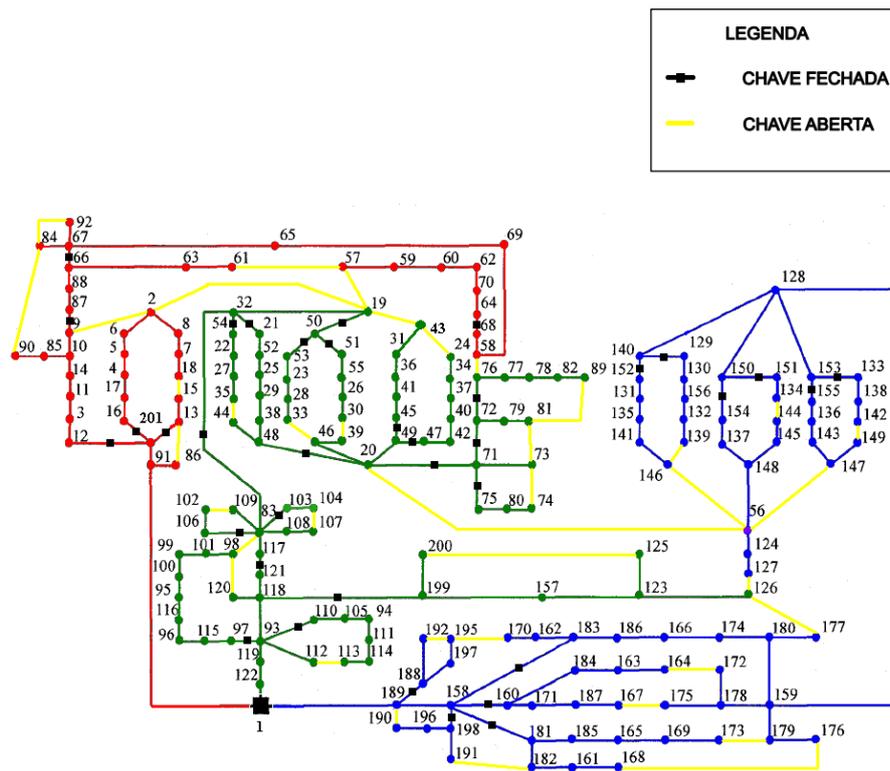


Figura 22 – Sistema de 201 barramentos.

