



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
MESTRADO ACADÊMICO EM ENGENHARIA ELÉTRICA

FELIPE CARVALHO SAMPAIO

**SISTEMA DE PROTEÇÃO ADAPTATIVA APLICADO A REDES DE DISTRIBUIÇÃO
E MICRORREDES EM MÉDIA TENSÃO**

FORTALEZA

2019

FELIPE CARVALHO SAMPAIO

SISTEMA DE PROTEÇÃO ADAPTATIVA APLICADO A REDES DE DISTRIBUIÇÃO E
MICRORREDES EM MÉDIA TENSÃO

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica do Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de Concentração: Engenharia Elétrica

Orientadora: Profa. Ph.D. Ruth Pastôra Saraiva Leão

Coorientador: Prof. Dr. Raimundo Furtado Sampaio

FORTALEZA

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- S183s Sampaio, Felipe Carvalho.
Sistema de Proteção Adaptativa Aplicado a Redes de Distribuição e Microrredes em Média Tensão /
Felipe Carvalho Sampaio. – 2019.
144 f. : il. color.
- Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-
Graduação em Engenharia Elétrica, Fortaleza, 2019.
Orientação: Profa. Dra. Ruth Pastôra Saraiva Leão.
Coorientação: Prof. Dr. Raimundo Furtado Sampaio.
1. Proteção. 2. Microrrede. 3. Geração Distribuída. 4. Metaheurística. 5. Otimização. I. Título.
CDD 621.3
-

FELIPE CARVALHO SAMPAIO

SISTEMA DE PROTEÇÃO ADAPTATIVA APLICADO A REDES DE DISTRIBUIÇÃO E
MICRORREDES EM MÉDIA TENSÃO

Dissertação apresentada ao Curso de Mestrado Acadêmico em Engenharia Elétrica do Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de Concentração: Engenharia Elétrica

Aprovada em: 22 de Novembro de 2019

BANCA EXAMINADORA

Profa. Ph.D. Ruth Pastôra Saraiva
Leão (Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Raimundo Furtado Sampaio (Coorientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Giovanni Cordeiro Barroso
Universidade Federal do Ceará (UFC)

Prof. Dr. Bruno de Athayde Prata
Universidade Federal do Ceará (UFC)

A Deus,

Aos meus pais, Raimundo e Glória

A minha namorada, Andresa

AGRADECIMENTOS

A Deus, sem Ele, nada disso seria possível.

Aos meus pais, Raimundo e Glória, por todo o apoio e dedicação na minha formação intelectual e humana. Tudo o que eu me tornei foi graças a vocês.

A minha namorada, Andresa, por todo amor, amizade, compreensão e cumplicidade. Obrigado por estar presente em todos os momentos e caminhar sempre ao meu lado.

Aos professores do Programa de Pós Graduação em Engenharia Elétrica da Universidade Federal do Ceará, em especial, aos meus orientadores Profa. Ruth Pastôra Saraiva Leão e Prof. Raimundo Furtado Sampaio pelos ensinamentos e orientações.

Aos membros da banca avaliadora desse trabalho, professores Giovanni Cordeiro Barroso e Bruno de Athayde Prata, pelas excelentes contribuições ao trabalho.

A todos os colegas que tive o prazer de conhecer durante a pós graduação. Obrigado pelos momentos de discussões, troca de conhecimentos e descontração. Essa caminhada não teria sido a mesma sem vocês.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico pelo apoio financeiro durante o mestrado.

“A ciência será sempre uma busca, jamais uma descoberta. É uma viagem, nunca uma chegada.”

(Karl Popper)

RESUMO

A demanda por fontes de energia elétrica limpas e renováveis fez surgir novos conceitos para o sistema elétrico, como o de geração distribuída e o de microrredes. Apesar de trazerem diversos benefícios ao sistema elétrico, a geração distribuída e as microrredes causam grandes impactos na operação do sistema de distribuição de energia elétrica, sendo o impacto no sistema de proteção um dos mais relevantes. Este trabalho propõe um sistema de proteção adaptativa capaz de mitigar os impactos da operação de microrredes no sistema de proteção. O sistema é aplicado à proteção de sobrecorrente direcional e propõe uma nova metaheurística para o cálculo dos ajustes, o algoritmo do morcego direcional *fuzzy* adaptativo. A metaheurística proposta é validada utilizando os dados de três redes de referência: rede 3 barras, rede 9 barras e rede 30 barras. O desempenho do algoritmo foi comparado com o do Algoritmo Genético, Otimização por Enxame de Partículas, Evolução Diferencial, Algoritmo do Morcego e Algoritmo do Morcego Direcional. Os resultados demonstraram que o algoritmo apresentou alta velocidade de convergência, alta robustez, esforço computacional praticável e proveu baixa descoordenação ao sistema de proteção. O sistema de proteção adaptativa foi testado em uma microrrede em média tensão em vários cenários de topologia da rede, de conexão de geração distribuída e nos modos de operação conectada e desconectada da rede de distribuição. O desempenho do sistema proposto foi comparado ao de um sistema de proteção com ajustes fixos otimizados. Os resultados demonstram que o sistema proposto apresentou melhor desempenho quando comparado ao sistema com ajustes fixos, tornando o sistema de proteção mais rápido e mais confiável.

Palavras-chave: Proteção. Microrrede. Geração Distribuída. Metaheurística. Otimização.

ABSTRACT

The demand for renewable and clean energy sources rose new electrical systems concepts, as distributed generation and microgrids. Although they bring many benefits for the electrical system, distributed generation and microgrids cause huge impacts in the operation of electrical distribution systems, being the impact on protection system one of the most relevant. This work presents an adaptive protection system to mitigate the microgrid operation impact on protection system. The system is applied to directional overcurrent protection relays and propose a new metaheuristic for optimal settings calculation, the Fuzzy-Adaptive Directional Bat Algorithm (FDBA). The algorithm is validated using the data of three benchmark grids: 3-bus, 9-bus and 30-bus grids. The proposed algorithm performance is compared against the performance of Genetic Algorithm, Particle Swarm Optimization, Differential Evolution, Bat Algorithm and Directional Bat Algorithm. The results demonstrate that FDBA presented high convergence speed, low discoordination, high robustness and acceptable computational burden. The adaptive protection system was tested in a microgrid under different scenarios of grid topology, distributed generation connection and grid connection: grid-connected and island. The proposed system performance was compared against a fixed optimal settings protection system performance. The results showed that the proposed system presented better performance, allowing more speed and lower risk of discoordination for the protection system.

Keywords: Protection. Microgrid. Distributed Generation. Metaheuristics. Optimization

LISTA DE FIGURAS

Figura 1 – Suprimento mundial de energia por fonte ao longo dos anos	26
Figura 2 – Produção de energia elétrica por tipo de fonte de 1986 a 2018	27
Figura 3 – Geração de energia a partir de fontes renováveis de 1998 a 2018	27
Figura 4 – Comparação entre os modelos de geração distribuída e centralizada.	28
Figura 5 – Estrutura básica de uma microrrede.	29
Figura 6 – Fluxo bidirecional em rede em anel (a) e em rede de distribuição com geração distribuída (b)	32
Figura 7 – Representação unifilar de um relé de sobrecorrente direcional	33
Figura 8 – Diagrama fasorial das grandezas envolvidas no relé de sobrecorrente direcional.	34
Figura 9 – Efeito do tipo de curva (a), do <i>PS</i> (b) e <i>TDS</i> (c) no comportamento da curva de operação.	35
Figura 10 – Zonas de proteção dos relés de sobrecorrente direcionais.	37
Figura 11 – Índice de tempo de coordenação	38
Figura 12 – Conjuntos crisp (a) e fuzzy (b) de pessoas altas	44
Figura 13 – Modelo de sistema de inferência <i>fuzzy</i>	45
Figura 14 – Comparação do movimento do morcego nos algoritmo AM e AMD.	50
Figura 15 – Variáveis <i>fuzzy</i> associadas à iteração.	54
Figura 16 – Variáveis <i>fuzzy</i> associadas à diversidade.	54
Figura 17 – Variáveis <i>fuzzy</i> associadas à amplitude.	55
Figura 18 – Variáveis <i>fuzzy</i> associadas à taxa de emissão de pulsos.	55
Figura 19 – Variáveis <i>fuzzy</i> associadas ao fator <i>w</i>	55
Figura 20 – Variáveis <i>fuzzy</i> associadas ao fator a frequência	55
Figura 21 – Rede 3 barras.	58
Figura 22 – Evolução da convergência dos algoritmos para a rede 3 barras.	59
Figura 23 – Índice de tempo de coordenação dos pares de relés principal e <i>backup</i> para a rede 3 barras.	59
Figura 24 – Intervalo de confiança das médias das funções objetivos de cada algoritmo.	60
Figura 25 – Intervalo de confiança das médias das funções objetivos do OEP, AMD e AMDFA.	61
Figura 26 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo	61
Figura 27 – Rede 9 barras.	62

Figura 28 – Evolução da convergência dos algoritmos para a rede 9 barras.	63
Figura 29 – Índice de tempo de coordenação dos pares de relés principal e <i>backup</i> para a rede 9 barras.	63
Figura 30 – Intervalo de confiança das médias das funções objetivos de cada algoritmo. .	64
Figura 31 – Intervalo de confiança das médias das funções objetivos do OEP, AMD e AMDFA.	65
Figura 32 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo	65
Figura 33 – Rede 30 barras.	66
Figura 34 – Evolução da convergência dos algoritmos para a rede 30 barras.	67
Figura 35 – Índice de tempo de coordenação dos pares de relés principal e <i>backup</i> para a rede 30 barras	67
Figura 36 – Intervalo de confiança das médias das funções objetivos de cada algoritmo. .	68
Figura 37 – Intervalo de confiança das médias das funções objetivos do ED, AMD e AMDFA.	69
Figura 38 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo	69
Figura 39 – Arquitetura do Sistema de Proteção Adaptativa.	73
Figura 40 – Fluxograma do SPA	75
Figura 41 – Rede canadense de distribuição.	76
Figura 42 – Médias dos tempos de operação principal para o sistema conectado a rede (a) e ilhado (b).	80
Figura 43 – Médias dos tempos de operação de <i>backup</i> para o sistema conectado a rede (a) e ilhado (b).	81
Figura 44 – Distribuição de frequência dos ITCs obtidos pelo SPA (a) e SAF (b).	82
Figura 45 – Tamanho da população - AG.	122
Figura 46 – Taxa de mutação - AG.	123
Figura 47 – Critério de seleção - AG.	124
Figura 48 – Tamanho da população - EOP.	125
Figura 49 – Fator de inércia inicial - EOP.	126
Figura 50 – Fator de inércia final - EOP.	127
Figura 51 – Constante c_1 - EOP.	128
Figura 52 – Tamanho da população - ED.	129
Figura 53 – Constante F - ED.	130

Figura 54 – Taxa de cruzamento - ED.	131
Figura 55 – Tamanho da população - AM.	132
Figura 56 – Constante α - AM.	133
Figura 57 – Constante γ - AM.	134
Figura 58 – Frequência mínima - AM.	135
Figura 59 – Frequência máxima - AM.	136
Figura 60 – Tamanho da população - AMD.	137
Figura 61 – Amplitude inicial - AMD.	138
Figura 62 – Amplitude final - AMD.	139
Figura 63 – Frequência de emissão inicial - AMD.	140
Figura 64 – Frequência de emissão final - AMD.	141
Figura 65 – Frequência mínima - AMD.	142
Figura 66 – Frequência máxima - AMD.	143
Figura 67 – Tamanho da população - AMDFA.	144

LISTA DE TABELAS

Tabela 1 – Técnicas de cálculo de ajustes aplicadas em SPAs.	21
Tabela 2 – Metaheurísticas aplicadas à coordenação ótima de relés de sobrecorrente direcionais.	25
Tabela 3 – Constantes das curvas de operação definidas pela IEC 60255-3.	35
Tabela 4 – Sistema de regras utilizado no SIF.	56
Tabela 5 – RTC dos relés da rede 3 barras.	58
Tabela 6 – Análise de robustez - rede 3 barras.	60
Tabela 7 – Análise de robustez - rede 9 barras.	64
Tabela 8 – Análise de robustez - rede 30 barras.	68
Tabela 9 – Parâmetros de entrada do SPA	74
Tabela 10 – Casos de conexão de GD considerados.	77
Tabela 11 – Casos de mudança topológica da rede.	78
Tabela 12 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 3 barras.	91
Tabela 13 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 9 barras.	92
Tabela 14 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 30 barras.	93

LISTA DE ABREVIATURAS E SIGLAS

AEDI	Algoritmo de Ervas Daninhas Invasivas
AF	Algoritmo das Formigas
AG	Algoritmo Genético
AMD	Algoritmo do Morcego Direcional
AMDFA	Algoritmo do Morcego Direcional <i>Fuzzy</i> Adaptativo
AOC	Algoritmo de Otimização dos Cucos
BG	Busca Gravitacional
CORSD	Coordenação Ótima de Relés de Sobrecorrente Direcionais
ED	Evolução Diferencial
GA	Grupo de Ajustes
GD	Geração Distribuída
ITC	Índice de Tempo de Coordenação
OBB-PL	Otimização Baseado em Biogeografia hibridizado com PL
OEP	Otimização por Enxame de Partículas
PL	Programação Linear
PNL	Programação Não Linear
PNLIM	Programação Não Linear Inteira Mista
PQS	Programação Quadrática Sequencial
RDA	Redes de Distribuição Ativas
RED	Recursos Energéticos Distribuídos
RTC	Relação de Transformação de Corrente
SAF	Sistema com Ajustes Fixos
SCADA	<i>Supervisory, Control And Data Aquisition</i>
SIF	Sistema de Inferência <i>Fuzzy</i>
SPA	Sistema de Proteção Adaptativa
TC	Transformador de Corrente
TP	Transformador de Potencial

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Objetivos	18
1.2	Estrutura do trabalho	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	Revisão bibliográfica	20
2.2	Geração distribuída	26
2.3	Microrredes	29
2.4	Proteção de microrredes	30
2.5	Proteção de sobrecorrente direcional	32
2.5.1	<i>Princípio de funcionamento</i>	32
2.5.2	<i>Curva de operação</i>	34
2.5.3	<i>Coordenação de relés direcionais de sobrecorrente</i>	36
2.6	Considerações finais	38
3	COORDENAÇÃO OTIMIZADA DE RELÉS DIRECIONAIS DE SOBRECORRENTE	39
3.1	Formulação do problema de coordenação ótima de relés direcionais de sobrecorrente	39
3.1.1	<i>Restrições</i>	39
3.1.1.1	<i>Restrição de coordenação</i>	39
3.1.2	<i>Restrição da escolha da corrente de pickup</i>	40
3.1.2.1	<i>Restrição da escolha do multiplicador TDS</i>	41
3.1.3	<i>Função objetivo</i>	41
3.2	Otimização Metaheurística	42
3.3	Sistemas de inferência fuzzy	43
3.4	Algoritmo do Morcego Direcional Fuzzy Adaptativo	45
3.4.1	<i>Algoritmo do morcego e algoritmo do morcego direcional</i>	47
3.4.2	<i>Metaheurística proposta</i>	52
3.5	Validação do algoritmo do morcego direcional fuzzy adaptativo	57
3.5.1	<i>Sistema 3 barras</i>	58
3.5.2	<i>Sistema 9 barras</i>	62

3.5.3	<i>Sistema 30 barras</i>	65
3.5.4	<i>Discussão dos resultados</i>	69
3.6	Considerações finais	71
4	SISTEMA DE PROTEÇÃO ADAPTATIVA APLICADO A MICRORREDES	73
4.1	Sistema de Proteção Adaptativa	73
4.1.1	<i>Arquitetura do SPA</i>	73
4.1.2	<i>Algoritmo do SPA</i>	74
4.2	Metodologia dos testes	75
4.2.1	<i>Microrrede teste</i>	75
4.2.2	<i>Cenários de testes</i>	77
4.2.2.1	<i>Parâmetros de simulação</i>	77
4.2.3	<i>Sistema com ajustes fixos otimizados</i>	78
4.3	Comparação dos resultados obtidos pelo SPA e pelo sistema com ajustes fixos	79
4.4	Considerações finais	83
5	CONCLUSÕES E TRABALHOS FUTUROS	84
5.1	Trabalhos Publicados	85
	REFERÊNCIAS	86
	ANEXOS	90
	ANEXO A – Dados de curto circuito das redes testes	91
A.1	Curto-circuito na rede 3 barras	91
A.2	Curto-circuito na rede 9 barras	92
A.3	Curto-circuito na rede 30 barras	93
	ANEXO B – Código em MATLAB das metaheurísticas implementadas	94
B.1	Algoritmo Genético	94
B.2	Otimização por Enxame de Partículas	100
B.3	Evolução Diferencial	104
B.4	Algoritmo do Morcego	108
B.5	Algoritmo do Morcego Direcional	111
B.6	Algoritmo do Morcego Direcional <i>Fuzzy</i> Adaptativo	115
	ANEXO C – Teste ANOVA dos parâmetros dos algoritmos	121

C.1	Valores adotados para cada parâmetro	121
------------	---	------------

1 INTRODUÇÃO

A demanda por energia elétrica mundial vem crescendo ao longo dos anos (+3,5 % em 2018) (ENERDATA, 2019), e os combustíveis fósseis (carvão e gás natural) lideram na participação de produção da eletricidade (BRITISH PETROLEUM, 2019). Com o esgotamento gradual das reservas energéticas de origem não renovável, o aumento crescente da demanda por energia e a preocupação com desenvolvimento econômico e social sustentável, muitos países, instituições e indústrias do setor elétrico têm investido na pesquisa, desenvolvimento e integração de Recursos Energéticos Distribuídos (RED) às redes elétricas de distribuição de energia, surgindo o conceito de redes elétricas de distribuição ativas (HOSSAIN *et al.*, 2019).

A integração de fontes renováveis nos sistemas de distribuição de energia elétrica também trouxe a tona um novo paradigma de geração, a geração distribuída, que é caracterizada pela presença de pequenas unidades de geração nas redes de distribuição e próximas aos consumidores. Este novo conceito difere do modelo de geração convencional, em que a energia provida aos consumidores vem de grandes centrais de geração localizadas distantes dos centros de carga (COLMENAR-SANTOS *et al.*, 2016).

Atualmente, o modelo de rede de distribuição ativa mais promissor é o de microrrede. Uma microrrede é composta de recursos energéticos distribuídos e cargas controláveis, que operam de forma coordenada. A implantação de microrredes permite uma maior penetração de fontes de Geração Distribuída (GD) com menor complexidade operacional para as redes de distribuição. As microrredes são projetadas para operar conectadas (*on grid*) e/ou desconectadas (*off grid*) da rede de distribuição. No primeiro modo de operação, a microrrede atua como uma única entidade controlável em relação à rede, sendo vista como carga ou fonte. Em geral, a capacidade de operar no modo isolado é a principal característica que diferencia uma microrrede de uma rede de distribuição com geração distribuída, conferindo maior confiabilidade no suprimento de energia elétrica aos seus consumidores (HATZIARGYRIOU, 2014).

A penetração em larga escala da geração distribuída, com fontes eólicas e fotovoltaicas, tem provocado grandes transformações na forma de operar as redes de distribuição de energia. A natureza intermitente, estocástica e distribuída dessas fontes afetam a confiabilidade e a estabilidade da rede elétrica de distribuição, causando fluxo de potência bidirecional, interação entre GDs e dispositivos de controle de reativo, maior complexidade dos sistemas de proteção, aumento da magnitude das correntes de falta e flutuações de tensão (MANDITEREZA; BANSAL, 2016).

No cerne dessas evoluções, as empresas de energia têm investido na transformação das redes de distribuição com topologia radial com fluxo de potência unidirecional, em redes com topologia radial com recurso, controladas por sistemas de recomposição automática capazes de localizar, isolar e restabelecer o fornecimento de energia, após a ocorrência de uma falta permanente (SAMPAIO *et al.*, 2017).

A evolução topológica das redes de distribuição com GD e a capacidade de ilhamento das microrredes proporcionam mudanças na magnitude e na direção das correntes de curto-circuito e de carga, trazendo consequências para a confiabilidade do sistema de proteção em função da perda de seletividade (*trip* falso, *trip* intempestivo), sensibilidade (proteção cega) e velocidade de atuação da proteção (*trip* atrasado) (HATZIARGYRIOU, 2014). Em Brearley e Prabu (2017), os autores citam como solução para esses impactos o uso de limitadores de corrente de falta; sistemas de proteção adaptativa; proteção de distância; proteção baseada em variáveis; e esquemas de proteção baseados em sistemas multiagentes. Dentre essas alternativas, os autores destacam a aplicação de Sistema de Proteção Adaptativa (SPA) como a melhor solução.

Um SPA é um sistema capaz de alterar os ajustes dos relés de proteção de acordo com as condições operativas da rede elétrica. Desta forma, o SPA altera os ajustes das funções de proteção adaptados para cada condição operativa, ao contrário do uso de ajustes fixos adotados nos sistemas de proteção tradicionais (GERS; HOLMES, 2011).

Diversos esquemas de proteção adaptativa têm sido propostos para alterar as configurações do relé para adequá-las às condições prevalentes da rede elétrica de distribuição. Neste trabalho é proposto um SPA para mudança automática de ajustes otimizados para relés de sobrecorrente direcional, visando mitigar os impactos da operação de GDs e microrredes no sistema de proteção.

1.1 Objetivos

O presente trabalho tem como objetivo geral apresentar um sistema de proteção adaptativa para proteção de sobrecorrente direcional aplicado a redes elétricas de distribuição e microrredes, que utiliza uma metaheurística para calcular os ajustes do sistema de proteção.

Como objetivos específicos do presente trabalho, podem ser destacados:

- (i) Formular o problema de coordenação ótima de relés de sobrecorrente direcionais como um problema de otimização não linear;
- (ii) Propor uma nova metaheurística, o algoritmo do morcego direcional *fuzzy* adaptativo, para

- coordenação ótima de relés de sobrecorrente direcionais aplicado à proteção adaptativa;
- (iii) Comparar os resultados do algoritmo proposto com outras metaheurísticas apresentadas na literatura;
 - (iv) Propor uma arquitetura de sistema de proteção adaptativa aplicado a microrredes;
 - (v) Aplicar o sistema proposto em uma microrrede teste operando em diversos cenários, com vistas a avaliar seu desempenho na resolução de problemas reais.

1.2 Estrutura do trabalho

Este trabalho está dividido em cinco capítulos, a saber:

Capítulo 1 em que foi apresentada uma introdução sobre o trabalho.

No Capítulo 2 é apresentada uma revisão literária de modelos de proteção adaptativa e a fundamentação teórica abordando sobre: geração distribuída, microrredes, proteção de microrredes, proteção de sobrecorrente direcional, coordenação de relés de sobrecorrente direcionais

No Capítulo 3 são apresentadas a formulação matemática do problema de coordenação ótima de relés de sobrecorrente direcionais e os conceitos introdutórios de otimização metaheurística e sistemas de inferência *fuzzy*. Além disso, é feita uma introdução ao AMDFA e apresentados os resultados da validação do mesmo em redes de referência.

No Capítulo 4 é apresentado o sistema de proteção adaptativo proposto neste trabalho e os resultados da validação do algoritmo em uma microrrede.

No Capítulo 5 são apresentadas as conclusões do trabalho a partir dos resultados obtidos e apresentadas propostas para desenvolvimentos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados a revisão literária sobre proteção adaptativa e os conceitos fundamentais para o desenvolvimento deste trabalho, como: geração distribuída, microrredes, proteção de microrredes e proteção de sobrecorrente direcional

No próximo tópico é apresentada a revisão bibliográfica do trabalho, identificando as lacunas da literatura que este trabalho visa preencher.

2.1 Revisão bibliográfica

Para mitigação dos impactos da operação de sistemas de distribuição no sistema de proteção, vários esquemas de SPA vêm sendo propostos. Em Laaksonen *et al.* (2014) é proposto um SPA aplicado a microrredes, que seleciona o Grupo de Ajustes (GA), previamente calculado pelo projetista, mais apropriado para cada relé de proteção com base no modo de conexão e topologia da microrrede.

O SPA proposto em Singh *et al.* (2016) monitora a rede e seleciona o GA mais apropriado, a partir de uma lista de GAs otimizada previamente, por meio de lógica fuzzy.

Apesar dos autores discutirem a possibilidade de alterar ajustes em tempo real, o SPA proposto em Paspaliotopoulos *et al.* (2017) calcula GAs otimizados para cada relé utilizando técnicas de Programação Não Linear (PNL).

Em Ojaghi e Mohammadi (2018) é aplicada uma técnica de *clusterização* para obter GAs que garantem a coordenação em cada modo de operação da rede elétrica. Os ajustes são calculados utilizando Programação Linear (PL).

CHATTOPADHYAY B.; SACHDEV (1996) propõe um sistema que calcula ajustes otimizados utilizando PL e envia-os em tempo real para os relés.

Os autores em Mahat *et al.* (2011) propõem um SPA aplicado a microrredes, que utiliza um algoritmo baseado em regras para calcular novos ajustes sempre que a microrrede muda de modo de operação (ilhado ou conectado à rede) e que uma unidade de geração se conecta ou desconecta da rede.

Coffele *et al.* (2015) propõem um SPA que utiliza um algoritmo baseado em regras para calcular novos ajustes sempre que há mudanças significativas na rede elétrica.

Em Liu *et al.* (2017), o SPA proposto calcula novos ajustes a partir de um sistema de regras com base no modo de controle das turbinas eólicas conectadas à rede elétrica.

Otimizadores não lineares são aplicados em Alam (2018) para o cálculo dos ajustes, os quais são enviados sempre que há mudanças significativas na rede elétrica.

O SPA proposto em Jain *et al.* (2019) utiliza a lógica interna dos relés para recalcular a corrente de *pickup* dos relés sempre que alguma mudança ocorre na rede, eliminando a necessidade de comunicação com um sistema de controle central.

Na Tabela 1 é apresentado um resumo das técnicas aplicadas para calcular os ajustes dos relés de proteção em sistemas de proteção adaptativa propostos na literatura.

Tabela 1 – Técnicas de cálculo de ajustes aplicadas em SPAs.

Técnica	Trabalho
Algoritmos baseado em regras	Laaksonen <i>et al.</i> (2014), Singh <i>et al.</i> (2016), Mahat <i>et al.</i> (2011), Liu <i>et al.</i> (2017), Jain <i>et al.</i> (2019)
Algoritmos de otimização determinísticos	CHATTOPADHYAY B.; SACHDEV (1996), Coffele <i>et al.</i> (2015) Papaspiliotopoulos <i>et al.</i> (2017), Alam (2018), Ojaghi e Mohammadi (2018)

Fonte: O próprio autor.