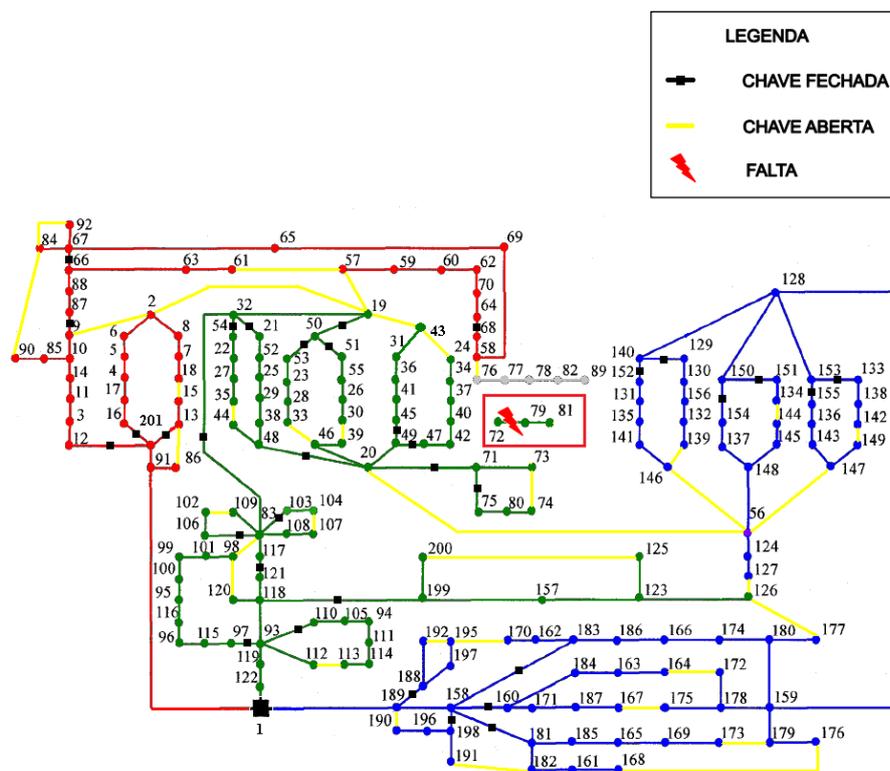


Figura 23 – Sistema de 201 barramentos com falha em 72-79. Adaptado de (LAMBERT-TORRES et al., 2009)

Tabela 5 – Soluções Pareto-ótimas para o sistema de 201 barramentos com falta em 72-79.

Linha(s) Defeituosa(s)	NA	NF	P (%)
72 - 79	58 - 76	Nenhuma	100,00
	Nenhuma	Nenhuma	97,77

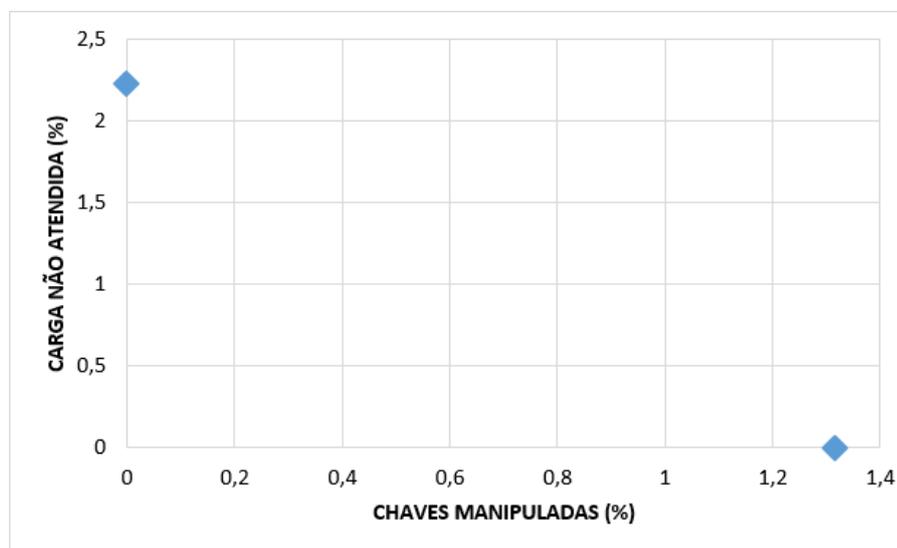


Figura 24 – Fronteira de Pareto para a restauração do sistema de 201 barramentos com falta em 72-79.

cerca de 90 segundos para entregar todas as respostas.

6.3.2 Duas faltas

Para realizar um experimento mais complexo foram consideradas uma falha entre os barramentos 9 e 10 e outra entre os barramentos 20. As falhas juntas desconectam cerca de 30% da demanda de carga da rede. A figura 25 mostra a configuração da rede após o isolamento das duas faltas.

A primeira execução do algoritmo também não conseguiu restabelecer toda a carga nesse caso, apesar das melhores soluções encontradas não violarem nenhum limite operacional. Portanto, uma nova execução do algoritmo foi executada e, para esse caso, uma solução que atende toda a demanda pode ser encontrada. A tabela 6 mostra todas as soluções encontradas e a fronteira de Pareto para essas soluções é mostrada na figura 26.

A primeira configuração, que restabelece toda a demanda de carga para esse caso, é mostrada na 27.

6.3.3 Comparação entre métodos

Para verificar a qualidade das respostas encontradas para o sistema de 201 barramentos, foi feita a comparação com os resultados obtidos com a utilização da técnica

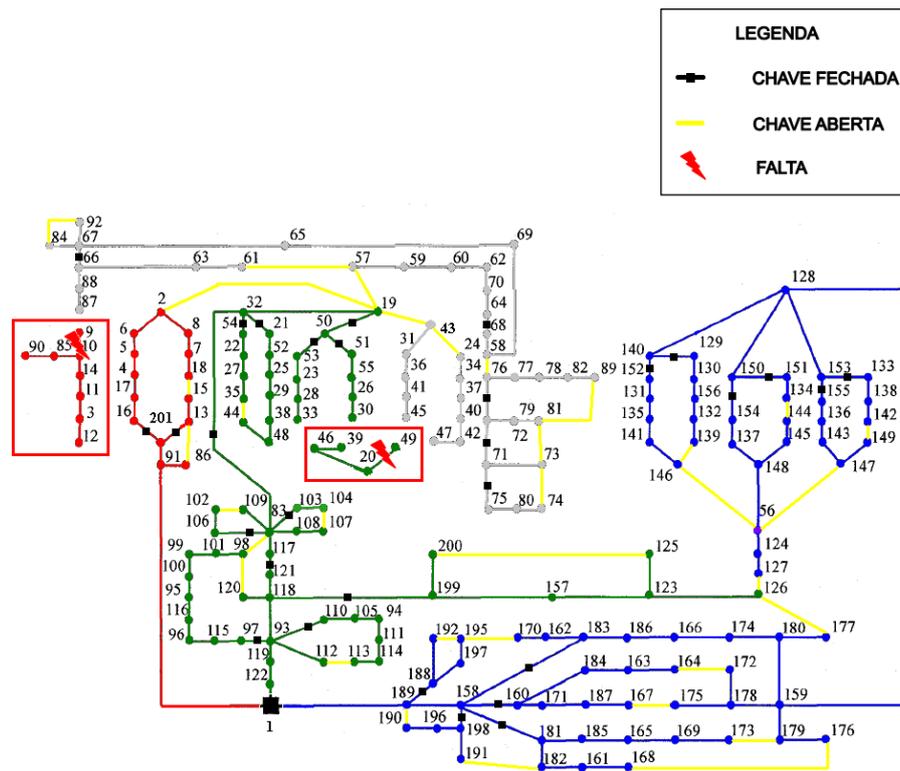


Figura 25 – Sistema de 201 barramentos com falhas em 9-10 e 20-49.

Tabela 6 – Soluções Pareto-ótimas para o sistema de 201 barramentos com falhas em 9-10 e 20-49.

Linha(s) Defeituosa(s)	NA	NF	P (%)
9 - 10 20 - 49	2 - 19 19 - 43 19 - 57 24 - 43 58 - 76	117 - 121	100,00
	19 - 43 19 - 57 24 - 43 58 - 76	83 - 106	99,02
	19 - 43 19 - 57 58 - 76	Nenhuma	95,83
	19 - 57 58 - 76	Nenhuma	93,14
	19 - 57	Nenhuma	85,48
	Nenhuma	Nenhuma	75,91