Os SPAs apresentados utilizam ou algoritmos baseados em regras ou algoritmos de otimização determinísticos para calcular os ajustes dos relés. No entanto, existem na literatura diversos trabalhos que propõem algoritmos de cálculos de ajustes para sistemas de proteção.

Mais especificamente, com relação aos relés de sobrecorrente direcional, diversas abordagens vêm sendo propostas para sistematizar o cálculo dos seus ajustes. Estes métodos podem ser classificados em três categorias: tentativa e erro, análise topológica e métodos de otimização (HUSSAIN *et al.*, 2013). Os métodos de tentativa e erro apresentam convergência lenta e não garantem a otimalidade dos ajustes calculados. Visando o uso de algoritmos mais rápidos, foram propostas técnicas para quebra de laços baseada na teoria dos grafos, denominadas de análise topológica. Essas técnicas apresentam melhores resultados e convergência mais rápida, no entanto apresentam alta dependência dos valores iniciais e nenhuma garantia de otimalidade (HUSSAIN *et al.*, 2013). Esses fatores contribuíram para um crescente interesse na aplicação dos algoritmos de otimização para cálculo de ajustes de relés direcionais de sobrecorrente.

A primeira vez que a coordenação de relés direcionais de sobrecorrente foi modelado e resolvido como um problema de otimização foi em Urdaneta *et al.* (1988). No trabalho, o problema de Coordenação Ótima de Relés de Sobrecorrente Direcionais (CORSD) foi modelado como um problema de programação linear e resolvido utilizando o algoritmo *Simplex*. Para aplicar esta modelagem, define-se o valor da corrente de *pickup* previamente e adota-se apenas o *Dial* como variável do problema. A seguir, outros trabalhos também aplicaram técnicas de

PL para resolver o problema de CORSD, como: método *Dual Simplex* (ELRAFIE; IRVING, 1993), *Simplex* de duas fases (CHATTOPADHYAY B.; SACHDEV, 1996) e programação linear intervalar (NOGHABI *et al.*, 2010).

No entanto, a modelagem do problema de CORSD como um problema de programação linear impõe a limitação de que o problema deve ter característica linear. A modelagem como problema de programação não linear permite que outras variáveis sejam consideradas no problema, como a corrente de *pickup* e as constantes das curvas de operação, dando maior versatilidade e possibilidade de otimização, apesar da maior complexidade. Desta forma, outros trabalhos propuseram a aplicação de técnicas de PNL para solução do problema de CORSD, como em Birla *et al.* (2006) e Papaspiliotopoulos *et al.* (2017).

Em 2004, uma nova proposta de modelagem para o problema de CORSD utilizando Programação Não Linear Inteira Mista (PNLIM) é proposta em Zeienldin *et al.* (2004). Esta abordagem procura tornar o modelo do problema mais próximo da aplicação real, em que os valores de corrente de *pickup* são escolhidos a partir de valores discretos e os de *dial* são valores contínuos.

Adotando um modelo não linear para o problema, a corrente de *pickup* é considerada uma variável contínua e o problema de otimização acaba perdendo precisão, uma vez que é necessário aproximar o valor final obtido no processo de otimização do valor da corrente de *pickup* disponível no relé. Apesar de tornar o modelo do problema mais preciso, a resolução do problema como PNLIM torna-se mais complexa. Além disso, nos relés digitais atuais, os valores discretos de corrente de *pickup* disponíveis são muito próximos entre si e podem ser considerados como contínuos sem perdas de precisão impactantes.

As técnicas de otimização de PL e PNL são técnicas exatas de otimização. Estas técnicas são capazes de obter a solução ótima e de garantir a otimalidade da solução obtida. Dependendo da complexidade do problema de otimização, estas técnicas apresentam excelente desempenho computacional e, em geral, fornecem boas soluções. No entanto, em problemas não convexos, ou seja, com vários máximos e mínimos locais, estas técnicas tendem a exigir maior esforço computacional, dependendo do tamanho do problema, e apresentam alta dependência de valores iniciais (TALBI, 2009).

O problema de CORSD é um problema de otimização multimodal e, dependendo da quantidade de relés da rede, pode tornar-se muito grande. Devido a isso, o interesse na aplicação de metaheurísticas na solução do problema de CORSD vem crescendo nos últimos

anos. Diferentemente das técnicas de otimização exatas, as metaheurísticas conseguem fornecer resultados suficientemente bons em problemas de larga escala em um tempo factível. No entanto, nestas técnicas, não há garantia de obtenção da solução ótima global.

Em So *et al.* (1997), foi proposta, pela primeira vez, a aplicação de um técnica metaheurística para solucionar o problema de CORSD. No trabalho, foi aplicado um Algoritmo Genético (AG) com codificação binária para solução do problema. Desde então, vários outros trabalhos vem propondo aplicação de técnicas metaheurísticas para solução deste problema.

Em Mansour *et al.* (2007) foi aplicado uma versão modificada do algoritmo de Otimização por Enxame de Partículas (OEP). No trabalho o problema é modelado como um problema de programação linear e o desempenho do OEP modificado é comparado ao desempenho do *Simplex*, demonstrando superioridade do algoritmo proposto.

Noghabi *et al.* (2009) propõe a aplicação de um algoritmo metaheurístico híbrido AG-PL para solução do problema de CORSD considerando diferentes topologias de rede. A performance do AG-PL foi comparada com o desempenho do AG tradicional e os resultados demonstraram que a hibridização melhorou substancialmente o desempenho do AG.

Bedekar e Bhide (2011) aplica um algoritmo híbrido AG-PNL, em que o AG é executado inicialmente para fornecer soluções iniciais para o PNL. Os resultados demonstram que a hibridização forneceu melhores resultados em relação ao AG tradicional.

O algoritmo *Seeker*, inspirado no comportamento humano, é aplicado em Amraee (2012) para solucionar o problema de CORSD. O algoritmo foi testado em três redes testes distintas e teve desempenho superior ao de três outros algoritmos propostos anteriormente na literatura.

Em Najy *et al.* (2013) é aplicado um AG para otimizar a alocação de limitadores de corrente de falta e minimizar o tempo de operação de relés de sobrecorrente direcionais. O algoritmo é aplicado a microrredes, portanto a otimização é feita considerando as duas condições da rede: ilhada e conectada à rede. Os resultados demonstraram que a alocação ótima de limitadores de corrente de falta permitiu uma melhoria do tempo de operação do sistema de proteção.

Chelliah *et al.* (2014) propõe dois algoritmos de evolução diferencial caóticos baseados em oposição. Os algoritmos foram testados em quatro redes testes e demonstraram resultados superiores a algoritmos propostos anteriormente.

Albasri *et al.* (2015) aplicou uma metaheurística híbrida Otimização Baseado em

Biogeografia hibridizado com PL (OBB-PL) ao problema de CORSD. O algoritmo apresentou desempenho computacional superior a outros já propostos na literatura. Segundo os autores, graças ao baixo tempo computacional demandado, o OBB-PL é indicado para aplicações *online*.

Shih *et al.* (2014) compara o desempenho dos algoritmos AG, Evolução Diferencial (ED) e Algoritmo das Formigas (AF) aplicados a CORSD em modo de execução *online*. Os algoritmos são testados utilizando uma rede de testes em vários casos distintos. Segundo os autores, o ED superou o desempenho dos demais para aplicações em modo *online*.

Em Jevtić e Radosavljević (2016) é aplicado o algoritmo de Busca Gravitacional (BG) hibridizado com a técnica de PNL Programação Quadrática Sequencial (PQS). O PQS realiza o processo de busca local do algoritmo GSA. Segundo os autores, o algoritmo demonstrou soluções efetivas e de alta qualidade.

Vatani *et al.* (2016) propõe a aplicação de um algoritmo genético baseado em lógica *fuzzy*. Nesta abordagem é aplicado um sistema de inferência *fuzzy* para atualizar o peso da função de penalidade de descoordenação da função objetivo. O algoritmo proposto é comparado a outro AG proposto na literatura e demonstra desempenho superior.

Shih *et al.* (2017) apresenta um comparativo entre diferentes versões de ED aplicado ao problema de CORSD. Os resultados demonstram que o ED melhorado superou os demais algoritmos nos critérios de tempo de execução, função objetivo e desvio padrão.

Dehghanpour *et al.* (2018) propõe a aplicação do Algoritmo de Otimização dos Cucos (AOC) hibridizado com PL aplicado a CORSD e alocação ótima de limitadores de corrente de falta em microrredes. Os resultados são obtidos para os dois modos de operação da microrrede: ilhado e conectado à rede elétrica. Os resultados são comparados aos resultados obtidos por OEP e AG e é demonstrado que o AOC-LP apresenta desempenho mais eficiente que os outros algoritmos.

Srinivas e K (2019) aplica a metaheurística Algoritmo de Ervas Daninhas Invasivas (AEDI) hibridizado com PQS. O algoritmo é testado em comparação com outros algoritmos já propostos e com o AEDI tradicional. Segundo os autores, o AEDI-PQS obteve soluções mais efetivas e melhores para o problema de CORSD.

Na Tabela 2 é apresentado um levantamento das técnicas de otimização metaheurística aplicadas a coordenação ótima de relés de sobrecorrente direcionais.

As técnicas mais aplicadas ao problema de CORSD são algoritmos genéticos e evolução diferencial. No entanto, nos últimos anos, muitos trabalhos vem sendo propostos com

Tabela 2 – Metaheurísticas aplicadas à coordenação ótima de relés de sobrecorrente direcionais.

Metaheurística	Trabalho
AG	(So et al., 1997), (NOGHABI et al., 2009), (BEDEKAR; Bhide, 2011), (NAJY et al., 2013), (VATANI et al., 2016)
OEP	(MANSOUR et al., 2007)
Seeker	(AMRAEE, 2012)
ED	(CHELLIAH et al., 2014), (SHIH et al., 2014), (SHIH et al., 2017)
OBB	(ALBASRI et al., 2015)
BG	(JEVTIC; RADOSAVLJEVIC, 2016)
AOC	(DEHGHANPOUR et al., 2018)
AEDI	(SRINIVAS; K, 2019)

Fonte: O próprio autor.

o intuito de investigar o desempenho de metaheurísticas na solução do problema de CORSD. Seguindo esta linha, o presente trabalho propõe uma nova metaheurística, o Algoritmo do Morcego Direcional *Fuzzy* Adaptativo, baseada no Algoritmo do Morcego, outra metaheurística aplicada em diversos problemas de otimização, mas ainda não aplicada no problema de CORSD.

Quanto ao tipo de aplicação, os algoritmos propostos para CORSD podem ser aplicados em dois modos de execução:

- *Offline*: Quando o algoritmo é utilizado para cálculo prévio dos ajustes dos relés. Em sistemas de proteção com ajustes fixos, pode ser aplicado para encontrar um grupo de ajustes para cada relé que se adeque a todos os modos de operação da rede. Em sistemas com proteção adaptativa, pode ser aplicado para encontrar grupos de ajustes que se adequem melhor a modos específicos de operação da rede. Como não operam em tempo real, esses algoritmos são projetados para ser executados durante um grande número de iterações e fornecer o melhor ajuste possível.
- *Online*: Quando o algoritmo é utilizado para cálculo dos ajustes em tempo real. Em sistemas de proteção adaptativa, pode ser utilizado para envio dos ajustes sempre que houver mudanças na rede elétrica. Neste caso, há a preocupação em fornecer os melhores ajustes possíveis dentro de um intervalo de tempo viável. Além disso, o algoritmo deve apresentar robustez suficiente para fornecer bons resultados sempre que solicitado, independente do modo de operação da rede elétrica e sem a necessidade de um operador para adaptar os parâmetros do algoritmo.

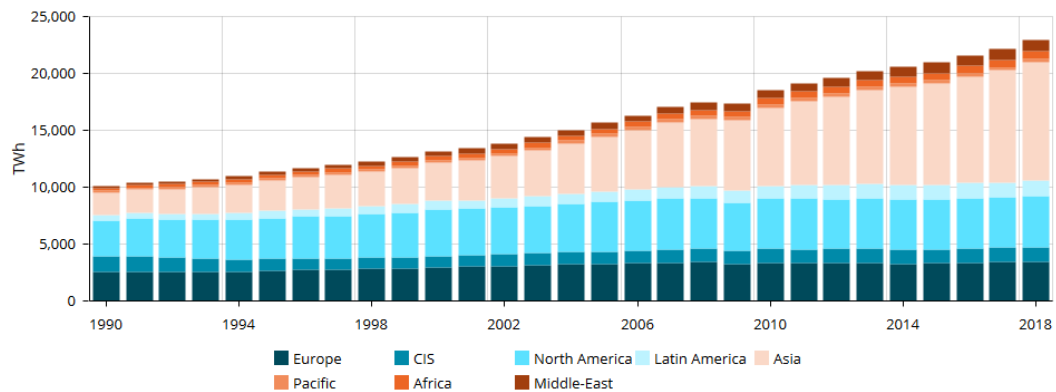
Dentre os trabalhos citados, apenas Shih *et al.* (2014) e Albasri *et al.* (2015) citam a possibilidade de aplicação dos algoritmos propostos em sistemas que operem em modo *online*. Em Shih *et al.* (2014), segundo os autores, o algoritmo evolução diferencial é o algoritmo mais apto para aplicações online, pois apresentou as melhores características de tempo de execução, qualidade de resultado, robustez e capacidade de convergência. No entanto, para

comparar os algoritmos quanto ao tempo de execução, apenas foi utilizada uma comparação relativa entre os algoritmos, sem nenhum parâmetro externo ou definição de aplicação específica de coordenação *online*. Não é utilizado nenhum critério absoluto que defina se o tempo de execução dos algoritmos é praticável. Além disso, todos os algoritmos testados necessitam que um projetista defina os parâmetros de operação para alcançar os melhores resultados. Já em Albasri *et al.* (2015) é citado que o OBB-PL apresenta boa característica para aplicações *online*, devido a sua capacidade de fornecer boas soluções e de requerer baixo tempo computacional. No entanto, é avaliado apenas o tempo computacional para atingir soluções viáveis, ou seja, com baixa descoordenação. Além disso, assim como em Shih *et al.* (2014), o BBO-LP necessita de parametrização para operar em novos modos de operação da rede elétrica.

2.2 Geração distribuída

O consumo mundial de energia elétrica vem crescendo a cada ano, como ilustrado na Figura 1, com aumento de 3,5 % em 2018.

Figura 1 – Suprimento mundial de energia por fonte ao longo dos anos



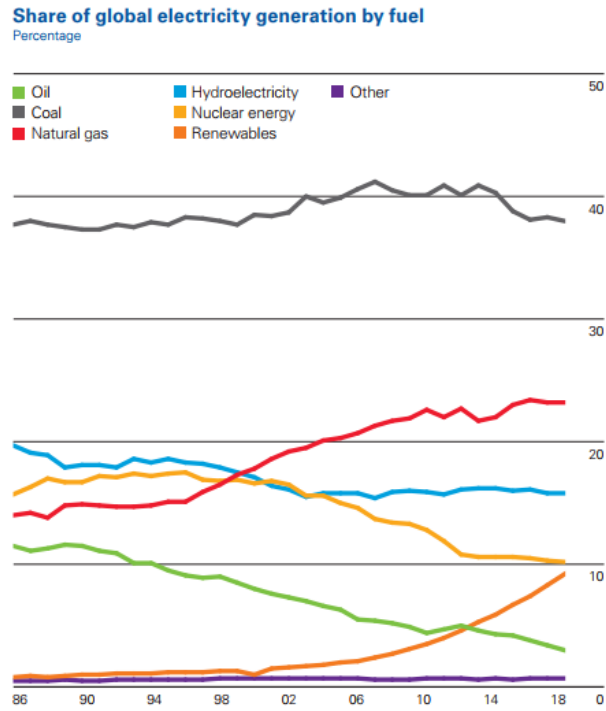
Fonte: (ENERDATA, 2019)

Em nível global, o carvão é o combustível dominante na geração de energia elétrica, representando 38 % (2018), a mesma parcela de 20 anos atrás. O gás é o segundo combustível mais usado com uma quota de 23,2 % (2018), como mostra a Figura 2.

Devido à preocupação com a redução das reservas de fontes de energia não renovável e com a emissão de gases de efeito estufa, tem-se intensificado o interesse pela diversificação da matriz energética mundial, aumentando a participação de fontes de energia renováveis e com menor impacto ambiental, como é o caso das fontes de energia solar, eólica etc.

Como pode ser observado na Figura 3, as energias renováveis na geração de eletricidade

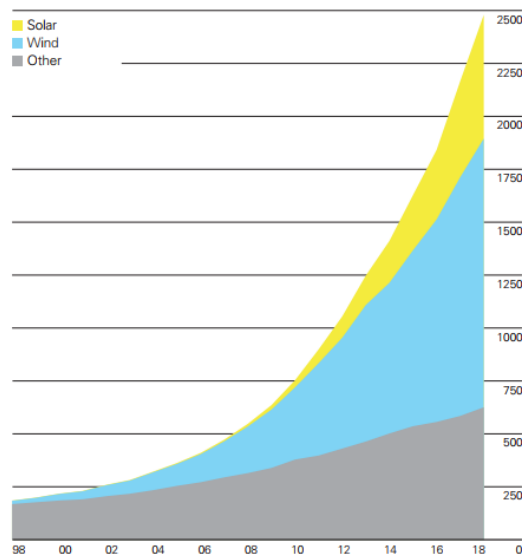
Figura 2 – Produção de energia elétrica por tipo de fonte de 1986 a 2018



Fonte: (BRITISH PETROLEUM, 2019)

dade (excluindo a hidrelétrica) aumentaram 14 % em 2018, um pouco abaixo do crescimento médio de 10 anos (16 %). A energia eólica (142 TWh) contribuiu mais para o crescimento da geração renovável do que a solar (131 TWh). A energia eólica foi responsável por cerca de 50 % da geração de energias renováveis nos últimos anos. A solar aumentou constantemente sua

Figura 3 – Geração de energia a partir de fontes renováveis de 1998 a 2018



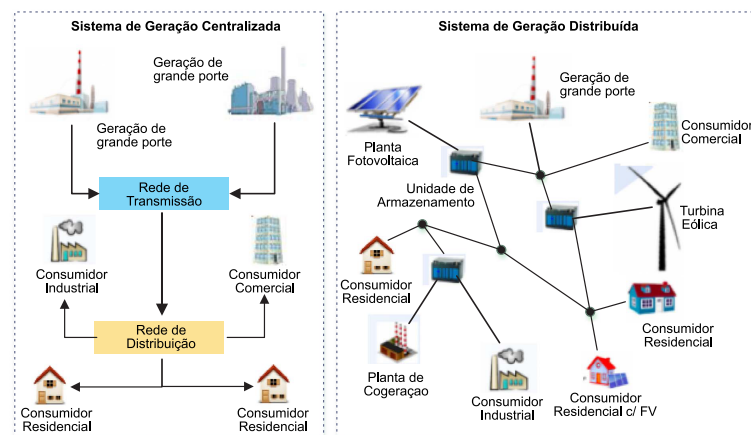
Fonte: (BRITISH PETROLEUM, 2019)

participação e agora representa 24 %.

A integração de fontes eólica e solar nos sistemas de distribuição de energia elétrica originou um novo conceito de geração, denominado GD. A GD é caracterizada pela conexão de pequenas unidades geradoras de energia ao longo da rede de distribuição e próxima dos consumidores, diferentemente do que acontece na geração convencional, em que são alocadas grandes unidades de geração distantes dos grandes centros de carga (COLMENAR-SANTOS *et al.*, 2016).

Na Figura 4 é ilustrada a diferença entre as redes elétricas com geração distribuída e com geração convencional centralizada. Na geração convencional, a energia é transportada através dos sistemas de transmissão e distribuição até chegar aos consumidores finais. Na geração distribuída, unidades de geração e armazenamento de energia são localizadas próximas as cargas.

Figura 4 – Comparação entre os modelos de geração distribuída e centralizada.



Fonte: Adaptado de Ehsan e Yang (2018)

A difusão de GD no sistema elétrico proporciona vantagens técnicas, econômicas e ambientais (COLMENAR-SANTOS *et al.*, 2016), tais como:

- Redução de perdas técnicas, melhora no perfil de tensão, melhora na qualidade da energia elétrica, melhora na eficiência energética e melhora na confiabilidade do sistema elétrico;
- Redução de custos operacionais, redução de custos de capital, redução de custos ambientais e redução de tarifas de consumo de energia elétrica;
- Redução do consumo de combustíveis fósseis e redução de emissão de gases do efeito estufa.

As redes elétricas de distribuição, que antes tinham característica passiva em relação ao sistema elétrico, com a presença da GD passam a apresentar característica ativa, com fluxo de

potência bidirecional e a ser denominadas de Redes de Distribuição Ativas (RDA).

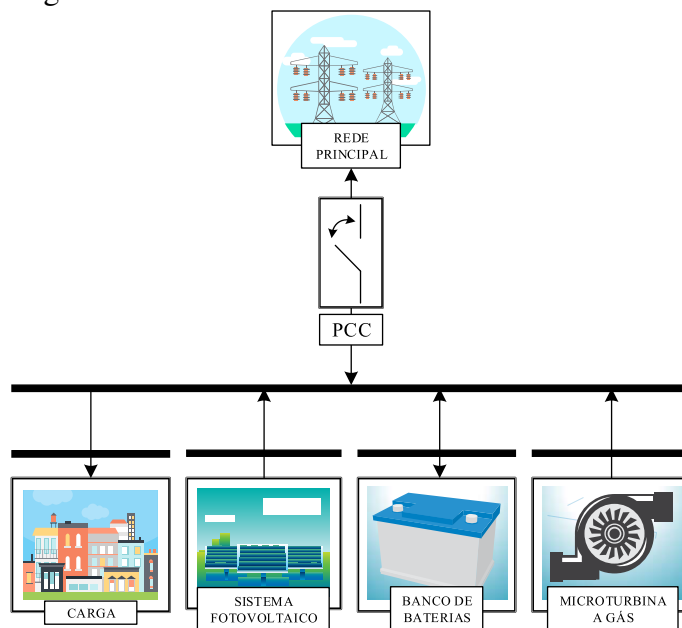
Segundo (HATZIARGYRIOU, 2014), a concretização das RDAs requer a implementação de novos conceitos, sendo um dos mais promissores o conceito de microrredes.

2.3 Microrredes

Microrredes são redes de distribuição em baixa ou média tensão com geração distribuída, unidades de armazenamento de energia e cargas flexíveis. Tais sistemas podem operar de maneira não autônoma, se conectadas à rede elétrica principal, ou de maneira autônoma, se desconectadas da rede elétrica principal. A operação de microrredes pode proporcionar diversos benefícios aos consumidores da microrrede e ao sistema elétrico como um todo, se gerenciada e controlada eficientemente (HATZIARGYRIOU, 2014).

Em geral, as GDs empregadas nas microrredes são baseadas em fontes de energia renováveis, não estocáveis e de natureza intermitente, como é o caso da energia solar fotovoltaica e da eólica. No entanto, com o objetivo de aumentar a confiabilidade do fornecimento de energia também são empregadas fontes despacháveis, como microturbina a gás e geradores a óleo combustível. Além disso, são empregados elementos armazenadores de energia, como é o caso do banco de baterias, agregando maior flexibilidade à operação do sistema (ALMEIDA, 2019). Na Figura 5 é apresentada a estrutura básica de uma microrrede.

Figura 5 – Estrutura básica de uma microrrede.



Fonte: (ALMEIDA, 2019)

Portanto, uma microrrede é essencialmente uma RDA, uma vez que congrega unidades de geração e cargas a nível de distribuição (CHOWDHURY *et al.*, 2009). No entanto, as microrredes diferenciam-se pela sua capacidade de operar em modo isolado da rede elétrica e de controlar os recursos energéticos disponíveis (HATZIARGYRIOU, 2014).

As microrredes representam uma mudança no paradigma de operação da GD. Do ponto de vista da rede principal, a microrrede é vista como uma unidade controlável, que pode ser acionada para atender às demandas e prestar serviços ancilares às redes de distribuição. As microrredes podem fornecer: aumento da confiabilidade do suprimento de energia elétrica; redução de perdas de transmissão de energia; melhora do perfil de tensão de fornecimento; correção de afundamentos de tensão, entre outras funcionalidades (LASSETER, 2002).

2.4 Proteção de microrredes

Um dos principais desafios para implementação de microrredes é o projeto do seu sistema de proteção (HATZIARGYRIOU, 2014). A presença de GDs, a mudança topológica da rede e o ilhamento alteram a magnitude e a direção das correntes de falta. Além disso, os tipos de geração distribuída implementados também afetam o sistema de proteção. Fontes de geração baseadas em máquinas rotativas diretamente conectadas à rede elétrica aumentam os níveis de curto-circuito. Por outro lado, fontes de geração interfaceadas através de conversores eletrônicos apresentam baixas contribuições de falta, afetando a sensibilidade do sistema de proteção (HATZIARGYRIOU, 2014).

Em geral, os desafios do sistema de proteção de microrredes podem ser resumidos em (HATZIARGYRIOU, 2014):

- Mudança na magnitude e direção das correntes de curto-circuito, dependendo se a geração está conectada ou não;
- Redução da sensibilidade e velocidade do sistema de proteção, devido à conexão de fontes de GD;
- Atuação intempestiva do disjuntor da distribuidora para faltas em linhas adjacentes, devido a contribuição das GDs;
- Maiores custos associados ao sistema de proteção, devido ao aumento nos níveis de curto-circuito;
- Falha na operação de religadores e fusíveis devido a contribuição das GDs;
- Reduzidas contribuições de falta em redes com fontes de geração interfaceadas por inversor,

especialmente quando a rede está operando em modo isolado.