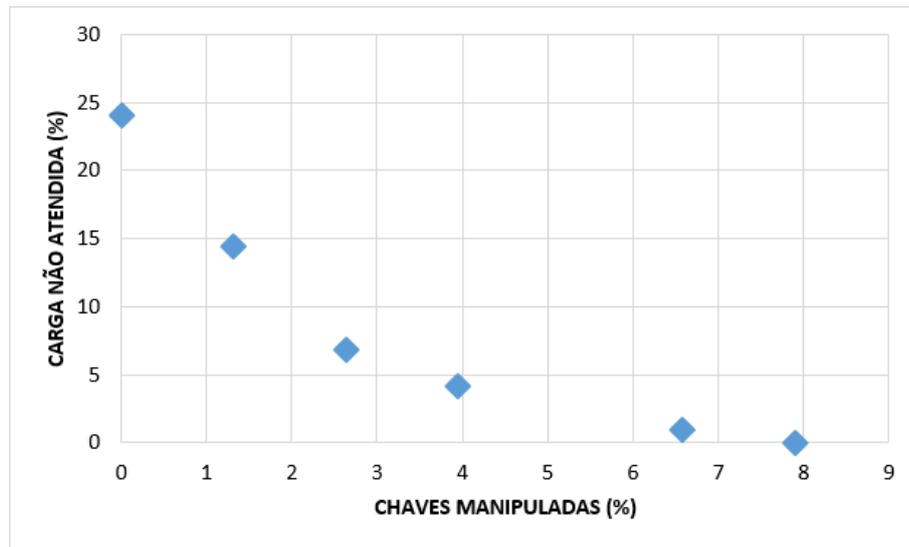


Figura 26 – Fronteira de Pareto para o sistema de 201 barramentos com faltas em 9-10 e 20-49.

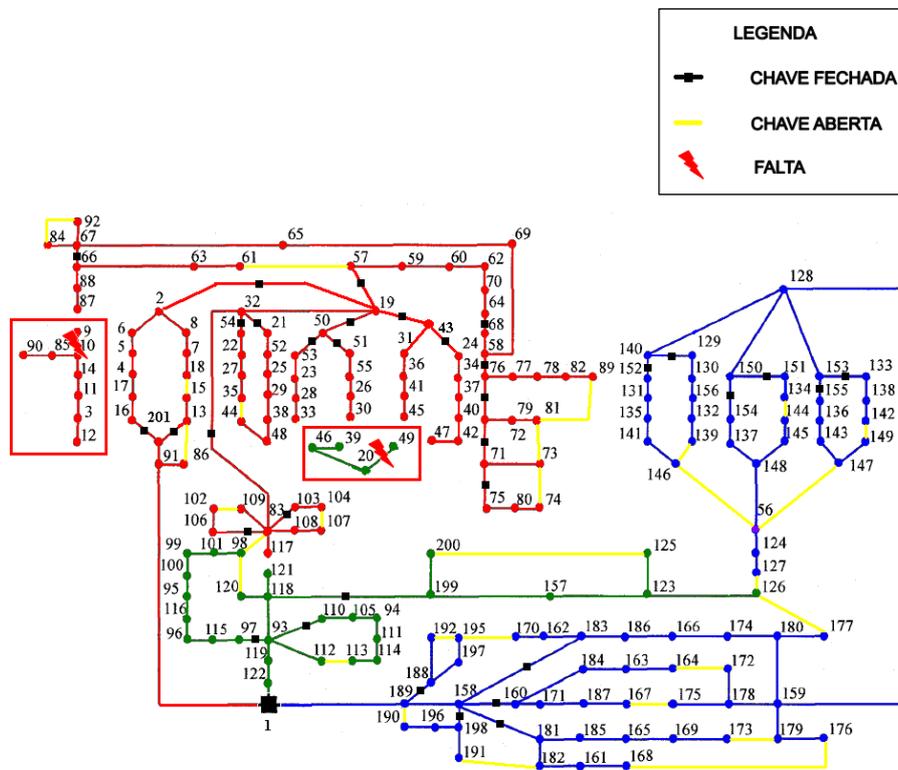


Figura 27 – Configuração ótima para o sistema de 201 barramentos com faltas em 9-10 e 20-49.

Tabela 7 – Comparação dos resultados dos métodos MOBFOA e PSO - Única solução

Linha(s) Defeituosa(s)	MOBFOA			PSO		
	NA	NF	P (%)	NA	NF	P (%)
1 - 91	2 - 19 20 - 56	20 - 48	100,00	2 - 19 20 - 56	20 - 48	100,00
	19 - 57 20 - 56	20 - 48	100,00			
	2 - 19	12 - 201	86,75			
	Nenhum	Nenhum	82,62			
76 - 77	81 - 89	Nenhuma	100,00	81 - 89	Nenhuma	100,00
	Nenhum	Nenhum	98,16			
93 - 110	112 - 113	Nenhuma	100,00	112 - 113	Nenhuma	100,00
	Nenhum	Nenhum	95,68			
128 - 140	56 - 146	Nenhuma	100,00	56 - 146	Nenhuma	100,00
	Nenhum	Nenhum	93,62			

PSO proposta em (LAMBERT-TORRES et al., 2009). O PSO usa apenas uma função objetivo e considera os demais objetivos por meio da ponderação. Porém, em execuções diferentes, o método encontra soluções diferentes igualmente ótimas. Assim, é possível realizar a comparação com um método multi-objetivo, como o proposto nesse trabalho.

Como discutido na seção 5.1.1, o isolamento das falhas é realizado de maneira diferente em cada um dos trabalhos. Por isso, um método auxiliar foi utilizado para realizar as comparações. Além disso, a metodologia proposta em (LAMBERT-TORRES et al., 2009) não trata problemas com múltiplas falhas. Por isso não é possível realizar o teste comparativo nesses casos.

Primeiramente foram comparados conjuntos de falhas em que o PSO apresentou uma única solução. A tabela 7 mostra cada um desses casos e os resultados encontrados pelos dois métodos em paralelo.

Também foram comparados os resultados em que o PSO teve mais de uma solução. A tabela 8 mostra as soluções encontradas. Pode-se perceber que o MOBFOA encontra um número maior de soluções. Isso acontece porque algumas funções objetivo consideradas no método mono-objetivo têm peso maior do que as outras. Vale destacar que as soluções encontradas pelo MOBFOA são resultado de uma única execução aleatória do método. No caso do PSO, para cada solução encontrada, uma execução foi realizada.

Diferente do PSO, o MOBFOA considera também as soluções que não restabelecem toda a carga demandada, devido à utilização das regras de dominância de Pareto. Dessa forma, a concessionária pode ponderar de acordo com a necessidade, escolhendo a melhor solução para cada situação específica. Fazer a ponderação das funções previamente impossibilita esse tipo de análise e deve garantir que os melhores pesos foram escolhidos.

Tabela 8 – Comparação dos resultados dos métodos MOBFOA e PSO - Várias soluções

Linha(s) Defeituosa(s)	MOBFOA			PSO		
	NA	NF	P (%)	NA	NF	P (%)
1 - 122	58 - 76 126 - 127	117 - 121	100,00			
	58 - 76 126 - 177	117 - 121	100,00	58 - 76 126 - 177	117 - 121	100,00
	19 - 57 20 - 56	20 - 48	100,00	19 - 57 20 - 56	20 - 48	100,00
	19 - 57 126 - 127	117 - 121	100,00	19 - 57 126 - 127	117 - 121	100,00
	19 - 57 126 - 177	117 - 121	100,00	19 - 57 126 - 177	117 - 121	100,00
	2 - 19 20 - 56	20 - 48	100,00			
	2 - 19 126 - 177	117 - 121	100,00			
				2 - 19 126 - 127	117 - 121	100,00
	19 - 57	117 - 121	87,68			
	58 - 76	117 - 121	87,68			
	2 - 19	117 - 121	87,68			
	Nenhuma	Nenhuma	54,51			
1 - 189	2 - 19 126 - 127	117 - 121	100,00			
	58 - 76 126 - 127	117 - 121	100,00	58 - 76 126 - 127	117 - 121	100,00
	19 - 57 126 - 177	177 - 121	100,00	19 - 57 126 - 177	177 - 121	100,00
	19 - 57 126 - 127	117 - 121	100,00	19 - 57 126 - 127	177 - 121	100,00
	2 - 19 126 - 177	177 - 121	100,00	2 - 19 126 - 177	177 - 121	100,00
				58 - 76 126 - 177	117 - 121	100,00
	Nenhuma	Nenhuma	62,87			
19 - 32	33 - 46	Nenhuma	100,00	33 - 46	Nenhuma	100,00
	2 - 19	Nenhuma	100,00	2 - 19	Nenhuma	100,00
	19 - 43	Nenhuma	100,00	19 - 43	Nenhuma	100,00
	30 - 39	Nenhuma	100,00	30 - 39	Nenhuma	100,00
	Nenhuma	Nenhuma	93,91			