Quatro tipos de proteção são citadas em (HATZIARGYRIOU, 2014) para mitigação destes impactos no sistema de proteção das microrredes:

- **Proteção de distância (ANSI 21)**. A proteção de distância calcula a impedância entre o relé e a falta relacionando a corrente de falta e a tensão medidos pelo relé. Se a impedância medida for inferior à impedância parametrizada no relé, significa que a falta está dentro da zona de proteção do relé. Portanto, a proteção de distância pode ser afetada pela presença de GDs, uma vez que a impedância calculada pelo relé está relacionada à tensão e às correntes de falta.
- **Proteção diferencial (ANSI 87)**. A proteção diferencial é aplicada para proteger equipamentos importantes, tais como: geradores, transformadores, barramentos e linhas de transmissão. Este tipo de proteção baseia-se na diferença das correntes que entram e que saem dos terminais do equipamento, operando apenas para faltas internas. No entanto, quando aplicado à proteção de linhas, é necessário um sistema de comunicação, devido à distância entre os terminais da linha. A necessidade de um sistema de comunicação torna este tipo de proteção mais caro que os demais, devido à exigência de esquemas de redundância.
- **Proteção de sobrecorrente (ANSI 51)**. É o tipo de proteção mais aplicado em redes de distribuição. Tradicionalmente é aplicada a redes de distribuição com fluxo unidirecional de corrente de falta.
- **Proteção de sobrecorrente direcional (ANSI 67)** Em redes malhadas, com topologia em anel ou com presença de GDs é necessária a aplicação da proteção de sobrecorrente direcional.

Alguns trabalhos propõem SPAs aplicados à proteção de sobrecorrente, visando a redução de custos, a facilidade de implementação e uso difundido deste tipo de proteção (LAAKSONEN *et al.*, 2014; COFFELE *et al.*, 2015; SINGH *et al.*, 2016; OJAGHI; MOHAMMADI, 2018; JAIN *et al.*, 2019). No entanto, a presença de geração distribuída nas redes de distribuição requer a capacidade do sistema de proteção de detectar a direção das correntes de falta. Portanto, SPAs aplicados a mudança de ajustes dos relés de sobrecorrente direcional vêm sendo pesquisados nos últimos anos (CHATTOPADHYAY B.; SACHDEV, 1996; MAHAT *et al.*, 2011; PAPASPILIOTOPOULOS *et al.*, 2017; LIU *et al.*, 2017; ALAM, 2018).

Visando reduzir os impactos causados pela operação das microrredes e a geração

distribuída no sistema de proteção, neste trabalho é proposto um sistema de proteção adaptativa aplicado à proteção de sobrecorrente direcional.

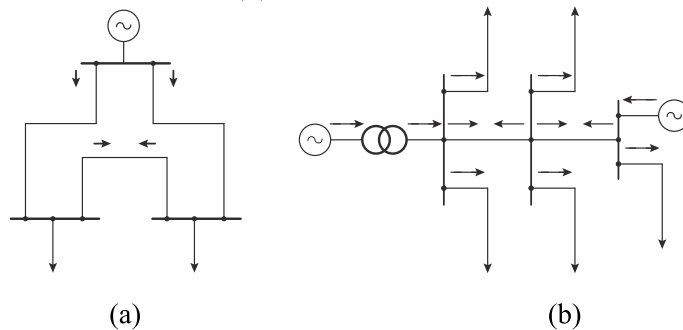
2.5 Proteção de sobrecorrente direcional

A proteção de sobrecorrente direcional é aplicada para proteger sistemas elétricos, em que as correntes de falta podem circular em ambas as direções. Esta proteção é tipicamente aplicada em sistemas elétricos malhados ou com topologia anel (GERS; HOLMES, 2011).

Tradicionalmente, as redes de distribuição, por apresentarem característica de fluxo de corrente unidirecional, requeriam a implementação da proteção de sobrecorrente tradicional. Com o aumento da penetração de GDs, a proteção de sobrecorrente direcional passou também a ser aplicada nas redes de distribuição.

Na Figura 6 é apresentada a característica de bidirecionalidade do fluxo em um sistema em anel (a) e em uma rede de distribuição com GD (b).

Figura 6 – Fluxo bidirecional em rede em anel (a) e em rede de distribuição com geração distribuída (b)



Fonte: Adaptado de (GERS; HOLMES, 2011).

A seguir são apresentados maiores detalhes sobre o relé de sobrecorrente direcional.

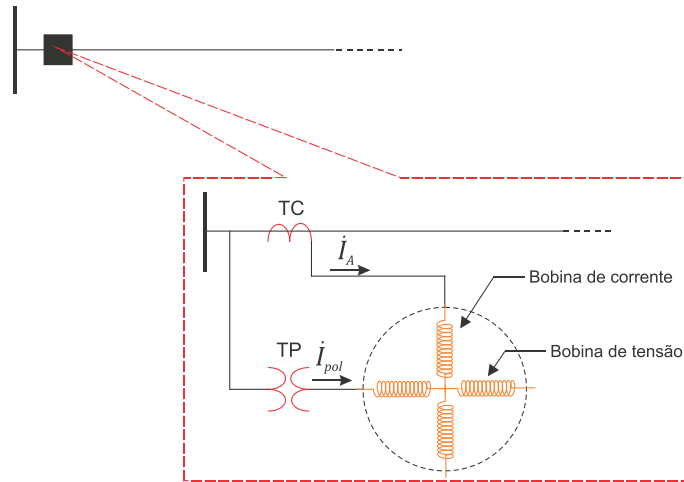
2.5.1 Princípio de funcionamento

O relé de sobrecorrente direcional atua para correntes de falta em uma direção específica pré determinada (KINDERMANN, 2005). Para detectar a direção do fluxo da corrente, o relé faz uma comparação fasorial entre a corrente de operação e a tensão de operação. A defasagem entre estas duas grandezas fornece a informação sobre a direção do fluxo (KINDERMANN, 2005). Logo, o relé de proteção de sobrecorrente direcional necessita de dois transformadores de instrumentação, o Transformador de Corrente (TC) e o Transformador de Potencial (TP), para

medir a corrente e a tensão no ponto de instalação, respectivamente. Na Figura 7 é apresentado o esquema de ligação dos relés de sobrecorrente direcionais.

Na Figura 7, as bobinas de corrente e tensão estão em quadratura. A corrente que entra nos terminais da bobina de corrente é a corrente de linha da fase A (I_A), refletida no secundário do TC. Já a corrente que entra na bobina de tensão é a corrente no secundário do TP proveniente da tensão de fase medida, neste caso a tensão entre as fases B e C (V_{BC}).

Figura 7 – Representação unifilar de um relé de sobrecorrente direcional



Fonte: (BOTTURA, 2014)

O diagrama fasorial das grandezas envolvidas no relé direcional de sobrecorrente é apresentado na Figura 8, em que \dot{I}_A é a corrente da fase A, ϕ_A é o fluxo magnético devido à corrente \dot{I}_A , V_{AN} é a tensão da fase A em relação ao neutro, V_{pol} é a tensão de polarização aplicada na bobina de tensão (neste caso a tensão V_{BC}), \dot{I}_{pol} é a corrente de polarização devido à aplicação de V_{pol} e ϕ_{pol} é o fluxo magnético devido à corrente de polarização \dot{I}_{pol} . Com relação aos ângulos, θ é o ângulo da corrente \dot{I}_A (adotando V_{BC} como o fasor de referência), α é o ângulo entre os fluxos magnéticos e γ é o ângulo de máximo conjugado.

O conjugado provocado pela interação entre os fluxos é dado pela Equação (2.1) (KINDERMANN, 2005).

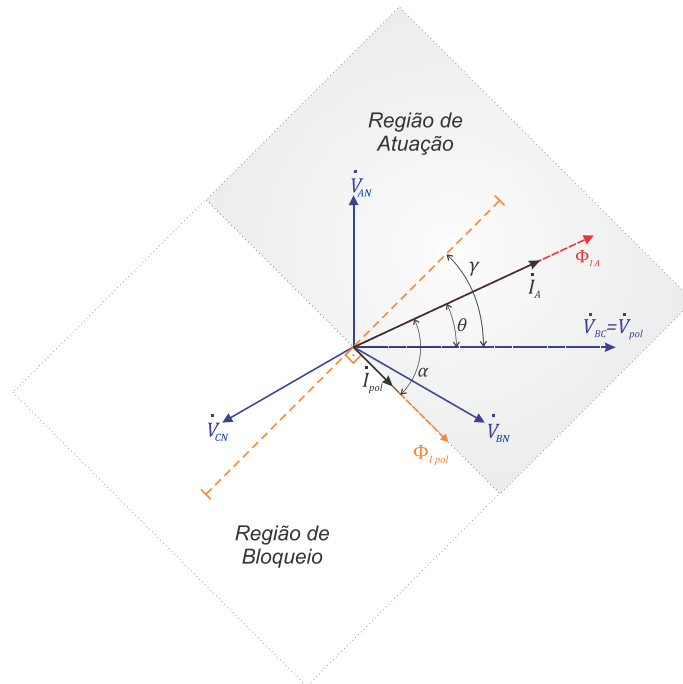
$$\tau = K\phi_A\phi_{pol}\sin(\alpha) \quad (2.1)$$

em que K é uma constante de proporcionalidade.

Pela análise do diagrama, nota-se que $\alpha = 90 - (\gamma - \theta)$. Portanto:

$$\tau = K\phi_A\phi_{pol}\cos(\gamma - \theta) \quad (2.2)$$

Figura 8 – Diagrama fasorial das grandezas envolvidas no relé de sobrecorrente direcional.



Fonte: (BOTTURA, 2014)

Da Equação (2.2), conclui-se que o torque é positivo (atuação do relé) quando $\theta = \gamma \pm 90^\circ$, exatamente a área delimitada pela cor cinza no diagrama fasorial. A região em que o torque assume valores negativos (não atuação) é a região destacada pela cor branca.

No exemplo apresentado, o ângulo de polarização (defasagem entre a corrente no TC e a tensão no TP) foi de 90° . Existem outros esquemas de conexão, que alteram o ângulo de polarização. Este ângulo afeta a região de atuação do relé, deslocando-a ao longo do plano. Aplicações específicas exigem ângulos de polarização distintos, como 30° e 60° , por exemplo (KINDERMANN, 2005). Para proteção de alimentadores, comumente adota-se ângulo de polarização em 90° .

Os relés digitais atuais utilizam o mesmo princípio para detecção da direção da corrente, no entanto a análise de defasagem é feita utilizando o sistema de controle lógico interno do equipamento. Não é necessário alterar as ligações dos TPs e TCs para alterar o ângulo de polarização.

2.5.2 Curva de operação

Os relés de sobrecorrente direcional operam com base na curva de tempo inverso, em que o tempo de operação é inversamente proporcional à corrente medida. Esta curva é

padronizada pela norma IEC 60255-3 (IEC, 1989) e segue o modelo da equação (2.3).

$$t_{op}(I) = TDS \frac{\beta}{\left(\frac{I}{PS}\right)^\alpha - 1} \quad (2.3)$$

Em que t_{op} é o tempo de operação do relé; TDS é o *Time Dial Setting*, também denominado *Dial*, é uma constante de multiplicação; PS é a corrente de *pickup*, que é a menor corrente a qual o relé começa a operar; I é a corrente medida pelos TCs de medição; e α e β são constantes que definem a característica da curva.

Os relés digitais permitem que o projetista selecione o tipo de curva de operação padrão para cada relé. Na Tabela 3 são apresentados os valores de α e β associados a cada tipo de curva.

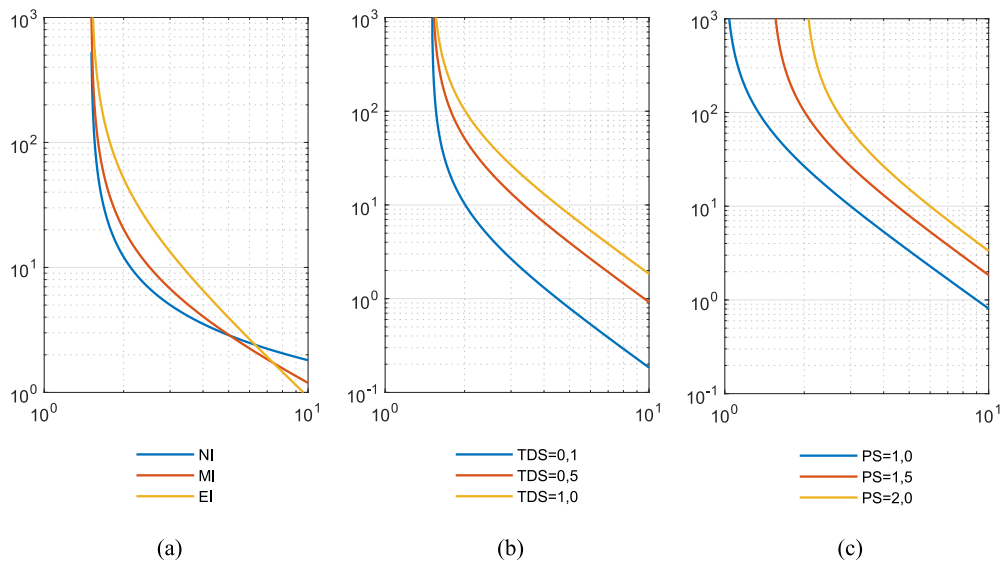
Tabela 3 – Constantes das curvas de operação definidas pela IEC 60255-3.

Tipo de Curva	α	β
Normalmente Inversa (NI)	0,14	0,02
Muito Inversa (MI)	13,5	1,0
Extremamente Inversa (EI)	80	2

Fonte: O próprio autor.

Na Figura 9 são apresentados o comportamento das curvas a partir da mudança do tipo de curva (a), do TDS (b) e do PS (c), mantendo os demais parâmetros constantes.

Figura 9 – Efeito do tipo de curva (a), do PS (b) e TDS (c) no comportamento da curva de operação.



Fonte: O próprio autor.

Cada tipo de curva apresenta uma inclinação distinta, que afeta a variação do tempo

de operação em função do aumento da corrente, sendo a curva normalmente inversa a menos inclinada e a curva extramente inversa a mais inclinada.

A variação do *TDS* também afeta a inclinação da curva. Quanto maior o valor de *TDS*, maior o valor do tempo de operação para um mesmo valor de corrente e, conseqüentemente, menor a inclinação da curva.

Matematicamente, o valor de *PS* representa uma assíntota vertical da curva de tempo inverso, valores de corrente inferiores a *PS* resultariam em tempos de operação negativos, portanto, impossíveis. Portanto, o aumento de *PS* causa um deslocamento horizontal da curva de tempo inverso dos relés.

2.5.3 Coordenação de relés direcionais de sobrecorrente

Qualquer sistema de proteção deve atender os seguintes requisitos (GERS; HOLMES, 2011):

- Confiabilidade: Capacidade do sistema de proteção de operar corretamente.
- Sensibilidade: Capacidade do sistema de proteção de sempre estar sensível à sua atuação.
- Segurança: Capacidade do sistema de proteção de evitar operação incorreta durante uma falta.
- Velocidade: Capacidade do sistema de proteção de atuar no menor tempo possível, de modo a evitar maiores danos, devido à exposição à corrente de falta.
- Seletividade: Capacidade do sistema de proteção de manter a continuidade do fornecimento de energia, desconectando a menor quantidade de carga possível e necessária para isolar a falta.
- Custo: Capacidade do sistema de proteção de oferecer a maior proteção necessária pelo menor custo possível.

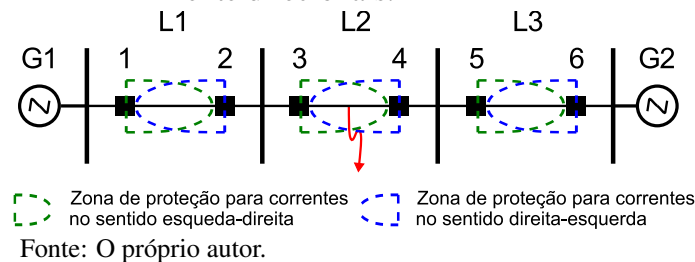
Todo relé do sistema elétrico protege uma região específica, denominada de zona de proteção. Para atender ao critério de seletividade, sempre que houver uma falta dentro da zona de proteção de um relé, este deve ser o primeiro relé a atuar. Neste caso, o relé faz o papel de proteção principal da zona de proteção.

Para atender também ao critério de confiabilidade, é necessário que exista um relé de retaguarda (ou *backup*), que atue o mais rápido possível, caso o relé de proteção principal falhe na sua operação. Deste modo, para uma falta, sempre existe, no mínimo, um par de relés principal/*backup*.

No caso da proteção de sobrecorrente direcional, dois relés, sensibilizados em direções de corrente opostas, são dispostos de modo a proteger linhas de transmissão e distribuição dos sistemas de energia elétrica com múltiplas unidades de geração. Desta forma, ambos os relés fazem o papel de proteção principal da linha.

Conforme é ilustrado na Figura 10, cada linha de distribuição possui dois relés de proteção principal, com zonas de proteção em sentidos opostos, relés: 1 e 2 protegem a linha L1; 3 e 4 protegem a linha L2; 5 e 6 protegem a linha L3.

Figura 10 – Zonas de proteção dos relés de sobrecorrente direcionais.



Quanto a proteção de retaguarda, apenas os relés configurados para a mesma direção de falta podem formar um par de relés principal/*backup*. Na rede da Figura 10, para uma falta na linha L2, o relé 1 faz o papel de proteção de *backup* do relé 3 (proteção principal), enquanto o relé 6 faz o papel de proteção de *backup* do relé 4 (proteção principal).

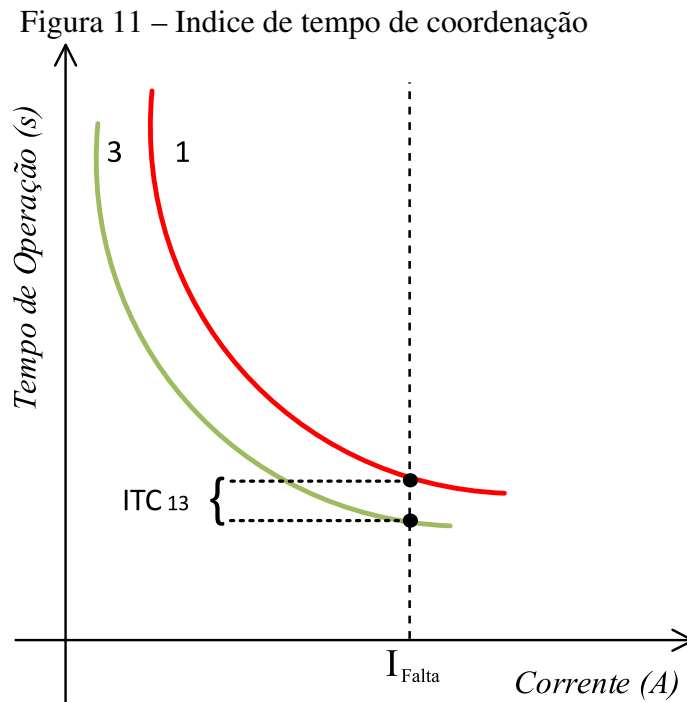
A proteção de retaguarda deve atuar o mais rápido possível, no entanto é necessário que esta seja mais lenta que a proteção principal. Caso contrário, em caso de falta, ambos os relés (principal e *backup*) atuariam, cargas sãs seriam desenergizadas indevidamente e o sistema de proteção deixaria de ser seletivo.

Quando os tempos de operação dos pares de relés principal/*backup* respeitam este princípio, diz-se que o sistema de proteção está coordenado. Para alcançar a coordenação, o tempo de operação do relé de retaguarda é calculado segundo a Equação (2.4).

$$t_r \geq t_p + ITC \quad (2.4)$$

Em que t_r é o tempo da operação da proteção de retaguarda, t_p é o tempo de operação da proteção principal e ITC é o índice de tempo de coordenação. O ITC é escolhido de modo a garantir que o relé de retaguarda atue mais lentamente e leva em conta o tempo de abertura do disjuntor, o tempo para extinção do arco elétrico e demais possíveis atrasos de operação (KINDERMANN, 2005). Tipicamente, o ITC é adotado entre 0,2 e 0,5 segundos (ANDERSON, 1999).

Na Figura 11 é possível analisar graficamente o papel do *ITC* nos tempos de coordenação dos relés 1 e 3 da Figura 10. Para a falta especificada, o relé 3 opera mais rápido e o relé 1 é projetado para operar *ITC* segundos mais lento.



Fonte: O próprio autor.

No processo de coordenação dos relés de sobrecorrente direcionais, os ajustes *PS* e *TDS* de cada relé devem ser escolhidos de modo a garantir a coordenação do sistema de proteção para todas as faltas no sistema elétrico. No problema de coordenação ótima de relés direcionais de sobrecorrente, os ajustes são escolhidos de modo a tornar o sistema de proteção o mais rápido possível (critério de velocidade) e coordenado (critério de seletividade).

2.6 Considerações finais

Neste Capítulo foram apresentados os principais conceitos que embasam o desenvolvimento deste trabalho. Em seguida, foi apresentada a revisão bibliográfica acerca dos trabalhos relacionados e identificadas as lacunas da literatura a serem preenchidas pelo presente trabalho.

Visando propor um novo algoritmo para cálculo de ajustes de relés direcionais de sobrecorrente direcionais em aplicações *online* de proteção adaptativa, este trabalho propõe uma nova metaheurística, o Algoritmo do Morcego Direcional *Fuzzy* Adaptativo (AMDFA).

3 COORDENAÇÃO OTIMIZADA DE RELÉS DIRECIONAIS DE SOBRECORRENTE

O objetivo deste Capítulo é apresentar a formulação matemática do problema de coordenação ótima de relés de sobrecorrete direcionais, apresentar conceitos introdutórios acerca de otimização metaheurística e sistemas de inferência *fuzzy*, apresentar a metaheurística proposta e os resultados da validação do algoritmo em três redes de referência.

3.1 Formulação do problema de coordenação ótima de relés direcionais de sobrecorrente

Esta seção tem como objetivo apresentar a formulação matemática adotada para o problema de CORSD. As restrições e a função objetivos adotadas são apresentadas nos tópicos seguintes.

3.1.1 Restrições

3.1.1.1 Restrição de coordenação

A primeira restrição do problema de coordenação de relés direcionais de sobrecorrente é a restrição de coordenação. Como já abordado anteriormente, os relés de proteção de retaguarda devem atuar mais lentamente em relação ao seu respectivo relé de proteção principal por um fator de *ITC* segundos. Formalizando matematicamente, a definição de *ITC* é dada pela Equação (3.1)

$$ITC = t_{jk} - t_{ik} \quad (3.1)$$

Em que *ITC* é o índice de tempo de coordenação, t_{ik} é o tempo de operação do relé principal *i* para uma falta *k* e t_{jk} é o tempo de operação do relé de *backup* *j* para a mesma falta *k*. Logo, a restrição de coordenação entre os relés é dada pela Equação (3.2).

$$\Delta t_{ijk} = ITC - ITC_{min} \geq 0 \quad (3.2)$$

Em que Δt_{ijk} é a diferença entre o *ITC* dos relés *i* e *j* e o ITC_{min} que é o valor mínimo de *ITC* aceitável para o sistema de proteção, adotado pelo projetista.

3.1.2 Restrição da escolha da corrente de pickup

O segundo tipo de restrição é relacionado à escolha dos ajustes. Para escolha da corrente de *pickup* devem ser levados em conta tanto os limites impostos pelo equipamento, quanto questões operativas do sistema elétrico em si.

Como a corrente de *pickup* é a menor corrente para a qual o relé inicia a contagem do tempo de operação, esta deve ser superior à maior corrente de carga medida pelo relé. Caso contrário, em situação normal de operação, o relé pode operar indevidamente apenas com a corrente normal de carga.

Em sistemas com alta impedância de falta ou com predominância de fontes de geração interfaceadas por inversor, as correntes de falta tendem a assumir valores muito baixos, muitas vezes na ordem de grandeza da corrente de carga. Desta forma, a corrente de *pickup* também não deve ser muito elevada, caso contrário o relé de proteção ficará insensível para estes casos de falta.

Portanto, a variável PS deve seguir a restrição dada pela Equação (3.3):

$$PS_{min} \leq PS_i \leq PS_{max} \quad (3.3)$$

Em que PS_i é a corrente de *pickup* do relé i , PS_{min} é o valor mínimo admitido para PS_i e PS_{max} é o valor máximo admitido para PS_i .

Desta forma, PS_{min} e PS_{max} foram adotados segundo as equações (3.4) e (3.5), respectivamente.

$$PS_{min} = \max(PS_{min}^{eq}; K1I_{max}^L) \quad (3.4)$$

$$PS_{max} = \min(PS_{max}^{eq}; K2I_{min}^F) \quad (3.5)$$

Em que PS_{min}^{eq} e PS_{max}^{eq} são os valores máximos e mínimos permitidos pelo equipamento, I_{max}^L é a corrente de carga máxima que passa pelo relé (obtido a partir de estudos de fluxo de carga) e I_{min}^F é a menor corrente de falta que passa pelo relé em todas as situações de falta (obtida a partir de estudos de curto circuito). Tanto I_{max}^L como I_{min}^F são multiplicadas por fatores de correção, $K1 > 1$ e $0 < K2 < 1$, para garantir que a corrente de *pickup* obedecerá às restrições necessárias.

3.1.2.1 Restrição da escolha do multiplicador TDS

Para a escolha do TDS são adotadas apenas as limitações impostas pelo próprio relé de proteção. A Equação (3.6) modela matematicamente a restrição do ajuste de TDS .

$$TDS_{min} \leq TDS_i \leq TDS_{max} \quad (3.6)$$

Em que TDS_i é o TDS do relé i , TDS_{min} e TDS_{max} são os valores máximos e mínimos permitidos para TDS_i .

3.1.3 Função objetivo

O objetivo do problema de coordenação ótima de relés direcionais de sobrecorrente é minimizar o tempo de operação do sistema de proteção como um todo, obedecendo às restrições de coordenação e escolha de ajustes dos relés de proteção.

As variáveis do problema de otimização são os ajustes PS e o TDS de cada relé. A curva de operação foi adotada como igual para todos os relés para garantir menor complexidade ao problema. Portanto, a função objetivo adotada neste trabalho é modelada segundo a Equação (3.7).

$$FOB = \sum_{k=1}^F \sum_{i=1}^N \sum_{j=1}^{M_i} ((t_{ik}^2 + t_{jk}^2) + \gamma(\Delta t_{ijk} - |\Delta t_{ijk}|)^2) \quad (3.7)$$

Em que F é o número de faltas consideradas, N é o número de relés e M_i é o número de relés de *backup* do relé i . Logo, t_{ik} é o tempo de operação do relé i para uma falta k , t_{jk} é o tempo de operação do relé de *backup* j para a falta k e Δt_{ijk} é a diferença entre CTI e CTI_{min} .