7 Conclusão

Por meio do modelo desenvolvido, pode-se avaliar o desempenho de uma versão da técnica bio-inspirada na estratégia de forrageamento das bactérias. As adaptações feitas para a versão multi-objetivo se mostraram eficazes na busca por soluções Pareto-ótimas. Por meio das comparações com os resultados obtidos com a rotina de força bruta no caso do sistema de 70 barras e com os resultados obtidos pelo método PSO para o sistema de 201 barras, pode-se notar que o método conseguiu entregar praticamente todas as soluções ótimas esperadas para os casos testados, considerando os dois objetivos propostos.

O tempo de execução médio do algoritmo para os casos mais complexos, cerca de 10 minutos, não é adequado para o problema de restauração. Ele pode ser drasticamente melhorado com a otimização do código implementado. Como o foco do trabalho não foi a eficiência das rotinas, o modelo criado para o sistema elétrico assim como as rotinas de isolamento e cálculo de fluxo de potência podem ser melhoradas para melhorar o tempo de resposta do método. Apesar da baixa eficiência para a resolução do problema em tempo real, o método poderia ser usado em tempo de planejamento e poderia ser adaptado facilmente para o problema de reconfiguração de RDEE, considerando que muitas das informações necessárias já são calculadas no método proposto, como as perdas de potência nos condutores.

A otimização multi-objetivo é muito interessante para a proposta. Com essa abordagem soluções que maximizam a quantidade de cargas atendidas são colocadas em paralelo com outras que, mesmo não atendendo muitas cargas, minimizam o número de chaveamentos. Dessa forma, a empresa distribuidora pode tomar a melhor decisão, considerando o impacto causado para cada uma das decisões, no que diz respeito à dificuldade e o custo gerado para implementar a configuração proposta e a possível multa por falta de fornecimento, entre outros fatores não abordados pela solução. Além disso, pensando em todos os outros objetivos que podem ser considerados, como os sugeridos como trabalhos futuros, uma decisão muito mais assertiva pode ser tomada, principalmente pensando no contexto das *smart grids*.

Para melhorar o tempo de resposta do método, alguns fatores também devem ser levados em consideração, como:

- Realizar um estudo de convergência mais completo para ajustar mais adequadamente os parâmetros iniciais e os critérios de parada do MOBFOA;
- Usar uma estrutura de dados que permita mais agilidade na varredura e modificação do sistema, como a RNP, usando a representação do sistema através de ilhas

elétricas;

- Melhorar a estrutura de dados usada para armazenar o repositório das soluções Pareto-ótimas em cada geração e diminuir a complexidade computacional da rotina de atualização das soluções;
- Melhorar a convergência do MOBFOA aproveitando as características de *swarm* do BFOA também para a versão multi-objetivo.

O trabalho serviu como uma porta de entrada para os problemas de redes de distribuição elétrica e o desenvolvimento do protótipo torna mais acessível o desenvolvimento de soluções com melhores resultados. Outras sugestões de melhorias na qualidade dos resultados são propostas as seguintes implementações:

- Considerar as variáveis de estado como variáveis binárias, evitando o erro causado pelo ajuste dos valores contínuos;
- Considerar o tempo e os demais custos envolvidos em cada chaveamento, priorizando chaves remotas;
- Considerar a sequência correta para a realização dos chaveamentos;
- Considerar a ocorrência de falhas sequenciais;
- Considerar as perdas de potência para cada configuração;
- Considerar o balanceamento de carga entre os alimentadores;
- Considerar a presença de geradores distribuídos no sistema;
- Considerar os programas controle de demanda;
- Considerar a operação do sistema na topologia fracamente malhada.