A primeira parcela do somatório, a soma dos quadrados dos tempos de operação dos relés de proteção principal e *backup*, expressa matematicamente o objetivo do problema de minimizar o tempo total do sistema de proteção. Os quadrado dos tempos de operação é adotado, ao invés de simplesmente os tempos de operação, para evitar que o algoritmo minimize o tempo total de operação adotando tempos de operação negativos para os relés.

A segunda parcela do somatório é uma função de penalidade, que tem a finalidade de incorporar a restrição de coordenação na função objetivo do problema de otimização. Se $\Delta t_{ijk} > 0$, ou seja, o par de relés principal/*backup* está coordenado, a segunda parcela do somatório será nula, portanto não há penalidade. Caso contrário, se $\Delta t_{ijk} < 0$, há descoordenação e a segunda parcela do somatório será igual a $4\gamma(\Delta t_{ijk})^2$. Logo, a segunda parcela será não nula e proporcional

ao quadrado da descoordenação, configurando uma penalidade. A constante γ é um peso dado à função de penalidade e foi adotado como igual a 100 nos testes deste trabalho.

Vale ressaltar que penalizar a função objetivo em casos de descoordenação não garante que não existirá descoordenação no sistema de proteção. O objetivo da função de penalidade é fazer com que não hajam descoordenações, mas se houverem, que estas sejam as menores possíveis. Esta abordagem não inviabiliza a modelagem, uma vez que, se a descoordenação obtida for baixa, na prática esta diferença terá pouca influência. Além disso, esta abordagem permite que o algoritmo encontre ajustes que proveem maior rapidez ao sistema de proteção, porém com *ITCs* pouco abaixo do limite mínimo.

Por esse motivo, o ideal é que o algoritmo de otimização aplicado seja capaz de prover ajustes capazes de minimizar o tempo total de operação, mas com baixos índices de descoordenação do sistema de proteção.

3.2 Otimização Metaheurística

O sistema de proteção adaptativa proposto neste trabalho utiliza uma metaheurística para resolver o problema de coordenação ótima de relés de sobrecorrente direcionais. Portanto, esta seção aborda conceitos básicos de otimização metaheurística, com o objetivo de embasar melhor a proposição da metaheurística aplicada.

O cálculo de soluções ótimas para problemas de otimização muitas vezes pode se tornar inviável devido ao alto custo computacional envolvido. Na prática, soluções subótimas já provêm resultados suficientemente bons para determinadas aplicações (TALBI, 2009).

No caso do problema de coordenação ótima de relés de sobrecorrente direcional, o tempo total (somatório de todos os tempos de operação) de operação ótimo pode diferir de um tempo total subótimo em poucos segundos. No entanto, encontrar o tempo total ótimo pode implicar em um custo computacional elevado, devido à complexidade do problema. Portanto, o ganho de velocidade do sistema de proteção pode não compensar a perda em confiabilidade do sistema, causado pela vulnerabilidade do sistema de proteção com ajustes não adequados, enquanto o SPA procura os ajustes ótimos.

Metaheurísticas são algoritmos utilizados para encontrar soluções suficientemente boas (subótimas) em um tempo hábil para problemas de otimização. Diferentemente dos algoritmos de otimização determinísticos, as metaheurísticas não garantem que a solução encontrada é a ótima e, diferentemente dos algoritmos aproximados, não definem o quão próximo a solução

obtida está da ótima. No entanto, as soluções providas pelas metaheurísticas possuem uma boa probabilidade de estarem próximas da ótima e proverem soluções suficientemente boas. Devido a isso, a aplicação de metaheurísticas para solução de problemas de engenharia e ciência ganhou grande notoriedade nas últimas duas décadas (TALBI, 2009).

No projeto de metaheurísticas, dois critérios contraditórios devem ser levados em conta: diversificação da busca pelo espaço de soluções e intensificação da busca em torno das melhores soluções encontradas. Regiões promissoras do espaço de busca são determinadas pelas melhores soluções encontradas. No processo de intensificação, as regiões promissoras são exploradas mais intensamente para descobrir melhores soluções na vizinhança. Já no processo de diversificação, o objetivo é explorar a maior parte possível do espaço de soluções para garantir que o processo de busca não seja realizado em apenas uma parte reduzida do espaço de soluções (TALBI, 2009).

Para uma metaheurísticas operar corretamente, é necessário que exista um balanço entre os dois critérios. Caso contrário, se o algoritmo estiver com característica mais forte de diversificação, o algoritmo tende a demorar mais para convergir para uma solução. Por outro lado, se estiver com característica mais forte de intensificação, o algoritmo converge mais rápido, porém tende a convergir prematuramente para mínimos locais. O ideal é que o algoritmo possua característica de diversificação mais forte nos estágios iniciais, em que não se conhece o espaço de soluções, e gradativamente passe para uma característica de intensificação nas iterações finais, quando o algoritmo provavelmente já encontrou soluções suficientemente boas e precisa explorar as suas vizinhanças para encontrar soluções melhores.

3.3 Sistemas de inferência fuzzy

A metaheurística proposta neste trabalho aplica um sistema de inferência *fuzzy* para calcular os seus parâmetros em tempo de execução. Esta seção visa introduzir conceitos básicos acerca da lógica *fuzzy* e dos sistemas de inferência *fuzzy*.

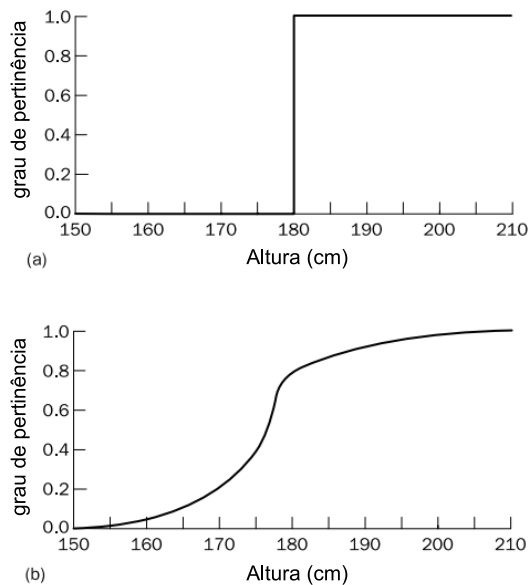
Em 1965, Lofti Zadeh propôs a teoria dos conjuntos *fuzzy* como uma variação da teoria clássica dos conjuntos (ZADEH, 1965). Na teoria clássica dos conjuntos, o grau de pertinência de um elemento só pode assumir dois valores: 0 (quando o elemento não pertence a um dado conjunto) ou 1 (quando o elemento pertence a um dado conjunto), portanto, a pertinência nessa teoria é um domínio discreto.

Na teoria dos conjuntos *fuzzy*, a pertinência de um elemento pode assumir qualquer

valor dentro do domínio contínuo $[0,1]$. Inclusive, um elemento pode apresentar grau de pertinência diferente de zero em mais de um conjunto ao mesmo tempo.

Para ilustrar melhor esse conceito, na Figura 12 é apresentado o que seria a representação do conjunto das pessoas altas em um conjunto *crisp* (como são denominados os conjuntos da teoria clássica) e em um conjunto *fuzzy*.

Figura 12 – Conjuntos crisp (a) e fuzzy (b) de pessoas altas



Fonte: Adaptado de (NEGNEVITSKY, 2011)

No conjunto *crisp* das pessoas altas, pessoas com altura inferior a 180 cm possuem pertinência igual a 0 (não pertencem ao grupo das pessoas altas) e pessoas com altura superior a 180 cm possuem pertinência igual a 1 (pertencem ao grupo das pessoas altas). Já no conjunto *fuzzy* das pessoas altas, existe uma função de pertinência que define um valor de pertinência para cada valor no domínio das alturas. Nesse exemplo, fica claro como conceitos abstratos e subjetivos, como é o caso do conceito de uma pessoa "Alta", é melhor representado na teoria dos conjuntos *fuzzy*. Por exemplo, na teoria clássica dos conjuntos, uma pessoa com 1,79 metros de altura teria pertinência nula ao conjunto das pessoas altas. Já no conjunto *fuzzy* esta pessoa teria uma pertinência diferente de zero.

As variáveis na teoria dos conjuntos *fuzzy* são variáveis linguísticas. Por exemplo, a variável linguística "Alto" utilizada no exemplo anterior. Na lógica *fuzzy*, as variáveis linguísticas são empregadas em regras *fuzzy* do tipo SE-ENTÃO. Por exemplo: "Se X é alto, então X é feio", sendo "feio" outra variável *fuzzy*. Assim como na teoria clássica dos conjuntos, a teoria dos conjuntos *fuzzy* possui operações lógicas de implicação, que definem a relação entre a regra

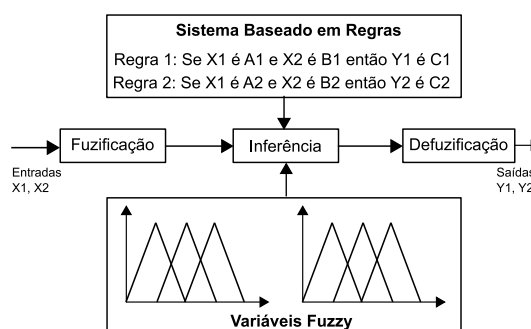
antecedente ("se X é alto") e a regra consequente ("então X é feio") (NEGNEVITSKY, 2011).

Um conjunto de regras SE-ENTÃO do tipo *fuzzy* podem ser combinadas para formar um Sistema de Inferência *Fuzzy* (SIF), que pode ser definido como um processo que mapeia um conjunto de entradas em um conjunto de saídas, utilizando a teoria dos conjuntos *fuzzy*.

Em geral, os SIF podem ser de dois tipos: Mandani ou Takagi-Sugeno (NEGNEVITSKY, 2011). A diferença entre os dois está no tipo de regra consequente, em que no tipo Mandani a regra consequente é um conjunto *fuzzy* e no tipo Takagi-Sugeno, a regra consequente é uma função matemática. Neste trabalho, o SIF aplicado possui variáveis *fuzzy* como saída, portanto um SIF do tipo Mandani.

O funcionamento dos SIF do tipo Mandani é dividido em três etapas (NEGNEVITSKY, 2011). A primeira etapa é a fuzificação, em que serão avaliadas as pertinências das variáveis de entrada *crisp* para cada função de pertinência do SIF. Na segunda etapa, as entradas fuzificadas são agregadas utilizando as operações lógicas das regras antecedentes e produzindo as saídas para os conjuntos *fuzzy* das regras consequentes. A terceira etapa consiste no processo de defuzificação, em que as saídas das regras consequentes fuzificadas serão combinadas e será aplicado um método de defuzificação que retornará uma variável *crisp* como saída. Neste trabalho foi aplicado o método do centróide para defuzificação (NEGNEVITSKY, 2011). Na Figura 13 são ilustrados os processos do sistema de inferência *fuzzy* do tipo Mandani.

Figura 13 – Modelo de sistema de inferência *fuzzy*.



Fonte: O próprio autor.

3.4 Algoritmo do Morcego Direcional *Fuzzy* Adaptativo

Em 2010, o Algoritmo do Morcego (AM) foi proposto como uma nova metaheurística (YANG, 2010) promissora na resolução de problemas de otimização não-convexos, em que há

a existência de uma grande quantidade de máximos e mínimos locais. No entanto, em muitos problemas, o AM apresentou problemas de convergência prematura e baixa capacidade de diversificação (CHAKRI *et al.*, 2017).

Desde então, várias propostas foram feitas para corrigir os problemas de diversificação do AM. Dentre estas, está o Algoritmo do Morcego Direcional (AMD), proposto por Chakri *et al.* (2017). O AMD propõe quatro modificações no algoritmo original, que melhoram o desempenho do AM, sendo sua principal característica a consideração de novas direções no processo de busca no espaço de solução.

Em geral, algoritmos metaheurísticos são implementados com parâmetros fixos, ou seja, com as mesmas características de diversificação e intensificação durante toda a execução. Nesse caso, espera-se que naturalmente o algoritmo transite do estágio de diversificação para o estágio de intensificação. No entanto, como os parâmetros são fixos, não há nenhuma ação para que isto ocorra. Pela natureza do algoritmo, espera-se que ele apresente este comportamento de alguma forma.

Ambos AM e AMD não operam com parâmetros fixos. Seus parâmetros variam de acordo com funções matemáticas de atualização. Neste caso, cabe ao projetista escolher as características destas funções para tornar a variação dos parâmetros mais rápida ou mais lenta.

Apesar de ser melhor que a abordagem de parâmetros fixos, a utilização de funções de atualização também não é a estratégia mais adequada, uma vez que os parâmetros da função são alterados de maneira fixa, sem nenhum *feedback* do comportamento do algoritmo. Em geral, a natureza desses algoritmos é estocástica, ou seja, a cada execução o algoritmo adota comportamentos distintos e pode fornecer resultados distintos. Para garantir a robustez do algoritmo, o ideal seria adotar uma função de alteração dos parâmetros distinta para cada execução. Como isso é inviável, o projetista deve fazer muitos testes até encontrar uma função que, na média, adapte os parâmetros efetivamente em todas as execuções.

A aplicação do SPA proposto neste trabalho possui ainda um fator de complexidade. Sempre que há modificações operativas na rede elétrica, as características do problema de otimização são alteradas e, conseqüentemente, o espaço de soluções também. Portanto, seria necessário fazer inúmeros testes para encontrar funções de adaptação que funcionem bem em todos os modos de operação. Além disso, o SPA é um sistema autônomo, que só exige a participação do operador no estágio inicial, antes de pô-lo em execução.

Para resolver este problema, o presente trabalho propõe como algoritmo de cálculo

de ajustes uma versão autoadaptável do AMD, o Algoritmo do Morcego Direcional *Fuzzy* Adaptativo. O AMDFA altera seus próprios parâmetros de operação ao longo da execução do algoritmo a partir de um SIF. Dessa forma, o algoritmo dispensa a definição de parâmetros de execução e, conseqüentemente, o papel do operador. Além disso, existe uma melhora significativa no desempenho do algoritmo, uma vez que o sistema de inferência adapta os parâmetros de operação em tempo de execução com base nas suas características de diversificação e intensificação.

3.4.1 Algoritmo do morcego e algoritmo do morcego direcional

O Algoritmo do Morcego é uma metaheurística bioinspirada no processo de ecolocalização dos morcegos. Esta técnica foi primeiramente proposta em (YANG, 2010) e considera três premissas básicas para o processo de ecolocalização:

- (i) Todos os morcegos utilizam ecolocalização para medir distâncias e a posição de todos os morcegos é codificada como um vetor solução (x_i) de um problema de otimização considerado.
- (ii) Morcegos voam aleatoriamente com velocidade v_i na posição x_i e emitem pulsos com frequência f_i (variando entre um valor mínimo f_{min} e valor máximo f_{max}) e amplitude A_i para procurar pela presa. Os morcegos podem ajustar automaticamente a frequência dos pulsos e a frequência de emissão de pulsos (r_i), dependendo da proximidade do alvo.
- (iii) A amplitude do sinal emitido varia de um valor inicial alto para um valor final mínimo e constante, enquanto a taxa de emissão de pulsos varia de um valor inicial baixo para um valor final elevado.

Como já mencionado, as variáveis do problema de coordenação ótima de relés de sobrecorrente direcionais são as correntes de *pickup* (PS) e os *Dials* (TDS) de cada relé. O conjunto de todas as variáveis forma um vetor solução, que no AM representa a posição de um morcego no espaço de soluções. Portanto, a população de morcegos considerada para o problema é apresentada na matriz da Equação (3.8).

$$P = \begin{bmatrix} PS_{1,1} & TDS_{1,1} & \cdots & PS_{1,n} & TDS_{1,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ PS_{m,1} & TDS_{m,1} & \cdots & PS_{m,n} & TDS_{m,n} \end{bmatrix} \quad (3.8)$$

em que m é o número de morcegos (ou número de soluções) e n é o número de relés considerados no problema. Logo, a matriz que representa a população de morcegos terá m linhas e $2n$ colunas.

Cada linha i da matriz P representa um vetor que define a posição x_i de um morcego no espaço de soluções. Esta representação matemática adotada é válida não somente para o AM, mas também para o AMD e o AMDFA.

Durante o processo de busca, a posição e a velocidade do morcego são modelados pelas Equações 3.9, 3.10 e 3.11.

$$f_i = f_{min} + (f_{max} - f_{min})rand \quad (3.9)$$

$$v_i^{t+1} = v_i^t + (x^* - x_i^t)f_i \quad (3.10)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3.11)$$

em que f_{min} e f_{max} são, respectivamente, a máxima e a mínima frequência, $rand \in [0, 1]$ é um vetor aleatório obtido a partir de uma distribuição de probabilidade uniforme e x^* é a posição do melhor morcego, ou seja, com maior valor de função objetivo.

A taxa de emissão de sinais r_i é utilizado para que o algoritmo alterne entre busca local e busca global (vide Algoritmo 1). Ao longo das iterações, r_i tende a aumentar e, conseqüentemente, os morcegos tendem a passar de uma característica de busca global para uma característica de busca local. Se o algoritmo passar para o movimento de busca local, a nova posição calculada substituirá a posição determinada pela equação de busca global. A equação do movimento de busca local é representado pela seguinte equação de passeio aleatório.

$$x_i^{t+1} = x_i^t + \varepsilon \langle A^t \rangle \quad (3.12)$$

em que $\varepsilon \in [-1, 1]$ é um número aleatório obtido a partir de um distribuição de probabilidade uniforme e $\langle A^t \rangle$ é a média das amplitudes dos sinais emitidos por cada morcego na iteração atual. Ao longo das iterações, a média das amplitudes tende a diminuir, conseqüentemente as buscas locais tendem a ocorrer mais frequentemente nas iterações iniciais (diversificação) e menos frequentemente nas iterações finais (intensificação).

A taxa de emissão de pulsos tem um importante papel no equilíbrio entre diversificação e intensificação do algoritmo. Iniciando a partir de um valor baixo, ele permite que os morcegos utilizem mais a busca local (Equação (3.18)) e explorem melhor o espaço de soluções, evitando a convergência prematura. Contudo, esse valor não pode se manter baixo, uma vez que é necessário que os morcegos explorem as melhores soluções encontradas. Como a taxa de emissão de sinais aumenta ao longo das iterações, os morcegos passam a se mover mais frequentemente pela busca global (Equação (3.11)), que dá maior capacidade de intensificação ao algoritmo.

O AM tem um mecanismo de rejeição de novas soluções para prevenir a convergência prematura. Com a redução da amplitude, também reduz-se a probabilidade de aceite de novas soluções. Então, nas iterações iniciais (amplitudes altas) mais soluções serão aceitas e nas iterações finais (amplitudes baixas) menos soluções serão aceitas.

A amplitude e a frequência de emissão de pulsos são atualizadas segundo as Equações (3.13) e (3.14).

$$A_i^{t+1} = \alpha A_i^t \quad (3.13)$$

$$r_i^{t+1} = r_i^\infty [1 - \exp(-\gamma t)] \quad (3.14)$$

em que $0 < \alpha < 1$ e $\gamma < 1$ são constantes que definirão quão rápido a amplitude reduzirá e a frequência de emissão de pulsos aumentará, respectivamente. A amplitude inicial A_i^0 e a frequência de emissão de pulsos final r_i^∞ são tipicamente adotados como valores entre 0 e 1. As etapas do AM são apresentadas no Algoritmo 1.

Algoritmo 1: Algoritmo do Morcego

```

Define a função objetivo
Inicializa a população inicial  $x_i$ 
Avalia as soluções iniciais  $F_i(x_i)$ 
Define as frequências iniciais  $f_i$ 
Inicializa as taxa de emissão de pulso  $r_i^0$  e as amplitudes iniciais  $A_i^0$ 
while  $t \leq t_{max}$  do
  Calcula a frequência Equação (3.9)
  Atualiza velocidades Equação (3.10)
  Atualiza as posições Equação (3.11)
  if  $rand > r_i$  then
    Gera uma solução em torno da melhor solução Equação (3.12)
  end if
  if  $rand < A_i$  &  $F(x_i) < F(x^*)$  then
    Aceita as novas soluções
    Aumenta  $r_i$  Equação (3.14)
    Reduz  $A_i$  Equação (3.13)
  end if
  Atualiza a melhor posição  $x^*$ 
end while

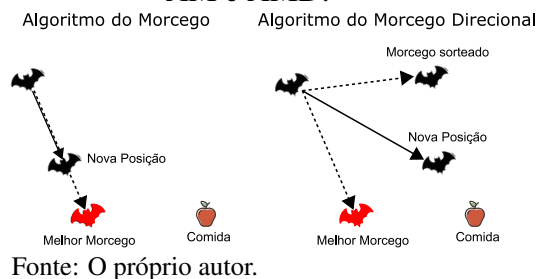
```

Como mencionado anteriormente, um dos principais problemas no AM é a convergência prematura, devido à baixa capacidade de diversificação do algoritmo em algumas condições. Para superar essa deficiência, o AMD foi proposto em Chakri *et al.* (2017).

O AMD segue as mesmas premissas de ecolocalização e possui um fluxograma similar ao do AM, no entanto este apresenta quatro modificações que melhoram as características de diversificação e intensificação do AM.

A primeira modificação é relacionada à diversificação do espaço de soluções. No AMD, o movimento do morcego é uma composição aleatória do movimento na direção do melhor morcego e do movimento na direção de um outro morcego selecionado aleatoriamente. Na Figura 14 é ilustrada a diferença no movimento entre dos morcegos entre os dois tipos de algoritmos.

Figura 14 – Comparação do movimento do morcego nos algoritmo AM e AMD.



O movimento do morcego no AMD dá maior capacidade de diversificação ao algoritmo, uma vez que o morcego movimenta-se também em direções diferentes as da posição do melhor morcego. Esta característica é importante para evitar que o algoritmo fique preso em mínimos locais. Na Figura 14, a nova posição do morcego no AMD é mais próxima da comida (alusão ao ótimo global) do que no AM, devido à composição do movimento também na direção do morcego sorteado.

Para cada morcego, um morcego diferente é aleatoriamente selecionado. Se a posição do morcego selecionado tiver melhor aptidão que o morcego atual, sua nova posição será dada pela Equação (3.15)

$$x_i^{t+1} = x_i^t + (x^* - x_i^t)f_1 + (x_k^t - x_i^t)f_2 \quad (3.15)$$

em que x_i^t é a posição do morcego i na iteração t , x^* é a posição do melhor morcego e x_k^t é a posição do morcego aleatoriamente selecionado; f_1 e f_2 são números aleatoriamente calculados pelas Equações (3.16) e (3.17). Estas constantes representam a frequência dos sinais emitidos

pelo morcego na direção do melhor morcego e do morcego aleatoriamente selecionado.

$$f_1 = f_{\min} + (f_{\max} - f_{\min})rand_1 \quad (3.16)$$

$$f_2 = f_{\min} + (f_{\max} - f_{\min})rand_2 \quad (3.17)$$

em que f_{\min} e f_{\max} são, respectivamente, a frequência máxima e mínima que o sinal pode assumir. Se o morcego aleatoriamente selecionado tiver menor valor de função objetivo que o morcego atual, o morcego atual se moverá apenas na direção do melhor morcego (Equação 3.11), assim como no AM.

A segunda mudança é relacionada à busca local do AM. No AMD, a Equação do passeio aleatório é a seguinte:

$$x_i^{t+1} = x_i^t + \varepsilon \langle A^t \rangle w_i^t \quad (3.18)$$

em que $\langle A^t \rangle$ é a média das amplitudes dos sinais emitidos por cada morcego na iteração t , $\varepsilon \in [-1, 1]$ é um vetor aleatório e w_i^t é um vetor que regula a escala da busca local ao longo das iterações. A evolução de w_i^t é dada pela Equação (3.19).

$$w_i^t = \frac{w_i^0 - w_i^\infty}{1 - t_{max}}(t - t_{max}) + w_i^\infty \quad (3.19)$$

em que w_i^0 e w_i^∞ são os vetores inicial e final, respectivamente. Neste trabalho, w_i^0 e w_i^∞ foram adotados como:

$$w_i^0 = (UB - LB)/12 \quad (3.20)$$

$$w_i^\infty = w_i^0/100 \quad (3.21)$$

em que UB e LB são os vetores que contêm os limites superiores e inferiores das variáveis PS e TDS , respectivamente. Logo, w_i decresce linearmente de w_i^0 a w_i^∞ ao longo das iterações. O objetivo é expandir o espaço de busca local dos morcegos nas iterações iniciais e retrair nas iterações finais.

A terceira modificação está relacionada com a atualização da amplitude A e da taxa de emissão r dos sinais. No AM, A e r têm uma rápida evolução ao longo das iterações, passando rapidamente do processo de aceitação para rejeição de novas soluções. Desta forma, o algoritmo acaba tendo problemas de convergência prematura. No AMD, A e r são calculados pelas Equações (3.22) e (3.23), que desaceleram a evolução das constantes e as tornam mais

fáceis de parametrizar.

$$A_i^t = \frac{A_0 - A_\infty}{1 - t_{max}}(t - t_{max}) + A_\infty \quad (3.22)$$

$$r_i^t = \frac{r_0 - r_\infty}{1 - t_{max}}(t - t_{max}) + r_\infty \quad (3.23)$$

em que 0 e ∞ representam os valores inicial e final de cada variável, respectivamente.

A última modificação está relacionada ao momento em que os parâmetros são atualizados e ao momento em que há aceitação de novas soluções. No AM, novas soluções são aceitas e A e r são atualizados somente se a nova solução gerada (x_i) for melhor que a melhor solução global (x^*) e um número gerado aleatoriamente for menor que A .

No AMD, novas soluções são aceitas e os parâmetros são atualizados se a nova solução gerada for melhor que a melhor posição histórica do morcego atual (x_i^*) e for gerado um número aleatório menor que A .

Além disso, é feita uma alteração na etapa de elitismo. No AM, é possível que a nova solução gerada tenha uma função objetivo melhor que a da melhor solução e ainda assim seja rejeitada, porque o número aleatório gerado é maior que A (vide Algoritmo 1). Para corrigir esse problema, no AMD, mesmo se a nova solução for rejeitada, se esta for melhor que a melhor solução já encontrada, a melhor solução (x^*) será atualizada.

As etapas do AMD são apresentadas no Algoritmo 2.

3.4.2 *Metaheurística proposta*

O AMDFA possui fluxograma semelhante ao do AMD. A principal diferença na atualização dos parâmetros A , r , w e f , que não é mais realizada a partir das respectivas Equações (3.22), (3.23), (3.19), (3.16) e (3.17). Agora, a atualização é feita por um SIF que tem como entradas as variáveis iteração e diversidade dos morcegos e retorna como saída os parâmetros do AMDFA.