Referências

- BASTOS, E. A. *Otimização de seções retangulares de concreto armado submetidas à flexo-compressão oblíqua utilizando algoritmos genéticos*. Dissertação (Mestrado) — Universidade Federal do Rio de Janeiro (UFRJ), 2004. 31, 32
- BROWN, R. E. Impact of smart grid on distribution system design. In: IEEE. *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*. [S.l.], 2008. p. 1–4. 18, 19
- CIVANLAR, S. et al. Distribution feeder reconfiguration for loss reduction. *IEEE Transactions on Power Delivery*, IEEE, v. 3, n. 3, p. 1217–1223, 1988. 21, 24
- DAS, D. A fuzzy multiobjective approach for network reconfiguration of distribution systems. *IEEE Transactions on Power Delivery*, v. 21, n. 1, p. 202–209, 2006. 21, 24, 56
- DEB, K. Multi-objective optimization using evolutionary algorithms: An introduction. *KanGAL Report*, n. 2011003, 2011. 31
- DEB, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, IEEE, v. 6, n. 2, p. 182–197, 2002. 41, 49
- DELBEM, A. C. B.; CARVALHO, A. C. P. L. de; BRETAS, N. G. Main chain representation for evolutionary algorithms applied to distribution system reconfiguration. *IEEE Transactions on Power Systems*, IEEE, v. 20, n. 1, p. 425–436, 2005. 22, 23
- DORIGO, M.; MANIEZZO, V.; COLORNI, A. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 26, n. 1, p. 29–41, 1996. 33
- EBERHART, R.; KENNEDY, J. A new optimizer using particle swarm theory. In: IEEE. *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*. [S.l.], 1995. p. 39–43. 33
- FARHANGI, H. The path of the smart grid. *IEEE power and energy magazine*, IEEE, v. 8, n. 1, 2010. 18, 19
- GOLSHANI, A. et al. Two-stage adaptive restoration decision support system for a self-healing power grid. *IEEE Transactions on Industrial Informatics*, IEEE, v. 13, n. 6, p. 2802–2812, 2017. 18
- HOLLAND, J. H. Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing*, SIAM, v. 2, n. 2, p. 88–105, 1973. 33
- HU, W.; YEN, G. G. Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system. *IEEE Transactions on Evolutionary Computation*, IEEE, v. 19, n. 1, p. 1–18, 2015. 41, 42, 48, 49
- KAUR, M.; KADAM, S. A novel multi-objective bacteria foraging optimization algorithm (mobfoa) for multi-objective scheduling. *Applied Soft Computing*, Elsevier, v. 66, p. 183–195, 2018. 10, 41, 42, 43, 48, 49, 50