A variável iteração mede o progresso do algoritmo e é definida pela Equação (3.24).

$$I = i/N_{max} \quad (3.24)$$

em que i é a iteração atual do algoritmo e N_{max} é o número máximo de iterações do algoritmo. Portanto, quando o algoritmo está nas iterações iniciais, I assume valores próximos de zero, e quando o algoritmo está nas iterações finais, I assume valores próximos a um.

Algoritmo 2: Algoritmo do Morcego Direcional

```

Define a função objetivo
Inicializa a população inicial  $x_i$ 
Avalia as soluções iniciais  $F_i(x_i)$ 
Inicializa as taxa de emissão de pulso  $r^0$ , amplitudes iniciais  $A^0$ ,  $w^0$  e  $w^\infty$ 
while  $t \leq t_{max}$  do
  Seleciona um morcego aleatório
  Calcula as frequências Equação (3.16) e (3.17)
  Atualiza as posições Equação (3.11) e (3.15)
  if  $rand > r_i$  then
    Atualiza  $w_i$ 
    Gera uma solução em torno da melhor solução Equação (3.18)
  end if
  if  $rand < A_i$  &  $F(x_i) < F(x_i^*)$  then
    Aceita as novas soluções
    Aumenta  $r_i$  Equação (3.23)
    Reduz  $A_i$  Equação (3.22)
  end if
  if  $F(x_i^{t+1}) < F(x^*)$  then
    Atualiza a melhor posição  $x^*$ 
  end if
end while

```

A variável diversidade é uma métrica de quão distante determinado morcego está do melhor morcego em relação aos demais morcegos. A distância entre os morcegos é calculada pela Equação (3.25) e a diversidade é dada pela Equação (3.26).

$$d_i = \|x_i - x^*\| \quad (3.25)$$

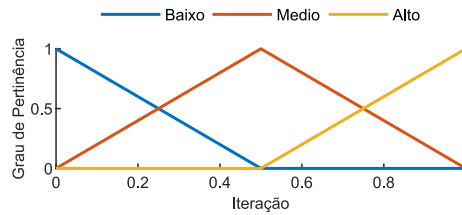
em que d_i é a distância do morcego i ao melhor morcego, x_i é o vetor posição do morcego i e x^* é vetor posição do melhor morcego.

$$D_i = \frac{d_i - d_{min}}{d_{max} - d_{min}} \quad (3.26)$$

em que D_i é a diversidade do morcego i , d_{min} é menor distância dentre os morcegos e d_{max} é a maior distância dentre os morcegos. Portanto, D_i é um valor normalizado e relativo que assume valores entre 0 e 1. Se D_i for próximo de 0, significa que o morcego i está no grupo dos morcegos mais próximos do melhor morcego, por outro lado, se D_i for próximo de 1 significa que o morcego i está no grupo dos morcegos mais distantes do melhor morcego.

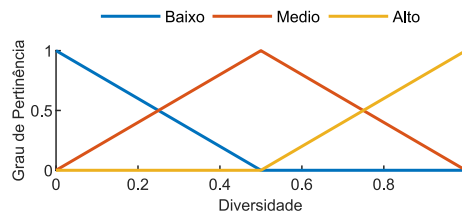
Nas Figuras 15 e 16 são apresentadas as funções de pertinência associadas a cada variável de entrada.

Figura 15 – Variáveis *fuzzy* associadas à iteração.



Fonte: O próprio autor.

Figura 16 – Variáveis *fuzzy* associadas à diversidade.



Fonte: O próprio autor.

As variáveis de saída do SIF são os parâmetros do AMDFA. O SIF retorna os próprios valores das variáveis r e A entre 0 e 1. O fator w_f será o fator que regula o tamanho do vetor w_i , segundo a Equação (3.27). Neste trabalho, w_f varia entre 0 e 500.

$$w_i^t = \frac{UB - LB}{w_f} \quad (3.27)$$

No AMDFA, a frequência do sinal emitido é atualizada a partir da Equação (3.28).

$$f_i = \Delta f \times rand \quad (3.28)$$

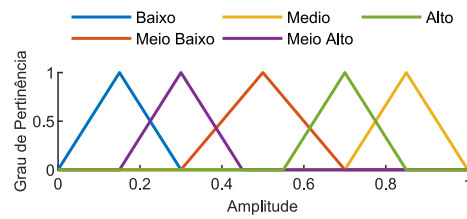
em que f_i é a frequência do sinal emitido pelo morcego i , Δf é o intervalo de variação da frequência do sinal e $rand$ é um número aleatório entre 0 e 1. É importante notar que f_i estará sempre dentro do intervalo entre 0 e Δf . Portanto, a capacidade de diversificação do algoritmo está diretamente relacionado à variável Δf , obtida a partir do SIF e que varia entre 1 e 2.

As funções de pertinência associadas às variáveis de saída são apresentadas nas Figuras 17, 18, 19 e 20.

No sistema de regras aplicado no AMDFA, todas as regras são descritas na forma: "Se I é $X1$ e D é $X2$, então r é $Y1$, A é $Y2$, w é $Y3$ e f é $Y4$ ". Foram aplicadas nove regras para o SIF conforme apresentado na Tabela 4.

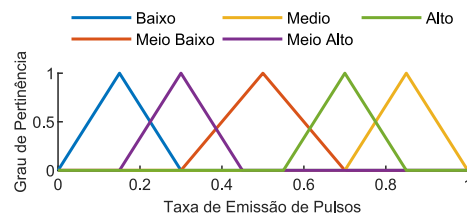
As regras foram adotadas seguindo a estratégia de fazer com que as partículas passem gradativamente de um estágio de diversificação nas iterações iniciais para um estágio de intensificação nas iterações finais. Por exemplo, na regra 1: "Se Iteração está no início e

Figura 17 – Variáveis *fuzzy* associadas à amplitude.



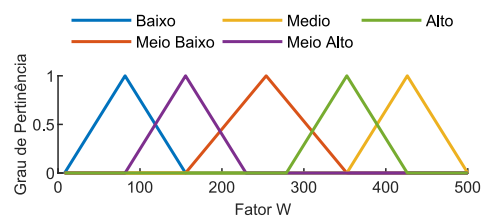
Fonte: O próprio autor.

Figura 18 – Variáveis *fuzzy* associadas à taxa de emissão de pulsos.



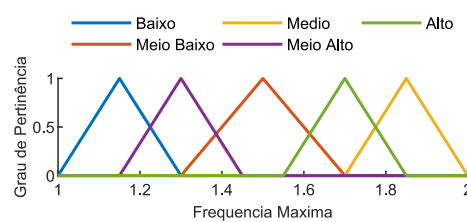
Fonte: O próprio autor.

Figura 19 – Variáveis *fuzzy* associadas ao fator w .



Fonte: O próprio autor.

Figura 20 – Variáveis *fuzzy* associadas ao fator a frequência



Fonte: O próprio autor.

Diversidade está Baixa, então r deve ser Baixo, A deve ser Alto, w deve ser Baixo e f deve ser Alto". Nas iterações iniciais, a diversidade deveria ser alta, então o SIF atuará no sentido de aumentar a diversidade do algoritmo.

Tabela 4 – Sistema de regras utilizado no SIF.

I	D	r	A	w_f	f
Início	Baixa	Baixo	Alto	Baixo	Alto
Início	Média	Baixo	Alto	Baixo	Alto
Início	Alta	Meio baixo	Meio alto	Meio baixo	Meio alto
Meio	Baixa	Meio baixo	Meio alto	Meio baixo	Meio alto
Meio	Média	Média	Média	Média	Média
Meio	Alta	Meio alto	Meio baixo	Médio alto	Meio baixo
Fim	Baixa	Meio alto	Meio baixo	Meio alto	Meio baixo
Fim	Média	Meio alto	Meio baixo	Meio alto	Meio baixo
Fim	Alta	Alto	Baixo	Alto	Baixo

Fonte: O próprio autor.

As etapas do AMDFA estão apresentadas no Algoritmo 3.

Algoritmo 3: Algoritmo do morcego direcional fuzzy adaptativo

```

Define a função objetivo
Inicializa a população inicial  $x_i$ 
Avalia as soluções iniciais  $F_i(x_i)$ 
while  $t \leq t_{max}$  do
  Calcula a iteração Equação (3.24)
  Calcula a diversidade de cada morcego Equação (3.26)
  Seleciona um morcego aleatório
  Calcula os limites das frequências utilizando o SIF
  Atualiza as posições Equações (3.11) e (3.15)
  Calcula  $r_i$  a partir do SIF
  if  $rand > r_i$  then
    Calcula o novo  $w_i$  a partir do SIF
    Gera uma solução em torno da solução atual Equação (3.18)
  end if
  Calcula  $A_i$  a partir do SIF
  if  $rand < A_i$  &  $F(x_i) < F(x_i^*)$  then
    Aceita as novas soluções
  end if
  if  $F(x_i^{t+1}) < F(x^*)$  then
    Atualiza a melhor posição  $x^*$ 
  end if
end while

```

3.5 Validação do algoritmo do morcego direcional fuzzy adaptativo

Para validar o desempenho do AMDFA na coordenação ótima de relés direcionais de sobrecorrente, o AMDFA foi aplicado na obtenção dos ajustes dos relés de três redes de referência comumente utilizadas neste problema: rede 3 barras, rede 9 barras e rede 30 barras. As redes de referência são todas de topologia anel, devido a esse problema de otimização ser tipicamente aplicado à proteção de sistemas de transmissão e sub-transmissão. No entanto, a modelagem matemática do problema de otimização é a mesma para sistemas de topologia radial e anel. No próximo capítulo, o SPA é validado em uma rede de distribuição radial com geração distribuída.

Foi feita uma análise comparativa da convergência, de aderência às restrições de coordenação (descoordenação), da robustez e do tempo computacional do AMDFA em relação às metaheurísticas: Algoritmo Genético (AG), Otimização por Enxame de Partículas (OEP), Evolução Diferencial (ED), Algoritmo do Morcego (AM) e Algoritmo do Morcego Direcional (AMD). O *script* em MATLAB de cada metaheurística implementada é apresentada no Anexo B.

Na comparação de desempenho, os algoritmos foram executados com os mesmos parâmetros em todos os casos, para tornar a comparação mais próxima de uma aplicação de proteção adaptativa. Para determinar os parâmetros de cada algoritmo, foi feito um delineamento fatorial completo entre os parâmetros dos algoritmos e aplicado um teste ANOVA para identificar o nível de significância da aplicação de cada parâmetro.

Os parâmetros considerados e os resultados de significância do ANOVA são apresentados no Anexo C. A partir dos resultados obtidos, os seguintes parâmetros foram adotados para cada algoritmo:

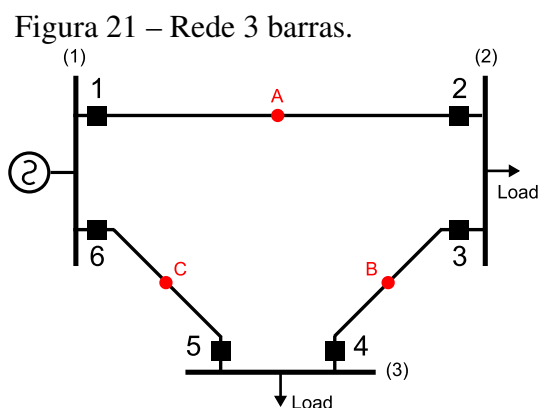
- AG: Tamanho da população = 300; Taxa de mutação = 0,05, Taxa de cruzamento = 1, Critério de seleção = 0,5;
- OEP: Tamanho da população = 300; Fator de inercia inicial = 0,6; Fator de inercia final = 0,1; $c_1 = 1$; $c_2 = 3$
- ED: Tamanho da população = 300; $F = 0,9$, Taxa de cruzamento = 0,9;
- AM: Tamanho da população = 300; $A_0 = 1,0$, $\alpha = 0,992$, $r_0 = 1,0$, $\gamma = 0,003$, $f_{min} = 0,5$ e $f_{max} = 1,0$
- AMD: Tamanho da população = 300; $A_0 = 0,9$, $A_\infty = 0,3$, $r_0 = 0,3$, $r_\infty = 0,6$, $f_{min} = 0,5$ e $f_{max} = 2,0$
- AMDFA: Tamanho da população = 300.

A título de comparação, todos os algoritmos foram executados durante 300 iterações. Os testes foram feitos no MATLAB utilizando um computador pessoal com processador Intel[®] Core[™] i5-3210M CPU @2.5 GHz x64 com 6 GB RAM.

Nos subtópicos a seguir, serão apresentados os resultados obtidos para cada rede de referência.

3.5.1 Sistema 3 barras

A rede 3 barras, apresentada na Figura 21, contém 1 gerador síncrono, 3 linhas de transmissão e 6 relés de sobrecorrente direcionais com curva normalmente inversa. Portanto, o problema de otimização envolve 12 variáveis (PS_1 a PS_6 e TDS_1 a TDS_6). Foram adotados os seguintes ajustes limites para os relés: $PS_{min} = 1,5$, $PS_{max} = 1,5$, $TDS_{min} = 0,1$ e $TDS_{max} = 1,1$. As correntes de curto-circuito foram obtidas para faltas nos pontos A, B e C. Na Tabela 5 são apresentados os valores de Relação de Transformação de Corrente (RTC) dos TC de cada relé. As correntes de curto-circuito que passam por cada relé são apresentadas no Anexo A.



Fonte: O próprio autor.

Tabela 5 – RTC dos relés da rede 3 barras.

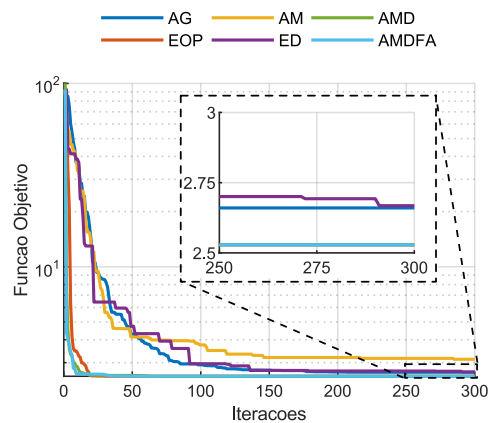
Relé	RTC	Relé	RTC
1	300-5	4	300-5
2	200-5	5	200-5
3	200-5	6	400-5

Fonte: O próprio autor.

Na Figura 22 é apresentada a evolução da convergência dos algoritmos testados. Para tornar a análise menos sujeita a estocasticidade, foi aplicada a mesma sequência de números

aleatórios para todos os algoritmos. O valor de função objetivo das últimas 100 iterações são destacadas para tornar mais clara a diferença entre os resultados finais.

Figura 22 – Evolução da convergência dos algoritmos para a rede 3 barras.

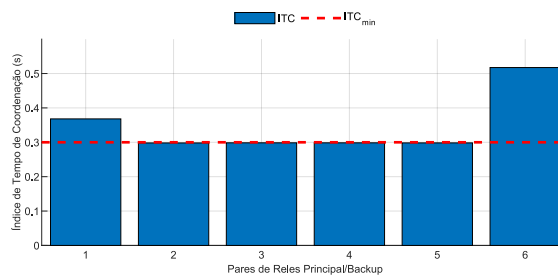


Fonte: O próprio autor.

O AMDFA obteve o melhor desempenho dentre os algoritmos, provendo menor tempo de operação ao sistema de proteção e apresentando uma das maiores velocidades de convergência.

No gráfico da Figura 23 é apresentado o Índice de Tempo de Coordenação (ITC) de cada par de relés principal/*backup* e o *ITC* mínimo adotado para esta rede (0,3 s). O menor *ITC* obtido ficou apenas 0,0019 segundos (0,63%) abaixo do *ITC* mínimo estabelecido. Portanto, o AMDFA proporcionou boa adesão às restrições de coordenação.

Figura 23 – Índice de tempo de coordenação dos pares de relés principal e *backup* para a rede 3 barras.



Fonte: O próprio autor.

Para avaliar a robustez do algoritmo, todos os algoritmos foram executados 300 vezes e foram coletados os seus valores de função objetivo. Na Tabela 6 são apresentados a

média e o desvio padrão obtidos.

Tabela 6 – Análise de robustez - rede 3 barras.

	GA	OEP	ED	AM	AMD	AMDFA
Média	2,6851	2,5329	2,6519	2,9984	2,5347	2,5287
Desvio Padrão	0,0346	0,0136	0,0313	0,4383	0,0168	0,0000
Tempo de execução médio (s)	2,0184	2,009	1,8714	4,1665	4,5545	3,7137

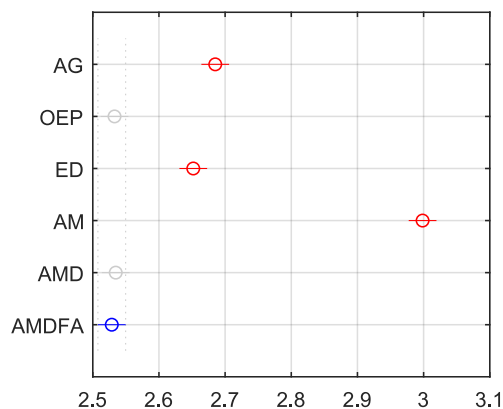
Fonte: O próprio autor.

O AMDFA demonstrou a melhor robustez dentre os algoritmos testados com a menor média e o menor desvio padrão dentre todos os algoritmos. O desvio padrão muito abaixo dos demais demonstra que o algoritmo convergiu com grande frequência para a média das soluções obtidas.

Quanto ao tempo computacional, era esperado que o AMDFA apresentasse tempos mais elevados, devido à maior complexidade do algoritmo e a inclusão da etapa de inferência *fuzzy* dos parâmetros. No entanto, devido a menor complexidade da coordenação de relés direcionais na rede 3 barras, os tempos médios de execução ficaram muito próximos.

Para avaliar a significância da diferença entre as médias obtidas para cada algoritmo, foi realizado um teste ANOVA entre as amostras de cada algoritmo. O diagrama da Figura 24 apresenta a comparação dos intervalos de confiança de cada amostra para um p-valor de 5 %. Nota-se que o OEP, o AMD e o AMDFA obtiveram as menores médias, no entanto as suas médias não podem ser consideradas estatisticamente distintas.

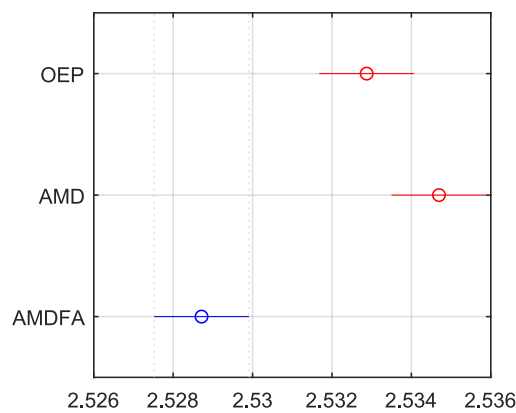
Figura 24 – Intervalo de confiança das médias das funções objetivos de cada algoritmo.



Fonte: O próprio autor.

No teste ANOVA, quando existem médias de fatores muito distintas, a média global pode tornar-se elevada e a diferença entre as médias dos fatores torna-se menos significativa. Visando este efeito, o teste ANOVA foi aplicado entre as amostras das três menores médias. O diagrama dos intervalos de confiança para os algoritmos OEP, AMD e AMDFA é apresentado na Figura 25. Verifica-se agora que o AMDFA obteve média significativamente menor que a média do AMD e OEP.

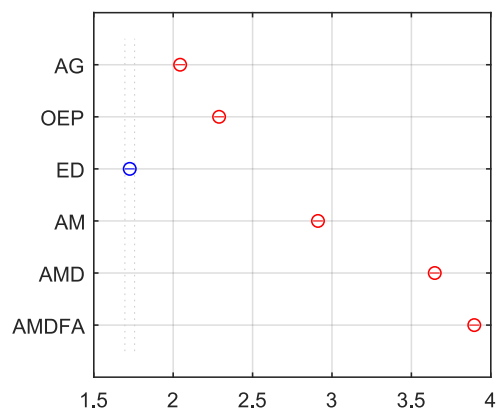
Figura 25 – Intervalo de confiança das médias das funções objetivos do OEP, AMD e AMDFA.



Fonte: O próprio autor.

O teste ANOVA também foi aplicado para verificar se a diferença entre os tempos médios de execução é estatisticamente significativa. Os intervalos de confiança das amostras de cada algoritmo são apresentados na Figura 26.

Figura 26 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo



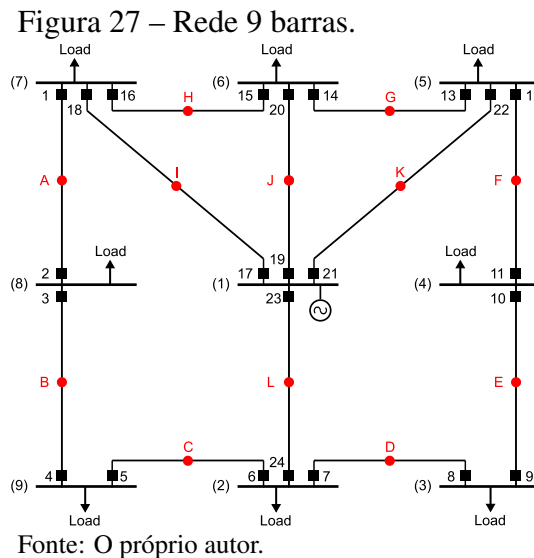
Fonte: O próprio autor.

Observa-se que o ED obteve tempo médio de execução significativamente menor em relação aos demais algoritmos, enquanto o AMDFA obteve o maior tempo médio de execução.

3.5.2 Sistema 9 barras

A rede 9 barras, apresentada na Figura 27, é alimentada por um gerador síncrono de 100 MVA, 33 kV e impedância de $j0,1$ p.u. O sistema contém também 12 linhas de transmissão com impedância $j0,1$ p.u. e 24 relés de sobrecorrente direcionais com curva normalmente inversa. As RTCs de todos os relés foram adotadas como 500-1. Mais informações sobre o sistema podem ser encontradas em (BEDEKAR; Bhide, 2011).

Foram adotadas as seguintes restrições para os ajustes dos relés: $PS_{min} = \max(1,5; 1.25I_{max}^L)$, $PS_{max} = (2,5; \frac{2}{3}I_{min}^F)$, $TDS_{min} = 0,025$ e $TDS_{max} = 1,2$, em que I_{max}^L é a corrente de carga máxima que passa pelo relé e I_{min}^F é a corrente de falta mínima que passa pelo relé. As correntes de curto-circuito foram obtidas para faltas nos pontos A a L e são apresentadas no Anexo A.

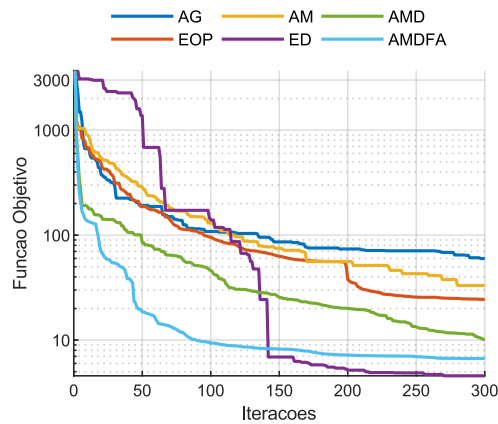


Para tornar a análise menos sujeita à estocasticidade, foi aplicada a mesma sequência de números aleatórios para todos os algoritmos. Na Figura 28 é apresentada a evolução da convergência dos algoritmos testados.

O AMDFA obteve o segundo melhor valor de função objetivo, inferior somente ao valor de função objetivo do algoritmo ED. Por outro lado, o AMDFA demonstrou melhor velocidade de convergência em relação ao ED, que só obteve resultado melhor que o AMDFA na iteração 142.

No gráfico da Figura 29 é apresentado o ITC de cada par de relés principal/*backup* e

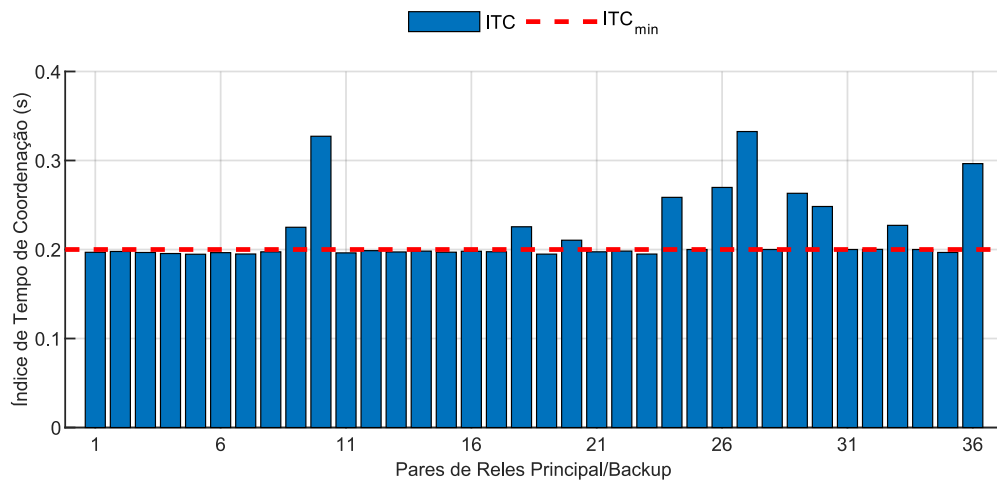
Figura 28 – Evolução da convergência dos algoritmos para a rede 9 barras.



Fonte: O próprio autor.

o ITC mínimo adotado para esta rede (0,2 s).

Figura 29 – Índice de tempo de coordenação dos pares de relés principal e *backup* para a rede 9 barras.



Fonte: O próprio autor.

A menor descoordenação obtida foi de 0,0053 segundos (2,65%) abaixo do ITC_{min} . Portanto, o AMDFA proporcionou boa adesão à restrição de coordenação.

Para avaliar a robustez do algoritmo, todos os algoritmos foram executados 300 vezes e foram coletados os seus valores de função objetivo. Na Tabela 7 são apresentados a média e o desvio padrão obtidos.

O AMDFA obteve a segunda menor média, com diferença pouco significativa em relação à menor (0,46 %), e o menor desvio padrão dentre todos os algoritmos. Com relação ao tempo de execução médio, como já era esperado, o AMDFA obteve o maior tempo dentre todos

Tabela 7 – Análise de robustez - rede 9 barras.

	GA	OEP	ED	AM	AMD	AMDFA
Média	76,49	23,95	4,765	39,21	8,527	7,094
Desvio Padrão	13,10	14,38	0,742	8,482	1,499	1,059
Tempo de execução médio (s)	10,41	9,04	5,867	15,695	13,218	8,337

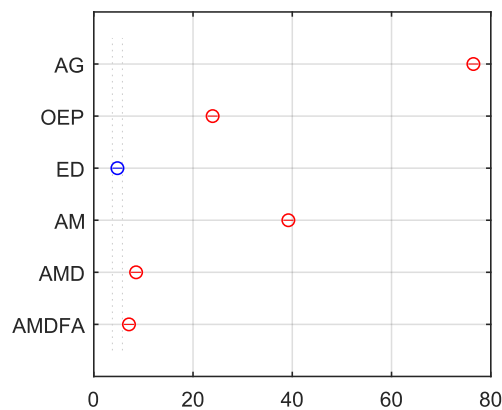
Fonte: O próprio autor.

os algoritmos (0,703 segundos acima do menor tempo de execução).

Para avaliar a significância da diferença entre as médias obtidas de função objetivo e tempo de execução de cada algoritmo, foi realizado um teste ANOVA entre as amostras de cada algoritmo.

O diagrama da Figura 30 apresenta a comparação dos intervalos de confiança de cada amostra para um p-valor de 5 %. Observa-se que o ED, o AMD e o AMDFA obtiveram as menores médias de função objetivo e segundo a análise dos intervalos de confiança, as três médias são estatisticamente distintas.

Figura 30 – Intervalo de confiança das médias das funções objetivos de cada algoritmo.

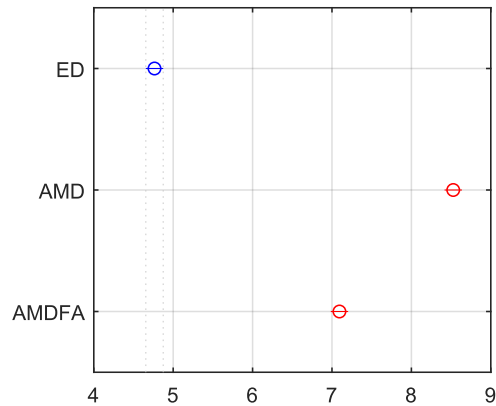


Fonte: O próprio autor.

Visando avaliar a diferença estatística entre as três menores médias isoladas, foi aplicado o teste ANOVA entre as amostras das três menores médias. O diagrama dos intervalos de confiança para os algoritmos ED, AMD e AMDFA é apresentado na Figura 25. Verifica-se que, de fato, o ED obteve a menor média e significativamente inferior as médias do AMD e AMDFA.

O teste ANOVA também foi aplicado para verificar se a diferença entre os tempos médios de execução é estatisticamente significativa. Os intervalos de confiança das amostras de

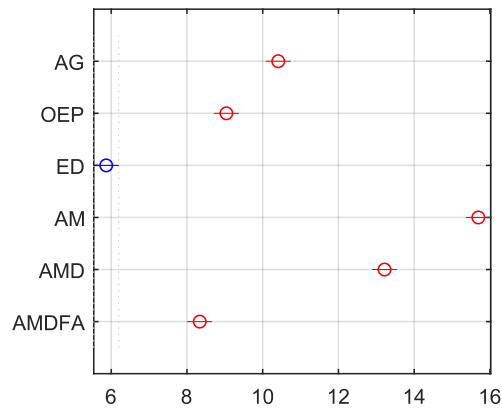
Figura 31 – Intervalo de confiança das médias das funções objetivos do OEP, AMD e AMDFA.



Fonte: O próprio autor.

cada algoritmo são apresentadas na Figura 32. Desta vez, o ED obteve o menor tempo médio de execução, seguido do AMDFA.

Figura 32 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo



Fonte: O próprio autor.

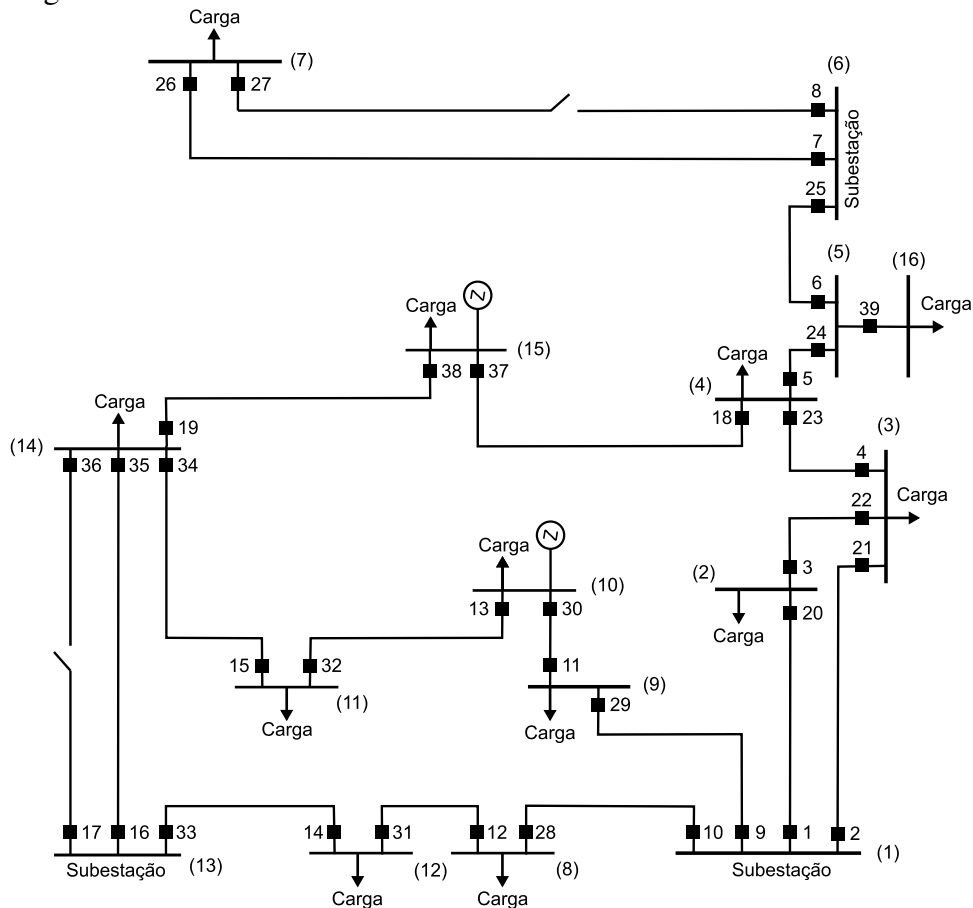
3.5.3 Sistema 30 barras

A rede 30 barras, apresentada na Figura 33, é alimentada por três subestações de 33 kV nas barras 1, 6 e 13. Além disso, o sistema também é alimentado por duas fontes de geração distribuída conectadas nas barras 10 e 15. O sistema possui 20 linhas de transmissão e é protegido por 39 relés de sobrecorrente direcional com curva normalmente inversa. As correntes de curto-circuito que passam por cada relé estão disponíveis no Anexo A. As RTC de cada relé

foram adotadas como 500-1.

Foram adotadas as seguintes restrições para os ajustes dos relés: $PS_{min} = \max(0,5; \frac{1}{3}I_{min}^F)$, $PS_{max} = (2,5; \frac{2}{3}I_{min}^F)$, $TDS_{min} = 0,1$ e $TDS_{max} = 1,1$, em que I_{min}^F é a corrente de falta mínima que passa pelo relé.

Figura 33 – Rede 30 barras.



Fonte: O próprio autor.

Na Figura 34 é apresentada a evolução da convergência dos algoritmos testados. Para tornar a análise menos sujeita a estocasticidade, foi aplicada a mesma sequência de números aleatórios para todos os algoritmos.

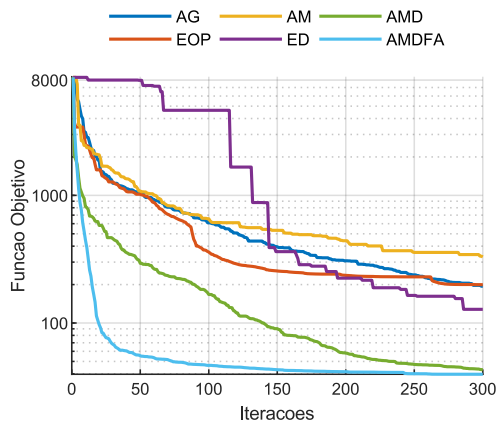
O AMDFA obteve o melhor valor de função objetivo e também a melhor velocidade de convergência dentre todos os algoritmos.

No gráfico da Figura 35 é apresentado o ITC de cada par de relés principal/*backup* e o ITC mínimo adotado para esta rede (0,2 s).

A menor descoordenação obtida foi de 0.0143 segundos (7,13%). Portanto, o AMDFA proporcionou boa adesão à restrição de coordenação.

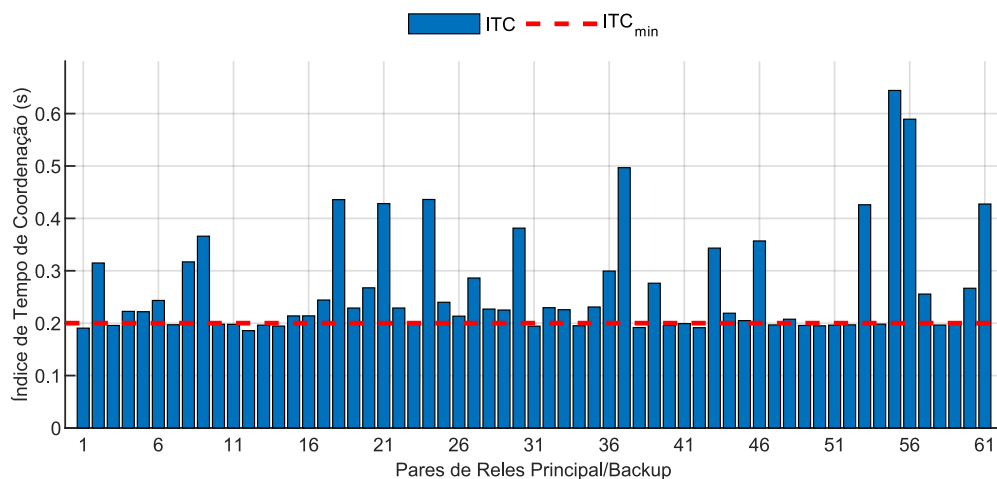
Para avaliar a robustez do algoritmo, todos os algoritmos foram executados 300

Figura 34 – Evolução da convergência dos algoritmos para a rede 30 barras.



Fonte: O próprio autor.

Figura 35 – Índice de tempo de coordenação dos pares de relés principal e *backup* para a rede 30 barras



Fonte: O próprio autor.

vezes e foram coletados os seus valores de função objetivo. Na Tabela 8 são apresentados a média e o desvio padrão obtidos.

O AMDFA apresentou a melhor robustez, obtendo a menor média e o menor desvio padrão dentre os algoritmos testados. Assim como nos testes anteriores, o AMDFA apresentou o maior tempo de execução de todos os algoritmos (24,07 segundos acima do menor tempo de execução).

Para avaliar a significância da diferença entre as médias obtidas de função objetivo e tempo de execução de cada algoritmo, foi realizado um teste ANOVA entre as amostras de cada algoritmo.

O diagrama da Figura 36 apresenta a comparação dos intervalos de confiança de

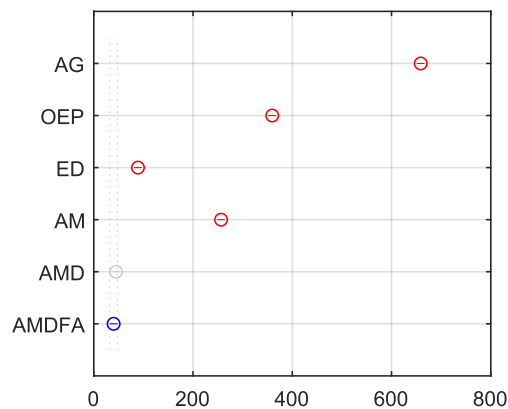
Tabela 8 – Análise de robustez - rede 30 barras.

	GA	OEP	ED	AM	AMD	AMDFA
Média	658,8	359,6	89,03	256,52	44,92	39,96
Desvio Padrão	78,11	130,5	39,87	42,08	4,560	0,399
Tempo de execução médio (s)	23,70	24,25	11,87	17,81	27,66	35,94

Fonte: O próprio autor.

cada amostra para um p-valor de 5 %. Observa-se que o ED, o AMD e o AMDFA obtiveram as menores médias de função objetivo. Analisando os intervalos de confiança, verifica-se que as médias de AMD e AMDFA são estatisticamente iguais.

Figura 36 – Intervalo de confiança das médias das funções objetivos de cada algoritmo.

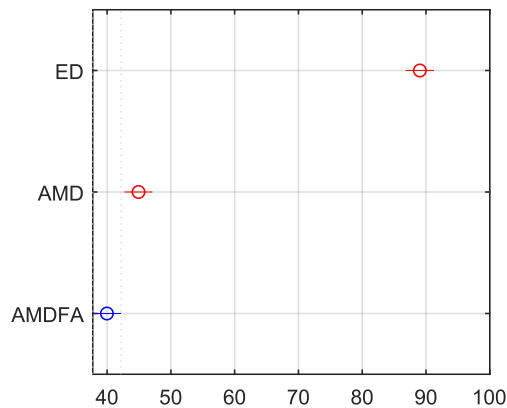


Fonte: O próprio autor.

Visando avaliar a diferença estatística entre as três menores médias isoladas, foi aplicado o teste ANOVA entre as amostras das três menores médias. O diagrama dos intervalos de confiança para os algoritmos ED, AMD e AMDFA é apresentado na Figura 31. Na nova análise, o AMDFA apresenta média estatisticamente menor que a média de ED e AMD.

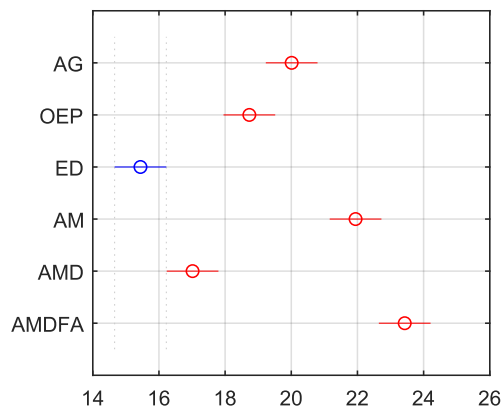
O teste ANOVA também foi aplicado para verificar se a diferença entre os tempos médios de execução é estatisticamente significativa. Os intervalos de confiança das amostras de cada algoritmo são apresentadas na Figura 38. Quanto ao tempo de execução, o ED apresentou tempo de execução médio significativamente inferior aos demais algoritmos.

Figura 37 – Intervalo de confiança das médias das funções objetivos do ED, AMD e AMDFA.



Fonte: O próprio autor.

Figura 38 – Intervalo de confiança das médias dos tempos de execução de cada algoritmo



Fonte: O próprio autor.

3.5.4 Discussão dos resultados

O AMDFA foi testado utilizando três redes de referência, avaliando convergência, descoordenação, robustez e tempo de execução. O desempenho do AMDFA foi comparado ao desempenho dos algoritmos: AG, OEP, ED, AM e AMD.

Com relação à convergência, o AMDFA obteve o melhor balanço entre velocidade de convergência e minimização da função objetivo em todos os casos. Em geral, os algoritmos AG, OEP e ED convergiram para mínimos locais. O AMD obteve desempenho muito semelhante ao AMDFA com relação à característica de convergência, no entanto, a característica de auto adaptação do AMDFA proveu melhor desempenho ao algoritmo proposto.

O ED apresentou bom desempenho em todos os testes, inclusive obtendo melhor

minimização da função objetivo no estudo de caso com a rede 9 barras. No entanto, o AMDFA apresentou melhor relação entre função objetivo e velocidade de convergência. Em geral, o ED apresentou boa característica de intensificação, porém baixa diversificação, principalmente nas iterações iniciais.

O balanço entre minimização do tempo de operação e velocidade de convergência é essencial para sistemas de proteção adaptativa, pois quanto mais rápido forem fornecidos bons ajustes, mais rápido o sistema de proteção será reajustado e melhor será a confiabilidade do sistema elétrico.

O AMDFA também apresentou boa característica de adesão às restrições de descoordenação. O menor índice de tempo de coordenação obtido foi de 1,9831 segundos, quando o limite mínimo do ITC foi de $0,2s$ (0,8845 %). Portanto, um valor muito próximo do limite e, portanto, passível de ser não ser considerada uma descoordenação na prática.

Além disso, o valor adotado de $ITC_{min} = 0,2s$ pode ser considerado um critério conservador, uma vez que o ITC pode ser escolhido entre 0,2 e 0,5 segundos (ANDERSON, 1999). No problema de otimização, diferentemente do processo de cálculo analítico de coordenação, a escolha de um ITC mais baixo torna o problema mais complexo, uma vez que força o algoritmo a encontrar menores tempos de operação.

Quanto a robustez, nos testes com as redes 3 barras e 30 barras, o AMDFA apresentou a menor média e o menor desvio padrão dentre todos os algoritmos. Já nos testes com a rede 9 barras, obteve a segunda menor média e o segundo menor desvio padrão, com diferenças muito baixas com relação aos menores.

Portanto, os resultados implicam que o algoritmo apresenta boa robustez, ou seja, a sua característica estocástica pouco afeta a qualidade dos resultados fornecidos. Esta característica é importante, especialmente, para sistemas de proteção adaptativa, uma vez que garante que o algoritmo sempre calculará bons ajustes para o sistema de proteção.

Já com relação ao tempo de execução, o AMDFA apresentou desempenho um pouco inferior com relação aos demais. Nos testes com a rede de 3 barras, o AMDFA apresentou tempo de execução semelhante aos demais algoritmos. Já nos testes com as redes 9 barras e 30 barras, o tempo de execução do AMDFA foi superior a todos os outros. Nota-se que o tempo de execução do AMDFA é mais sensível que os demais algoritmos quando o número de variáveis do problema (associado ao número de relés na rede) aumenta.

Apesar disso, o maior tempo de execução médio do AMDFA (35,94 segundos),

obtido para a rede de 30 barras com 39 relés e, portanto, 78 variáveis, é um tempo razoável para aplicações práticas, tendo em vista que uma interrupção para não ser considerada contínua deve durar até 180 segundos (ANEEL, 2015). Desta forma, em um esquema de proteção adaptativa integrado a um sistema de recomposição automática, por exemplo, em que o sistema de recomposição automático deve: isolar a falta, corrigir os ajustes do sistema de proteção para, só então, reenergizar as cargas, a execução do AMDFA representaria apenas 19,96 % do tempo total necessário.

Além disso, a característica de boa convergência do AMDFA garante que, se for necessário que o AMDFA pare a busca por novas soluções, devido a limitações de tempo, ainda assim o AMDFA proporcionará melhores soluções que os demais algoritmos. Nos testes com as redes 3 barras e 30 barras, o AMDFA obteve o menor valor de função objetivo durante todas as iterações. Nos testes com a rede 9 barras, o AMDFA obteve o menor valor de função objetivo durante 142 iterações (47.33 % do total de iterações), até ser superado pelo ED.

Além dos recursos aqui discutidos, a característica de auto adaptação do algoritmo permite que este apresente bom desempenho, sem a necessidade de escolha de muitos parâmetros de execução. O único parâmetro que precisa ser escolhido é o tamanho da população inicial, que é comum a todas as metaheurísticas testadas e que está associado diretamente a complexidade do problema, sendo um parâmetro que pode ser definido de maneira mais genérica que os demais. Um delineamento fatorial completo, como feito para o AMDFA nos testes, poderia identificar um valor ideal de tamanho de população. Desta forma, um SPA que aplica o AMDFA para o cálculo dos ajustes, teria integração mais simples com o sistema elétrico.

3.6 Considerações finais

Neste Capítulo foram apresentados os conceitos relacionados ao AMDFA, proposto neste trabalho. Ao final do capítulo, foi apresentada a validação do algoritmo metaheurístico proposto para coordenação ótima de relés de sobrecorrente direcionais em redes não radiais.

O algoritmo superou o desempenho dos algoritmos AG, OEP, ED, AM e AMD, apresentando características de boa velocidade de convergência, baixa descoordenação e alta robustez. O algoritmo proposto apresentou desempenho pior, em relação aos demais algoritmos, quanto ao esforço computacional, porém este ainda manteve-se dentro dos limites práticos da operação do sistema elétrico.

Além disso, a característica do AMDFA de auto adaptação reduz o esforço do

projetista de escolher os parâmetros que proporcionem o desempenho ideal do algoritmo. Caso contrário, a implementação deste em um SPA tornaria-se mais complexa.

4 SISTEMA DE PROTEÇÃO ADAPTATIVA APLICADO A MICRORREDES

Neste capítulo é apresentado o sistema de proteção adaptativa proposto e é analisado o desempenho do mesmo operando em uma microrrede. Para isto, foi feita uma análise da correção dos ajustes feita pelo SPA em diferentes cenários de operação da microrrede, considerando variações de conexão de GD, topologia da rede e ilhamento. Os resultados foram comparados aos ajustes obtidos por um sistema de proteção com ajustes fixos otimizados.

4.1 Sistema de Proteção Adaptativa

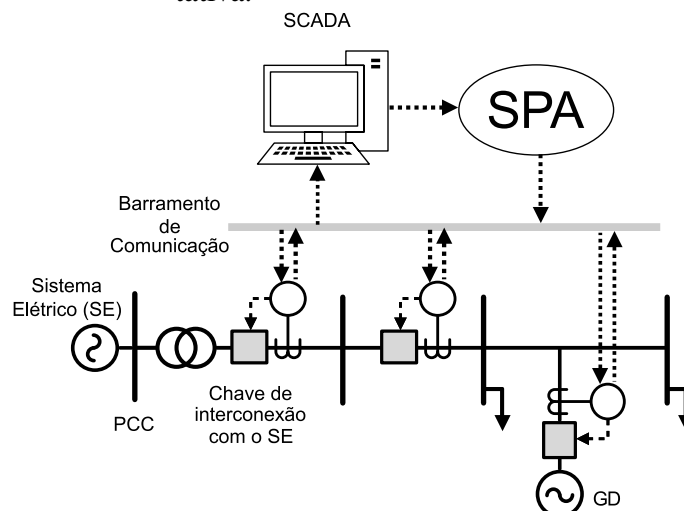
4.1.1 Arquitetura do SPA

O SPA proposto neste trabalho altera os ajustes dos relés de proteção de sobrecorrente direcional mediante as seguintes condições:

- conexão/desconexão de unidade de geração;
- mudança de topologia da rede e;
- ilhamento da microrrede

Para isto, o SPA proposto baseia-se em uma arquitetura centralizada de automação, em que todos os equipamentos monitorados e controlados comunicam-se com um único SCADA localizado na subestação (ou no centro de comando da distribuidora). Para integrar-se, basta que o SPA esteja conectado na mesma rede de comunicação que o SCADA. Na Figura 39 é apresentada a arquitetura do SPA.

Figura 39 – Arquitetura do Sistema de Proteção Adaptativa.



Fonte: O próprio autor.

Os estados das chaves são monitorados pelo Sistema Supervisório de Controle e Aquisição de Dados (em inglês *Supervisory, Control And Data Aquisition* (SCADA)). Sempre que há alteração do estado de alguma chave na rede, o SCADA informa ao SPA sobre a mudança. Com base nas novas condições da rede, o SPA calcula novos ajustes e os envia diretamente aos relés.

A arquitetura proposta tem a vantagem de aproveitar as características dos sistemas de automação das redes elétricas, que tipicamente já são monitoradas por SCADA. Portanto, a integração do SPA torna-se mais simples, uma vez que o sistema de monitoramento já está implementado.

4.1.2 Algoritmo do SPA

Antes de entrar em operação, o SPA requer que parâmetros de entrada relacionados ao sistema elétrico sejam fornecidos. Na Tabela 9 os parâmetros de entrada são apresentados e divididos em: Dados de proteção e dados elétricos.

Tabela 9 – Parâmetros de entrada do SPA

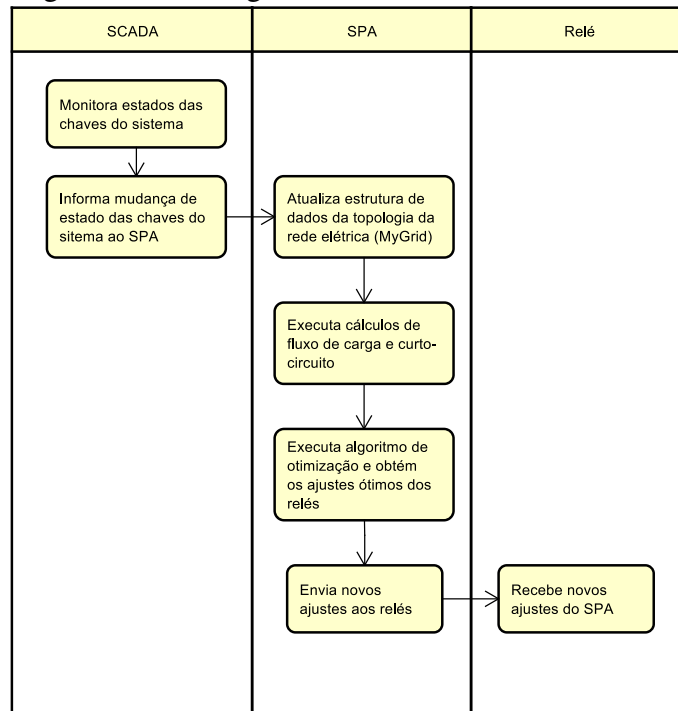
Parâmetros de Entrada	
Dados de Proteção	PS máximo de cada relé
	PS mínimo de cada relé
	TD máximo de cada relé
	TD mínimo de cada relé
	Índice de tempo de Coordenação mínimo
	Endereço IP de cada relé
	Relação de transformação dos TCs
Dados Elétricos	Impedância dos condutores
	Potência das cargas
	Impedância equivalente do sistema elétrico
	Potência das GDs
	Impedância das GDs
	Impedância dos transformadores

Fonte: O próprio autor.