- LAMBERT-TORRES, G. et al. Particle swarm optimization applied to system restoration. In: IEEE. *PowerTech, 2009 IEEE Bucharest*. [S.l.], 2009. p. 1–6. [10](#), [22](#), [24](#), [46](#), [53](#), [55](#), [62](#), [63](#), [67](#)
- MANSOUR, M. et al. Energy restoration in distribution systems using multi-objective evolutionary algorithm and an efficient data structure. In: IEEE. *PowerTech, 2009 IEEE Bucharest*. [S.l.], 2009. p. 1–7. [22](#), [23](#), [41](#)
- MARQUES, L. T. et al. A new multi-objective evolutionary algorithm for service restoration: Non-dominated sorting genetic algorithm-ii in subpopulation tables. In: IEEE. *PowerTech, 2017 IEEE Manchester*. [S.l.], 2017. p. 1–6. [41](#)
- MATHIAS-NETO, W. P. *Restauração automática de redes de distribuição de energia elétrica de grande porte com geração distribuída*. Dissertação (Mestrado) — Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira (UNESP), Fevereiro 2011. [23](#), [24](#)
- MATHIAS-NETO, W. P.; MANTOVANI, J. R. S. A node-depth encoding-based tabu search algorithm for power distribution system restoration. *Journal of Control, Automation and Electrical Systems*, Springer, v. 27, n. 3, p. 317–327, 2016. [23](#), [24](#)
- MAZZETTO, M. *Inteligência computacional para sistemas self-healing: um estudo de caso em redes elétricas inteligentes*. 2016. Monografia (Bacharelado em Engenharia de Computação), Universidade Tecnológica Federal do Paraná (UTFPR), Pato Branco, Brazil. [10](#), [28](#), [45](#), [46](#), [56](#)
- NBR, A. 5460: 1992. *Sistemas Elétricos de Potência-Terminologia*. [25](#)
- NIU, B. et al. Multi-objective bacterial foraging optimization. *Neurocomputing*, Elsevier, v. 116, p. 336–345, 2013. [41](#), [42](#), [43](#), [48](#)
- PANIGRAHI, B. et al. Multiobjective bacteria foraging algorithm for electrical load dispatch problem. *Energy Conversion and Management*, Elsevier, v. 52, n. 2, p. 1334–1342, 2011. [49](#)
- PASSINO, K. M. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Systems*, v. 22, n. 3, p. 52–67, 2002. [10](#), [34](#), [35](#), [37](#)
- SANCHES, D. S. *Algoritmos Evolutivos Multi-Objetivo para Reconfiguração de Redes em Sistemas de Distribuição de Energia Elétrica*. Tese (Doutorado) — Escola de Engenharia de São Carlos, Universidade de São Paulo (USP), 2013. [10](#), [25](#), [26](#), [31](#), [42](#)
- SANTOS, A. C. d. *Algoritmo evolutivo computacionalmente eficiente para reconfiguração de sistemas de distribuição*. Tese (Doutorado) — Universidade de São Paulo (USP), 2009. [19](#)
- SHIRMOHAMMADI, D. Service restoration in distribution networks via network reconfiguration. In: IEEE. *Transmission and Distribution Conference, 1991., Proceedings of the 1991 IEEE Power Engineering Society*. [S.l.], 1991. p. 626–632. [21](#), [24](#)
- SHIRMOHAMMADI, D. et al. A compensation-based power flow method for weakly meshed distribution and transmission networks. *IEEE Transactions on power systems*, IEEE, v. 3, n. 2, p. 753–762, 1988. [30](#)

- SIANO, P. Demand response and smart grids—a survey. *Renewable and sustainable energy reviews*, Elsevier, v. 30, p. 461–478, 2014. [18](#)
- SILVA, G. P. da. *Restauração de Redes de Energia Utilizando Algoritmos Genéticos Multiobjetivo*. Dissertação (Mestrado) — Universidade Federal de Minas Gerais (UFMG), Agosto 2014. [18](#), [23](#), [24](#), [27](#), [39](#), [42](#)
- TOUNE, S. et al. Comparative study of modern heuristic algorithms to service restoration in distribution systems. *IEEE Transactions on Power Delivery*, IEEE, v. 17, n. 1, p. 173–181, 2002. [22](#), [23](#)
- XING, B.; GAO, W.-J. Bacteria inspired algorithms. In: *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*. [S.l.]: Springer, 2014. cap. Bacteria Inspired Algorithms, p. 21–38. [36](#)
- YI, J. et al. Multi-objective bacterial foraging optimization algorithm based on parallel cell entropy for aluminum electrolysis production process. *IEEE Transactions on Industrial Electronics*, IEEE, v. 63, n. 4, p. 2488–2500, 2016. [41](#), [42](#)
- ZITZLER, E.; LAUMANN, M.; THIELE, L. Spea2: Improving the strength pareto evolutionary algorithm. *TIK-report*, Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK), v. 103, 2001. [41](#), [42](#)