Além disso, o SPA requer também a estrutura de dados que represente a topologia da rede para executar cálculos de fluxo de carga e curto-circuito. No SPA desenvolvido neste trabalho, foi utilizado o *software* MyGrid (GREI, 2019) para esta finalidade. O MyGrid permite ao SPA não só conhecer a topologia da rede elétrica como também executar cálculos de fluxo de carga e curto-circuito. O algoritmo do SPA é apresentado na fluxograma da Figura 40.

Como é apresentado na Figura 40, após o recebimento da mensagem do SCADA informando sobre a mudança topológica, o SPA atualiza a topologia da rede e executa os cálculos

Figura 40 – Fluxograma do SPA



Fonte: O próprio autor.

de fluxo de carga e curto-circuito. A partir desses cálculos o SPA determina as correntes de carga e as contribuições de curto-circuito que passam por cada relé. Essas informações são utilizadas para compor as restrições (Equações 3.4 e 3.5) do problema de otimização a ser resolvido pelo algoritmo.

De posse das correntes de carga, das contribuições de curto-circuito e dos parâmetros de entrada do sistema, o SPA calcula os novos ajustes do sistema de proteção, utilizando o algoritmo de otimização proposto. Por fim, o SPA obtém os ajustes otimizados e envia-os aos relés de proteção.

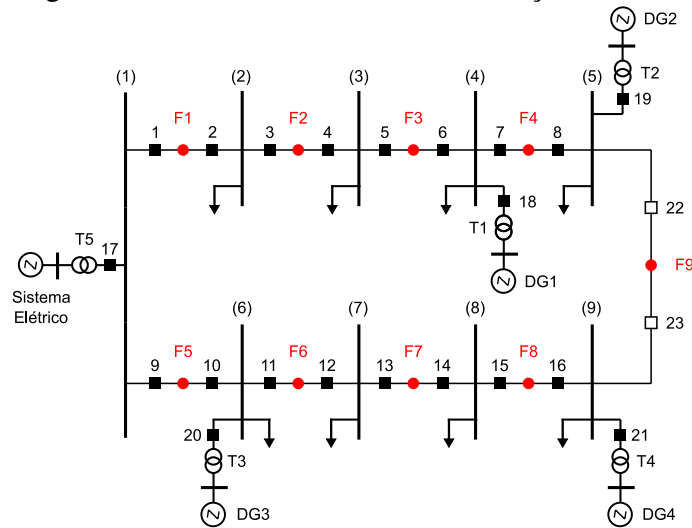
4.2 Metodologia dos testes

4.2.1 Microrrede teste

Para avaliar o desempenho do SPA em uma microrrede, foram utilizados os dados da rede de distribuição denominada: Rede canadense de distribuição em média tensão (DEHGHANPOUR *et al.*, 2018). A topologia da microrrede é apresentada na Figura 41.

O sistema elétrico alimenta a microrrede em 115 kV e apresenta potência de curto circuito de 500 MVA. A barra 1 é considerada como o ponto comum de conexão da rede de distribuição ao sistema elétrico. Para operar em modo isolado, a chave 17 é aberta, desconectando

Figura 41 – Rede canadense de distribuição.



Fonte: O próprio autor.

a microrrede do sistema elétrico. O sistema conecta-se à rede elétrica através do transformador T5 de 115-12,47 kV, 5 MVA e $X_{tr} = 0,1$ p.u. A rede é também alimentada por quatro fontes de GD (GD1 a GD4), cada uma de 3 MVA, 480 V e $X_d = 0,2$ p.u. Cada fonte conecta-se à rede através de um transformador (T1 a T4) de 0,48-12,47 kV, 5 MVA e $X_{tr} = 0,1$ p.u. As linhas possuem impedância de $0,1529 + j0,1406$ ohm/km e comprimento de 500 m, cada uma. Cada carga representada é de 2 MVA com fator de potência 0,9 indutivo. As correntes de curto-circuito foram obtidas a partir dos pontos de falta (F1 - F9) apresentados na Figura 41.

O sistema possui 23 equipamentos de disjunção associados com relés de sobrecorrente direcionais. Neste trabalho, as chaves 22 e 23 foram adicionadas para possibilitar a recomposição de trechos desenergizados pelo alimentador adjacente.

Cada relé foi considerado com curva NI e RTC de 500-1. Seguindo o modelo apresentado na Seção 3.1, os valores limites de ajustes foram adotados segundo as equações 4.1 e 4.2.

$$\max(0,1; 1,3I_{max}^L) \leq PS_i \leq \min(3,2; \frac{2}{3}I_{min}^F) \quad (4.1)$$

$$0,05 \leq TDS_i \leq 1,5 \quad (4.2)$$

em que 0,1; 3,2; 0,05 e 1,5 são, respectivamente, os valores mínimos e máximos de PS e TDS do relé comercial SEL-751 (SCHWEITZER ENGINEERING LABORATORIES, 2015).

4.2.2 Cenários de testes

O SPA foi testado em múltiplos cenários de operação da microrrede, considerando variações de conexão de geração distribuída, topologia da rede e ilhamento.

Todas as 16 combinações possíveis de conexão das quatro unidades de geração foram consideradas na análise. Na Tabela 10 são apresentados os casos de conexão e o seu respectivo código adotado (primeira coluna). O valor 1 significa que a fonte de GD está conectada e o valor 0 significa desconectada.

Tabela 10 – Casos de conexão de GD considerados.

Caso	Estado de Conexão				Caso	Estado de Conexão			
	GD1	GD2	GD3	GD4		GD1	GD2	GD3	GD4
A	1	1	1	1	I	1	0	0	1
B	0	1	1	1	J	1	0	1	0
C	1	0	1	1	K	1	1	0	0
D	1	1	0	1	L	0	0	0	1
E	1	1	1	0	M	1	0	0	0
F	0	0	1	1	O	0	1	0	0
G	0	1	0	1	P	0	0	1	0
H	0	1	1	0	Q	0	0	0	0

Fonte: O próprio autor.

Nos casos de mudança topológica foram testados os casos em que ocorre isolamento de um trecho em falta (ou em manutenção) e reenergização das cargas desenergizadas. Desta forma, foram considerados 9 casos possíveis de mudança topológica. Todos os casos considerados e seus códigos de identificação estão representados na Tabela 11.

Foram considerados também os casos em que a rede estava ilhada e conectada ao sistema elétrico. No entanto, na avaliação da microrrede isolada (chave 17 aberta), todos os casos mostrados na Tabela 10 foram considerados, exceto o caso Q, pois não haveria nenhuma fonte conectada ao sistema elétrico. Portanto, o SPA foi testado em 279 casos distintos de operação da microrrede.

4.2.2.1 Parâmetros de simulação

O SPA desenvolvido nesta etapa do trabalho foi desenvolvido em Python 3, devido à facilidade de integração com o sistema elétrico, por ser uma linguagem *open source* e com grande disponibilidade de módulos de comunicação externos. Além disso, a ferramenta de cálculo de

Tabela 11 – Casos de mudança topológica da rede.

Casos	Chaves abertas	
1	1	2
2	3	4
3	5	6
4	7	8
5	9	10
6	11	12
7	13	14
8	15	16
9	22	23

Fonte: O próprio autor.

curto circuito e fluxo de carga utilizada pelo SPA, a MyGrid, é desenvolvida em Python. Dessa forma, o desenvolvimento do SPA em Python torna a integração entre as duas ferramentas mais simples.

Por ser uma linguagem de programação interpretada, a Python tende a ser mais lenta na execução de *loops*. Para superar esta deficiência, o módulo Numba foi utilizado no desenvolvimento do algoritmo (LAM *et al.*, 2015). Este módulo traduz os códigos em Python para linguagem de máquina otimizado em tempo de execução.

O sistema foi executado em computador pessoal com processador Intel[®] Core[™] i5-3210M CPU @2.5 GHz x64 com 6 GB RAM.

Para garantir maior robustez ao AMDFA, este foi executado com 300 indivíduos na população por 1000 iterações.

4.2.3 Sistema com ajustes fixos otimizados

O SPA proposto teve o desempenho comparado com o de um Sistema com Ajustes Fixos (SAF). Para obter estes ajustes, o problema de coordenação foi modelado considerando todos os cenários citados anteriormente em uma única função objetivo. Desta forma, os ajustes obtidos são otimizados para todos os modos de operação da rede, por esse motivo podem ser mantidos fixos.

Devido a dimensão do problema ser muito grande, o AMDFA foi executado com 300 indivíduos na população durante 10 mil iterações. Desta forma, garante-se que o algoritmo converge para a solução boa o suficiente.

Foram consideradas como restrições apenas as restrições de ajustes dos relés, desconsiderando a correção da corrente de carga e corrente de falta mínima. Além disso, foi adotado um peso $\gamma = 500$ para a função de penalidade da função objetivo.

4.3 Comparação dos resultados obtidos pelo SPA e pelo sistema com ajustes fixos

Para avaliar cada caso especificado anteriormente, foram coletadas informações sobre o tempo do sistema de proteção principal, tempo de operação do sistema de proteção de *backup* e a descoordenação proporcionada.

Para cada caso de conexão de GD (Tabela 10), e cada caso de topologia (Tabela 11) foi avaliado o tempo de operação das proteções principal e *backup* providos pelo SPA e pelo SAF. Nos gráficos das Figuras 42 e 43 são apresentadas as médias dos tempos de operação da proteção principal e de *backup*, respectivamente, em cada caso de conexão de GD para o sistema conectado à rede e ilhado. Os gráficos com reta pontilhada apresentam a média geral de todos os casos para cada sistema.

Com relação à minimização do tempo de operação da proteção principal quando o sistema está conectado à rede, ambos os sistemas obtiveram desempenhos semelhantes. Nos casos com maior penetração de GD, o SAF apresentou melhor desempenho e nos casos com menor penetração de GD, o desempenho do SPA prevaleceu sobre o do SAF. Na média geral, o SAF proveu tempo de operação 0,2586% inferior em relação ao SPA, uma diferença ínfima.

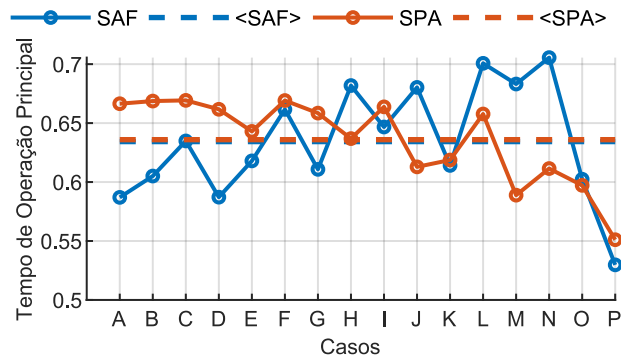
Por outro lado, quando o sistema está operando em modo ilhado, o SPA provê velocidade de operação muito maior ao sistema de proteção. Como pode ser observado na Figura 42 (b), as médias de tempo de operação foram inferiores em todos os casos de penetração de GD. A média geral de tempo de operação principal do SPA foi 26,58% inferior ao do SAF.

Quanto ao tempo de operação da proteção de *backup*, quando o sistema está conectado à rede, ambos os sistemas apresentaram comportamento semelhante nos casos com maior penetração de GD, porém nos casos com menor penetração, o SPA obteve desempenho melhor. Na média geral, o SPA proporcionou redução de tempo 4,77% em relação ao SAF.

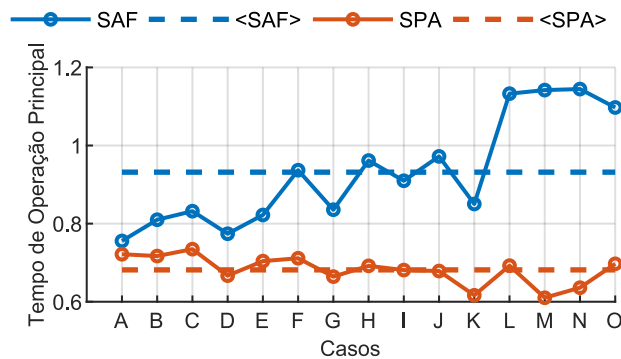
Quando o sistema está operando em modo ilhado, novamente os sistemas apresentaram desempenho semelhante nos casos com maior penetração de GD e houve prevalência do SPA, conforme houve redução da penetração de GD. Porém, na média geral o SPA obteve desempenho melhor que o SAF, com redução de tempo de 6,83%.

Devido ao SAF ser projetado para fornecer um conjunto de ajustes otimizados para

Figura 42 – Médias dos tempos de operação principal para o sistema conectado a rede (a) e ilhado (b).



(a)



(b)

Fonte: O próprio autor.

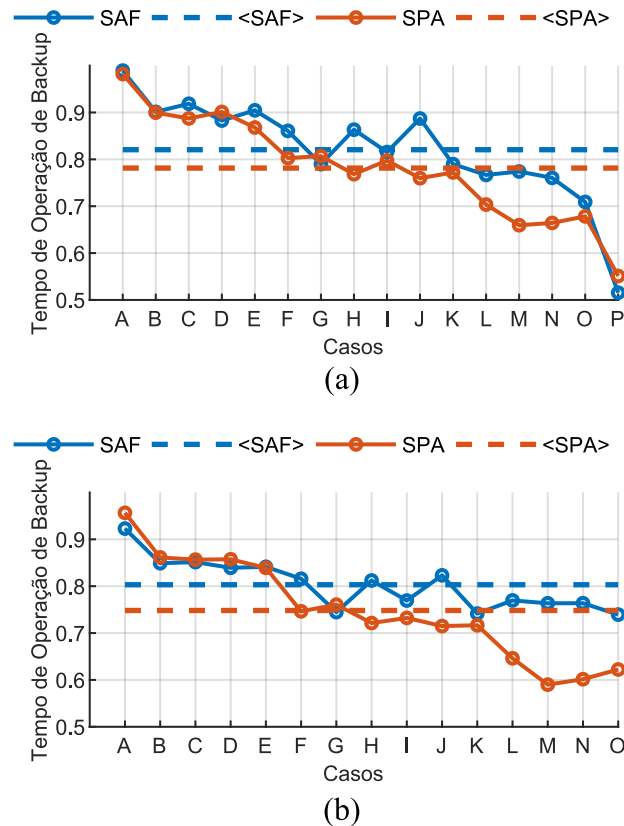
todos os casos de operação da microrrede, este não consegue fornecer um desempenho bom em cada caso de operação.

Além disso, quanto maior for o número de relés a serem coordenados em um caso, maior será a contribuição desse caso na redução da função objetivo. Desta forma, os ajustes fornecidos pelo SAF acabam sendo mais adequados aos casos em que a microrrede está operando em modo conectado e com maior penetração de geração distribuída. Por isso, o desempenho inferior do SAF em relação ao SPA quando a microrrede apresenta menor presença de GD e está operando em modo ilhado.

Outro fator considerado na análise, foi a descoordenação causada pelo SPA e pelo SAF para fornecer os ajustes otimizados. Para isto, foram coletados os ITCs de cada par de relé principal e *backup* para cada um dos 279 casos de simulação apresentados. O ITC mínimo adotado para ambos os sistemas foi 0,300 segundos.

Na Figura 44 é apresentada a distribuição de frequência dos ITCs obtidos pelo SPA e pelo SAF. Em ambos os gráficos, são apresentados a média (μ) e o desvio padrão (σ) obtidos

Figura 43 – Médias dos tempos de operação de *backup* para o sistema conectado a rede (a) e ilhado (b).



Fonte: O próprio autor.

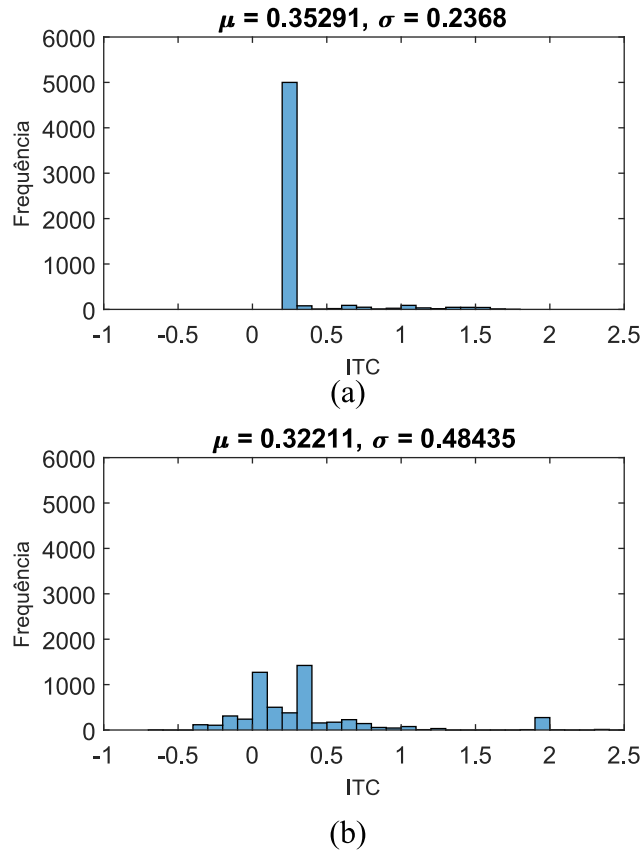
em cada distribuição.

Na média, ambos os sistemas respeitaram a restrição de coordenação, ficando acima do ITC mínimo de 0,3 segundos. No entanto, a média do SPA foi superior a do SAF e o desvio padrão do SPA foi menor que a do SAF. Ou seja, os ajustes fornecidos pelo SPA apresentam maior margem de ITC em relação à descoordenação e apresentam maior concentração em torno do valor médio.

Além disso, a maior parte (89,89 %) dos ITCs do SPA ficaram dentro do intervalo entre 0,2 e 0,3 segundos e nenhum valor abaixo deste intervalo. Considerando que o ITC pode ser adotado entre 0,2 e 0,5 segundos (ANDERSON, 1999), o SPA garantiu a coordenação do sistema em 91,3 % dos casos. Nos 8,7 % de casos restantes, o tempo de operação de *backup* apenas levaria mais que 0,5 segundos para atuar, ao invés de atuar descoordenadamente, uma situação mais agravante.

Em contrapartida, no SAF apenas 6,79 % dos ITCs ficaram entre 0,2 e 0,3 segundos e 35,16 % ficaram entre 0,2 e 0,5 segundos. Por outro lado, 45,68 % dos ITCs ficaram entre -0,4

Figura 44 – Distribuição de frequência dos ITCs obtidos pelo SPA (a) e SAF (b).



Fonte: O próprio autor.

e 0,2 segundos, ou seja, estavam descoordenados.

Logo, o SPA apresentou desempenho muito superior ao SAF na coordenação do sistema. O SAF enfrenta dificuldades na coordenação do sistema ao tentar fornecer um conjunto de ajustes que se adeque a todos os casos de operação da microrrede. Ao tentar coordenar alguns casos, acaba causando descoordenação nos demais casos.

Para realizar cada simulação, na média, o SPA levou 13,14 segundos. Para realizar todas as simulações e obter os ajustes, o SAF levou 4473,5 segundos (aproximadamente 1 hora e 15 minutos). A comparação de tempo entre os dois sistemas não é justa, uma vez que são propostas de aplicação distintas. No entanto, demonstra-se que o SPA é capaz de fornecer bons ajustes ao sistema de proteção em um tempo praticável. Considerando que o tempo máximo para uma recomposição automática seja de três minutos, o SPA levaria apenas 7,3 % deste tempo para corrigir os ajustes e permitir a recomposição do sistema.

4.4 Considerações finais

Conclui-se com este capítulo que o SPA apresenta melhor desempenho que um sistema de proteção com ajustes fixos otimizados, uma vez que seus ajustes permitiram maior rapidez ao sistema elétrico sem perda de confiabilidade.

Além disso, o SPA demonstrou-se viável do ponto de vista prático, uma vez que só necessitou de 13,14 segundos para calcular os ajustes de uma microrrede com 23 relés de sobrecorrente direcionais.

Para operar, o SPA necessita que exista uma rede de comunicação com os relés de proteção da rede, haja monitoramento das chaves e das fontes de geração distribuída. Em subestações modernas, já são implementados sistemas que contemplam o monitoramento das chaves e a comunicação com os relés de proteção. Além disso, é previsto que as redes de distribuição futuras apresentem amplo monitoramento da rede, inclusive das suas fontes de geração. Logo, devido a sua arquitetura simples, o SPA também apresenta facilidade de integração ao sistema elétrico.

5 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho propôs um sistema de proteção adaptativa aplicado a microrredes. O sistema altera os ajustes do sistema de proteção sempre que há: conexão ou desconexão de geração distribuída; mudança topológica da rede; e ilhamento ou conexão à rede elétrica principal.

O sistema proposto adapta os ajustes do sistema de proteção de sobrecorrente direcional. Para o cálculo dos ajustes, o sistema aplica uma metaheurística para obter os ajustes do sistema de proteção.

O algoritmo foi testado em três redes testes de referência e teve o desempenho comparado ao desempenho dos seguintes algoritmos: Algoritmo Genético, Otimização por Enxame de Partículas, Evolução Diferencial, Algoritmo do Morcego e Algoritmo do Morcego Direcional. Os resultados demonstraram que o algoritmo proposto superou os demais algoritmos testados em todos os critérios, comprovando-se a sua aptidão para aplicações de proteção adaptativa.

Além disso, o SPA foi testado em uma microrrede em múltiplos cenários de penetração de geração distribuída, mudança topológica e ilhamento. O desempenho do SPA foi comparado ao desempenho de um sistema de proteção convencional com ajustes fixos. Os resultados demonstraram que os ajustes fornecidos pelo SPA tornaram o sistema de proteção mais rápido e proveram melhor coordenação ao sistema elétrico em relação aos ajustes fornecidos pelo sistema com ajustes fixos.

Os resultados obtidos nos testes demonstraram que o algoritmo de otimização proposto e projetado neste trabalho para aplicação em sistemas de proteção adaptativa apresenta rápida convergência, boa adesão às restrições de coordenação, alta robustez e tempo de execução factível para aplicações em tempo real.

Dessa forma, conclui-se que a metaheurística proposta apresenta características ideais para sistemas de proteção adaptativa, superando outros algoritmos difundidos na literatura. Além disso, o SPA proposto mostrou-se ser uma ferramenta poderosa para mitigar os impactos da operação de microrredes no sistema de proteção.

Visando melhorias no presente trabalho, são sugeridos as seguintes considerações para trabalhos futuros:

- analisar a viabilidade de expansão da modelagem do problema de coordenação, considerando o modo de operação instantâneo na curva do tempo de operação dos relés;
- estudar metodologia para cálculo ótimo de grupos de ajustes e analisar a sua viabilidade

frente a metodologia proposta;

- incluir a consideração de alocação de limitadores de corrente de falta na microrrede e analisar o efeito na coordenação ótima do sistema de proteção.
- Incluir no modelo do problema, a consideração de outros tipos de proteção empregados nos sistemas de distribuição, como: proteção de distância, fusíveis, etc.

5.1 Trabalhos Publicados

A seguir estão listados os trabalhos publicados e submetidos durante o período de desenvolvimento do trabalho.

- DA SILVA, K. Q.; SAMPAIO, F. C.; MAGALHÃES, L. F.; ANDRADE, A. A.; SAMPAIO, R. F.; LEÃO, R. P. S.; MELO, L. S.; BARROSO, G. C. An Adaptive Protection System for Distribution Network with Distributed Generation. In: Simpósio Brasileiro de Sistemas de Energia 2018, Nitéroí, 2018.
- JUREMA, L. M. C.; LIMA JUNIOR, A. A.; VIEIRA M. C.; SAMPAIO, F. C.; LEÃO, R. P. S.; SAMPAIO, R. F.; MELO, L. S.; BARROSO, G. C. Network Configurator and Simulator System Based on CIM Standard and Node-Depth Encoding. In: Congresso Brasileiro de Automática 2018, João Pessoa, 2018.
- SAMPAIO, F. C.; BELMINO, L. M.; LEÃO, R. P. S.; SAMPAIO, R. F.; MELO, L. S.; BARROSO, G. C. Optimal Directional Overcurrent Relays Coordination using Direcional Bat Algorithm. In: International Smart Grid Technologies - Latin America 2019, Gramado, 2019.
- SAMPAIO F. C.; DA FONSECA, J. M. L.; BARNABÉ, G. P.; MACHADO, L. A.; LEÃO, R. P. S.; SAMPAIO, R. F. Comparação dos Métodos de Monte Carlo e Estimação por Pontos para Avaliação de VTCD em Rede de Distribuição com Geração PV e Eólica. In: XXV Seminário Nacional de Produção e Transmissão de Energia Elétrica, Belo Horizonte, 2019.
- SAMPAIO, F. C.; LEÃO, R. P. S.; SAMPAIO, R. F.; MELO, L. S.; BARROSO; G. C. A Multi-Agent-Based Integrated Self-Healing and Adaptive Protection System for Power Distribution Systems with Distributed Generation. Electric Power Systems Research. ISSN 0378-7796. (em processo de revisão)

REFERÊNCIAS

- AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA. **PRODIST: Módulo 8 - Qualidade de Energia Elétrica (Revisão 10)**. Brasília, 2015. 88 p.
- ALAM, M. N. Adaptive Protection Coordination Scheme using Numerical Directional Overcurrent Relays. **IEEE Transactions on Industrial Informatics**, v. 3203, n. c, p. 1–10, 2018. ISSN 15513203.
- ALBASRI, F. A.; ALROOMI, A. R.; TALAQ, J. H. Optimal Coordination of Directional Overcurrent Relays Using Biogeography-Based Optimization Algorithms. **IEEE Transactions on Power Delivery**, v. 30, n. 4, p. 1810–1820, 2015. ISSN 08858977.
- ALMEIDA, R. G. **Gerenciamento e Controle Distribuído Hierárquico Primário e Secundário Aplicado a Microrrede CA**. Dissertação (Mestrado) — Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Ceará, 2019.
- AMRAEE, T. Coordination of directional overcurrent relays using seeker algorithm. **IEEE Transactions on Power Delivery**, v. 27, n. 3, p. 1415–1422, 2012. ISSN 08858977.
- ANDERSON, P. M. **Power system protection**. New York: McGraw-Hill IEEE Press, 1999. ISBN 0780334272.
- BEDEKAR, P. P.; BHIDE, S. R. Optimum coordination of directional overcurrent relays using the hybrid GA-NLP approach. **IEEE Transactions on Power Delivery**, v. 26, n. 1, p. 109–119, 2011. ISSN 08858977.
- BIRLA, D.; MAHESHWARI, R. P.; GUPTA, H. O. A new nonlinear directional overcurrent relay coordination technique, and banes and boons of near-end faults based approach. **IEEE Transactions on Power Delivery**, v. 21, n. 3, p. 1176–1182, July 2006. ISSN 0885-8977.
- BOTTURA, F. B. **Algoritmo de Coordenação Híbrido para Coordenação de Relés Direcionais de Sobrecorrente em um Sistema Elétrico Malhado**. Dissertação (Mestrado) — Escola de Engenharia de São Carlos, Universidade de São Paulo, 2014.
- BREARLEY, B. J.; PRABU, R. R. A review on issues and approaches for microgrid protection. **Renewable and Sustainable Energy Reviews**, Elsevier, v. 67, p. 988–997, 2017. ISSN 18790690.
- BRITISH PETROLEUM. **BP Statistical Review of World Energy**. [S.l.], 2019. 64 p. (68).
- CHAKRI, A.; KHELIF, R.; BENOURET, M.; YANG, X. S. New directional bat algorithm for continuous optimization problems. **Expert Systems with Applications**, Elsevier Ltd, v. 69, p. 159–175, 2017. ISSN 09574174.
- CHATTOPADHYAY B.; SACHDEV, M. S. S. T. S. An on-line relay coordination algorithm for adaptive protection using linear programming technique. **IEEE Transactions on Power Delivery**, v. 11, n. 1, p. 165–173, Jan 1996. ISSN 1937-4208.
- CHELLIAH, T. R.; THANGARAJ, R.; ALLAMSETTY, S.; PANT, M. Coordination of directional overcurrent relays using opposition based chaotic differential evolution algorithm. **International Journal of Electrical Power and Energy Systems**, Elsevier Ltd, v. 55, p. 341–350, 2014. ISSN 01420615.

CHOWDHURY, S.; CHOWDHURY, S.; CROSSLEY, P. **Microgrids and active distribution networks**. London: Institution of Engineering and Technology, 2009. ISBN 978-1849190145.

COFFELE, F.; BOOTH, C.; DYSKO, A. An Adaptive Overcurrent Protection Scheme for Distribution Networks. **IEEE Transactions on Power Delivery**, v. 30, n. 2, p. 561–568, 2015. ISSN 08858977.

COLMENAR-SANTOS, A.; REINO-RIO, C.; BORGE-DIEZ, D.; COLLADO-FERNÁNDEZ, E. Distributed generation: A review of factors that can contribute most to achieve a scenario of DG units embedded in the new distribution networks. **Renewable and Sustainable Energy Reviews**, Elsevier, v. 59, p. 1130–1148, 2016. ISSN 18790690.

DEHGHANPOUR, E.; Kazemi Karegar, H.; KHEIROLLAHI, R.; SOLEYMANI, T. Optimal coordination of directional overcurrent relays in microgrids by using cuckoo-linear optimization algorithm and fault current limiter. **IEEE Transactions on Smart Grid**, v. 9, n. 2, p. 1365–1375, 2018. ISSN 19493053.

EHSAN, A.; YANG, Q. Optimal integration and planning of renewable distributed generation in the power distribution networks: A review of analytical techniques. **Applied Energy**, Elsevier, v. 210, n. July 2017, p. 44–59, 2018. ISSN 03062619.

ELRAFIE, H. B.; IRVING, M. R. Linear programming for directional overcurrent relay coordination in interconnected power systems with constraint relaxation. **Electric Power Systems Research**, v. 27, n. 3, p. 209–216, 1993. ISSN 03787796.

ENERDATA. **Electricity Domestic Consumption**. 2019. Disponível em: <<https://yearbook.enerdata.net/electricity/electricity-domestic-consumption-data.html>>. Acesso em: 23 set. 2019.

GERS, J. M.; HOLMES, E. J. **Protection of Electricity Distribution Networks**. 3. ed. [S.l.]: Institution of Engineering and Technology, 2011. ISBN 9781849192231.

GREI. **MyGrid**. 2019. Disponível em: <<https://github.com/grei-ufc/MyGrid>>.

HATZIARGYRIOU, N. **Microgrids: Architectures and Control**. [S.l.]: Wiley, 2014. (Wiley - IEEE). ISBN 9781118720684.

HOSSAIN, M. A.; POTA, H. R.; HOSSAIN, M. J.; BLAABJERG, F. Evolution of microgrids with converter-interfaced generations: Challenges and opportunities. **International Journal of Electrical Power and Energy Systems**, Elsevier, v. 109, n. January, p. 160–186, 2019. ISSN 01420615. Disponível em: <<https://doi.org/10.1016/j.ijepes.2019.01.038>>.

HUSSAIN, M. H.; RAHIM, S. R.; MUSIRIN, I. Optimal overcurrent relay coordination: A review. **Procedia Engineering**, Elsevier B.V., v. 53, p. 332–336, 2013. ISSN 18777058.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. **Electrical Relays - Single Input Energizing Quantity Measuring Relays with Dependent or Independent Time**. Londres, 1989. 19 p.

JAIN, R.; LUBKEMAN, D. L.; LUKIC, S. M. Dynamic Adaptive Protection for Distribution Systems in Grid-Connected and Islanded Modes. **IEEE Transactions on Power Delivery**, IEEE, v. 34, n. 1, p. 281–289, 2019. ISSN 08858977.

JEVTIĆ, M.; RADOSAVLJEVIĆ, J. Hybrid GSA-SQP algorithm for optimal coordination of directional overcurrent relays. **IET Generation, Transmission & Distribution**, v. 10, n. 8, p. 1928–1937, 2016. ISSN 1751-8687. Disponível em: <<http://digital-library.theiet.org/content/journals/10.1049/iet-gtd.2015.1223>>.

KINDERMANN, G. **Proteção de Sistema Elétricos de Potência 1**. 2005. 148 p.

LAAKSONEN, H.; ISHCENKO, D.; OUDALOV, A. Adaptive Protection and Microgrid Control Design for Hailuoto Island. **IEEE Transactions on Smart Grid**, v. 5, n. 3, p. 1486–1493, 2014. ISSN 19493053.

LAM, S. K.; PITROU, A.; SEIBERT, S. Numba: A llvm-based python jit compiler. In: **PROCEEDINGS OF THE SECOND WORKSHOP ON THE LLVM COMPILER INFRASTRUCTURE IN HPC**. New York, NY, USA: ACM, 2015. (LLVM '15), p. 7:1–7:6. ISBN 978-1-4503-4005-2. Disponível em: <<http://doi.acm.org/10.1145/2833157.2833162>>.

LASSETER, R. H. Microgrids. In: **2002 IEEE POWER ENGINEERING SOCIETY WINTER MEETING. CONFERENCE PROCEEDINGS**. [S.l.: s.n.], 2002. v. 1, p. 305–308 vol.1.

LIU, Z.; SU, C.; HØIDALEN, H. K.; MEMBER, S.; CHEN, Z.; MEMBER, S. A Multiagent System-Based Protection and Control Scheme for Distribution System With Distributed-Generation Integration. **IEEE Transactions on Power Delivery**, v. 32, n. 1, p. 536–545, 2017.

MAHAT, P.; CHEN, Z.; BAK-JENSEN, B.; BAK, C. L. A simple adaptive overcurrent protection of distribution systems with distributed generation. **IEEE Transactions on Smart Grid**, v. 2, n. 3, p. 428–437, 2011. ISSN 19493053.

MANDITEREZA, P. T.; BANSAL, R. Renewable distributed generation: The hidden challenges - A review from the protection perspective. **Renewable and Sustainable Energy Reviews**, Elsevier, v. 58, p. 1457–1465, 2016. ISSN 18790690.

MANSOUR, M. M.; MEKHAMER, S. F.; EL-KHARBAWE, N. E. S. A modified particle swarm optimizer for the coordination of directional overcurrent relays. **IEEE Transactions on Power Delivery**, v. 22, n. 3, p. 1400–1410, 2007. ISSN 08858977.

NAJY, W. K.; ZEINELDIN, H. H.; WOON, W. L. Optimal protection coordination for microgrids with grid-connected and islanded capability. **IEEE Transactions on Industrial Electronics**, v. 60, n. 4, p. 1668–1677, 2013. ISSN 02780046.

NEGNEVITSKY, M. **Artificial intelligence : a guide to intelligent systems**. Harlow, England New York: Addison Wesley/Pearson, 2011. ISBN 978-1408225745.

NOGHABI, A. S.; MASHHADI, H. R.; SADEH, J. Optimal Coordination of Directional Overcurrent Relays Considering Different Network Topologies Using Interval Linear Programming. **IEEE Transactions on Power Delivery**, v. 25, n. 3, p. 1348–1354, 2010. ISSN 0885-8977.

NOGHABI, A. S.; SADEH, J.; MASHHADI, H. R. Considering different network topologies in optimal overcurrent relay coordination using a hybrid GA. **IEEE Transactions on Power Delivery**, v. 24, n. 4, p. 1857–1863, 2009. ISSN 08858977.

OJAGHI, M.; MOHAMMADI, V. Use of Clustering to Reduce the Number of Different Setting Groups for Adaptive Coordination of Overcurrent Relays. **IEEE Transactions on Power Delivery**, v. 33, n. 3, p. 1204–1212, 2018. ISSN 08858977.

PAPASPILIOTOPOULOS, V. A.; KORRES, G. N.; KLEFTAKIS, V. A.; HATZIARGYRIOU, N. D. Hardware-In-the-Loop Design and Optimal Setting of Adaptive Protection Schemes for Distribution Systems With Distributed Generation. **IEEE Transactions on Smart Grid**, v. 32, n. 1, p. 393–400, 2017. ISSN 0885-8977.

SAMPAIO, R. F.; MELO, L. S.; LEÃO, R. P. S.; BARROSO, G. C.; BEZERRA, J. R. Automatic restoration system for power distribution networks based on multi-agent systems. **IET Generation, Transmission Distribution**, v. 11, n. 2, p. 475–484, 2017. ISSN 1751-8695.

SCHWEITZER ENGINEERING LABORATORIES. **SEL-751 Feeder Protection Relay - Instruction Manual**. 1. ed. Washington, 2015.

SHIH, M. Y.; Conde Enríquez, A.; HSIAO, T. Y.; Torres Treviño, L. M. Enhanced differential evolution algorithm for coordination of directional overcurrent relays. **Electric Power Systems Research**, v. 143, p. 365–375, 2017. ISSN 03787796.

SHIH, M. Y.; Conde Enríquez, A.; Torres Treviño, L. M. On-line coordination of directional overcurrent relays: Performance evaluation among optimization algorithms. **Electric Power Systems Research**, v. 110, p. 122–132, 2014. ISSN 03787796.

SINGH, M.; VISHNUVARDHAN, T.; SRIVANI, S. G. Adaptive protection coordination scheme for power networks under penetration of distributed energy resources. **IET Generation, Transmission & Distribution**, p. 3919–3929, 2016. ISSN 17518687.

SO, C. W.; LI, K. K.; LAI, K. T.; FUNG, K. Y. Application of genetic algorithm for overcurrent relay coordination. In: **SIXTH INTERNATIONAL CONFERENCE ON DEVELOPMENTS IN POWER SYSTEMS PROTECTION**. [S.l.: s.n.], 1997. p. 66–69. ISSN 0537-9989.

SRINIVAS, S. T. P.; K, S. S. Application of improved invasive weed optimization technique for optimally setting directional overcurrent relays in power systems. **Applied Soft Computing Journal**, Elsevier B.V., v. 79, p. 1–13, 2019. ISSN 1568-4946. Disponível em: <<https://doi.org/10.1016/j.asoc.2019.03.045>>.

TALBI, E.-G. **Metaheuristics : from design to implementation**. Hoboken, N.J: John Wiley & Sons, 2009. ISBN 978-0470278581.

URDANETA, A.; NADIRA, R.; Perez Jimenez, L. Optimal coordination of directional overcurrent relays in interconnected power systems. **IEEE Transactions on Power Delivery**, v. 3, n. 3, p. 903–911, jul 1988. ISSN 08858977. Disponível em: <<http://ieeexplore.ieee.org/document/193867/>>.

VATANI, M. R.; NADERI, M. S.; GHAREHPETIAN, G. B.; SANJARI, M. J.; Solati Alkaran, D. Optimal Overcurrent Relay Coordination in Interconnected Networks by Using Fuzzy-Based GA Method. **IEEE Transactions on Smart Grid**, v. 9, n. 4, p. 3091–3101, 2016. ISSN 1949-3053.

YANG, X.-S. A new metaheuristic bat-inspired algorithm. In: **Nature inspired cooperative strategies for optimization (NICSO 2010)**. [S.l.]: Springer, 2010. p. 65–74.

ZADEH, L. A. Fuzzy Sets.pdf. **Information and Control**, v. 8, p. 338–353, 1965. ISSN 00199958.

ZEIENLDIN, H.; EL-SAADANY, E. F.; SALAMA, M. A. A novel problem formulation for directional overcurrent relay coordination. In: **2004 Large Engineering Systems Conference on Power Engineering**. [S.l.: s.n.], 2004. p. 48–52.

ANEXO A – DADOS DE CURTO CIRCUITO DAS REDES TESTES

Neste anexo são apresentadas as correntes de curto-circuito nos pares de relés principal/retaguarda em cada caso das redes de referência no Capítulo 3. Nas tabelas são apresentados os Relés Principais (RP), Relés de Retaguarda (RR), Correntes nos Relés Principais (IP), Correntes nos Relés de Retaguarda (IR).

A.1 Curto-circuito na rede 3 barras

Tabela 12 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 3 barras.

RP	RR	IP	IR
1	5	1978.9	175
2	4	1525.7	545
3	1	1683.9	617.22
4	6	1815.4	466.17
5	3	1499.7	384
6	2	1766.3	145.34

Fonte: O próprio autor.

A.2 Curto-circuito na rede 9 barras

Neste caso, os relés de retaguarda representados como 0 significa que o relé principal não possui proteção de retaguarda.

Tabela 13 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 9 barras.

RP	RP	IP	IR	RP	RR	IP	IR
1	15	4863.6	1168.3	14	16	4172.5	1031.7
1	17	4863.6	1293.9	14	19	4172.5	1264.1
2	4	1634.4	1044.2	15	13	4172.5	1031.7
3	1	2811.4	1361.6	15	19	4172.5	1264.1
4	6	2610.5	1226	16	2	3684.5	653.6
5	3	1778	1124.4	16	17	3684.5	1293.9
6	8	4378.5	711.2	17	0	7611.2	0
6	23	4378.5	1345.5	18	2	2271.7	653.6
7	5	4378.5	711.2	18	15	2271.7	1168.3
7	23	4378.5	1345.5	19	0	7435.8	0
8	10	1778	1124.4	20	13	2624.2	1031.7
9	7	2610.5	1226	20	16	2624.2	1031.7
10	12	2811.4	787.2	21	0	7611.2	0
11	9	1634.4	1044.2	22	11	2271.7	653.6
12	14	2811.4	1168.3	22	14	2271.7	1168.3
12	21	2811.4	1293.9	23	0	7914.7	0
13	11	3684.5	653.6	24	5	1665.5	711.2
13	21	3684.5	1293.9	24	8	1665.5	711.2

Fonte: O próprio autor.

A.3 Curto-circuito na rede 30 barras

Tabela 14 – Correntes de curto circuito nos pares de relés principal e retaguarda na rede 30 barras.

RP	RR	IP	IR	RP	RR	IP	IR
3	1	4089.7	4086.7	10	20	7339.3	1095.8
4	2	5411.2	2138.8	1	21	7665.3	698.8
22	2	4333	2147	9	21	7212.6	721.2
4	3	5411.2	3243.6	10	21	7339.3	716.1
5	4	4960.8	3001.3	20	22	3481.5	3481.5
18	4	4719.4	3002.1	21	23	5411.8	2193.5
6	5	2416	2416	22	23	4333	2204.6
7	6	5669	1790.9	18	24	4719.4	1717.7
8	6	5607.7	1774.8	23	24	3689.7	1724.2
27	7	1472.3	1472.3	24	25	2695	2695
26	8	1026.8	1026.8	1	28	7665.3	1552
12	9	5034.9	5034.9	2	28	7985.7	1545.8
11	10	3457.1	3457.1	10	28	7339.3	1538
13	11	3727.3	2875	1	29	7665.3	1380.6
14	12	2906.5	2906.5	2	29	7985.7	1375.2
15	13	2660.5	2660.5	9	29	7212.6	1379
16	14	6185.6	1668.1	29	30	2518.9	2518.9
17	14	7492.9	1641.1	28	31	2036.8	2036.8
19	15	5445.2	1527.3	30	32	2998.8	2149
35	15	4222	1533.2	31	33	3263.6	3263.6
36	15	6420.2	1509.7	32	34	2930.4	2930.4
19	16	5445.2	3128.3	17	35	7492.9	1885.4
34	16	5796.6	3123.9	33	35	6456.2	1954.5
36	16	6423.2	3052.4	16	36	6185.6	490.9
19	17	5445.2	801.3	33	36	6456.2	500.6
34	17	5796.6	800.1	5	37	4960.8	1961
35	17	4222	794	23	37	3689.7	1968.5
38	18	3133.2	2292.2	34	38	5796.6	1886.8
37	19	3788.9	2940.9	35	38	4222	1896.7
2	20	7985.7	1053.9	36	38	6420.2	1867.7
9	20	7212.6	1103.5	-	-	-	-

Fonte: O próprio autor.

ANEXO B – CÓDIGO EM MATLAB DAS METAHEURÍSTICAS IMPLEMENTADAS

B.1 Algoritmo Genético

```

1 function [best, fmin, conv, conv_md] = ag_oc(popsize, nvar,
      t_max, mr, selection, LB, UB, w1, w2, ff_parameters)
2
3 % Fitness Function Parameters
4
5 a = ff_parameters.a;
6 b = ff_parameters.b;
7 faults = ff_parameters.faults;
8 cti = ff_parameters.cti;
9 ctr = ff_parameters.ctr;
10
11 % GA Parameters
12
13 keep = floor(selection*popsize); % number of survivors
14 nmut = ceil(popsize*nvar*mr); % number of mutations
15 M = ceil((popsize-keep)/2); % number of matings
16
17 % Create the initial population
18
19 pop = zeros(popsize, nvar);
20 fitness = zeros(popsize, 1);
21
22 for i=1:popsize
23
24     pop(i,:) = LB + (UB - LB).*rand(1, nvar);
25     fitness(i) = fitness_function(faults, pop(i,:), ctr, a, b,
      cti, w1, w2);
26
27 end

```

```

28
29 [fitness,index] = sort(fitness);      % min cost in element 1
30 pop = pop(index,:);                  % sort continuous
31
32
33 % ----- Rank Weighting
34     ----- %
35 % weights chromossomes
36 prob=flipud([1:keep] / sum([1:keep]));
37
38 % probability distribution function
39 odds = [0 cumsum(prob(1:keep)) ];
40
41 t=0;                                  % initialize the generations
42     counter
43 conv = zeros(1,t_max);                % best fitness per generation
44 conv_md = zeros(1,t_max);
45
46 progressbar(0)
47 while t < t_max
48
49     progressbar(t/t_max)
50
51     t=t+1;                            % updates the generation
52
53     conv(t) = fitness(1);             % add the new fitness
54     conv_md(t) = mean(fitness);
55
56 %----- Pairing - Rank Weighting
57     ----- %

```



```
57
58 pick1 = rand(1,M);
59 pick2 = rand(1,M);
60
61 % ma and pa contain the indices of the chromosomes
   that will
62 % mate
63 ma = zeros(1,M);
64 pa = zeros(1,M);
65 ic=1;
66
67 while ic <= M
68
69     for id=2:keep+1      % sweep the odds vector
70
71         if pick1(ic)<=odds(id) & pick1(ic) > odds(id-1)
72
73             ma(ic) = id-1;
74
75         end
76
77         if pick2(ic)<=odds(id) & pick2(ic) > odds(id-1)
78
79             pa(ic) = id-1;
80
81         end
82     end
83     ic=ic+1;
84 end
85
86 % ----- Mating (single point crossover)
   ----- %
```

```
87
88     ix = 1:2:keep;                % index of mates
89     xp = ceil(rand(1,M)*nvar);    % crossover point
90     r = rand(1,M);              % mixing parameter
91
92
93     for ic=1:M
94
95         xy = pop(ma(ic),xp(ic))-pop(pa(ic),xp(ic));
96
97         % 1st offspring
98
99         pop(keep+ix(ic),:) = pop(ma(ic),:);
100        pop(keep+ix(ic),xp(ic)) = pop(ma(ic),xp(ic))-r(ic)
101        .*xy;
102
103        % 2nd offspring
104        pop(keep+ix(ic)+1,:) = pop(pa(ic),:);
105        pop(keep+ix(ic)+1,xp(ic)) = pop(pa(ic),xp(ic))+r(ic)
106        .*xy;
107
108        if xp(ic) < nvar
109
110            % 1st offspring
111
112            pop(keep+ix(ic),:) = [pop(keep+ix(ic),1:xp(ic))
113            ...
114            pop(keep+ix(ic)+1,xp(ic)+1:
115            nvar)];
```

```
115         % 2nd offspring
116         pop(keep+ix(ic)+1,:) = [pop(keep+ix(ic)+1,1:xp(
            ic)) ...
117                                 pop(keep+ix(ic),xp(ic)+1:
                                    nvar)];
118
119         end
120
121     end % for
122
123     % Repairing
124
125     for i=1:popsiz
126
127         for j=1:nvar
128
129             if pop(i,j) > UB(j)
130
131                 pop(i,j) = UB(j);
132
133             end
134
135             if pop(i,j) < LB(j)
136
137                 pop(i,j) = LB(j);
138
139             end
140
141         end
142
143     end
144
```

```
145 % ----- Mutation
146 % -----
147
148 mrow=sort(ceil(rand(1,nmut)*(popsize-1))+1);
149 mcol=ceil(rand(1,nmut)*nvar);
150
151 for ii = 1:nmut
152
153     lin = mrow(ii);col = mcol(ii);
154     pop(lin,col) = (UB(col)-LB(col))*rand()+...
155                     LB(mcol(ii));
156
157 end
158
159 % Update Fitness
160
161 for i=1:popsize
162
163     fitness(i) = fitness_function(faults,pop(i,:),ctr,a
164                                   ,b,cti,w1,w2);
165
166
167     [fitness,index] = sort(fitness);
168     pop = pop(index,:);
169
170 end % while
171
172 fmin = fitness(1);
173 best = pop(1,:);
174
```

```

175     progressbar(1)
176
177 end

```

B.2 Otimização por Enxame de Partículas

```

1 function [best, fmin, conv, conv_md] = pso_oc(n, nvar, niter, w,
      c1, c2, tol, LB, UB, w1, w2, ff_parameters)
2
3 % Fitness Function Parameters
4
5 a = ff_parameters.a;
6 b = ff_parameters.b;
7 faults = ff_parameters.faults;
8 cti = ff_parameters.cti;
9 ctr = ff_parameters.ctr;
10
11 % Initial population and fitness
12
13 x = zeros(n, nvar);
14 fitness = zeros(n, 1);
15
16 for i=1:n
17
18     x(i,:) = LB + (UB - LB).*rand(1, nvar);
19
20     fitness(i) = fitness_function(faults, x(i,:), ctr, a, b, cti
      , w1, w2);
21
22 end
23
24 % Initializes the velocity

```

```
25
26 v = zeros(n,nvar);
27
28 % Find the current best
29
30 [fitness_gbest,index] = min(fitness);
31 gbest = x(index,:);
32 pbest = x;
33 fitness_pbest = fitness;
34
35 % Start the iteration
36
37 w0 = w.w0;
38 wf = w.wf;
39
40 t = 0;
41 conv=zeros(1,niter);
42 conv_md=zeros(1,niter);
43 progressbar(0)
44
45 while t < niter
46
47     progressbar(t/niter);
48
49     t = t + 1;
50
51     conv(t) = fitness_gbest;
52     conv_md(t) = mean(fitness_pbest);
53
54     % Updates velocity
55
56     for i=1:n
```

```
57
58     w = w0 - (w0-wf)*t/niter;
59
60     v(i,:) = w*v(i,:) + ...
61             c1.*(gbest - x(i,:)).*rand(1,nvar) + ...
62             c2.*(pbest(i,:) - x(i,:)).*rand(1,nvar);
63
64     end
65
66     % Updates position
67
68     for i=1:n
69
70         x(i,:) = x(i,:) + v(i,:);
71
72     end
73
74     % Handle Boundary Violation
75
76     for i=1:n
77
78         for j=1:nvar
79
80             if x(i,j) < LB(j)
81
82                 x(i,j) = LB(j);
83
84             end
85
86             if x(i,j) > UB(j)
87
88                 x(i,j) = UB(j);
```

```
89
90         end
91
92     end
93
94     % Updates fitness
95
96     fitness(i) = fitness_function(faults,x(i,:),ctr,a,b
97         ,cti,w1,w2);
98
99     end
100
101     % Updates pbest and fitness_pbest
102
103     for i=1:n
104
105         if fitness(i) < fitness_pbest(i)
106
107             pbest(i,:) = x(i,:);
108             fitness_pbest(i) = fitness(i);
109
110         end
111
112     end
113
114     % Updates gbest
115
116     [fmin,index] = min(fitness_pbest);
117
118     if fmin < fitness_gbest
119
120         gbest = pbest(index,:);
```



```
120         fitness_gbest = fmin;
121
122     end
123
124 end
125
126     best = gbest;
127     fmin = fitness_gbest;
128
129     progressbar(1);
130
131 end
```

B.3 Evolução Diferencial

```
1 function [best,fmin,conv,conv_md,counters] = de_oc(n,nvar,F
    ,cr,niter,LB,UB,w1,w2,ff_parameters)
2
3 % Fitness Function Parameters
4
5 a = ff_parameters.a;
6 b = ff_parameters.b;
7 faults = ff_parameters.faults;
8 cti = ff_parameters.cti;
9 ctr = ff_parameters.ctr;
10
11 % Initialize the population/solutions
12
13 pop = zeros(n,nvar);
14 fitness = zeros(1,n);
15
16 for i=1:n
```

```
17
18     pop(i,:) = LB + (UB-LB).*rand(1,nvar);
19     fitness(i) = fitness_function(faults ,pop(i,:),ctr,a,b,
20         cti,w1,w2);
21 end
22
23 % Find the current best
24
25 [fmin,index] = min(fitness);
26 best = pop(index,:);
27
28 t = 0;
29 conv = zeros(1,niter);
30 conv_md = zeros(1,niter);
31
32 progressBar(0)
33
34 c_div = zeros(1,niter);
35 c_erro = zeros(1,niter);
36
37 while t < niter
38
39     progressBar(t/niter)
40
41     t = t+1;    % updates generation
42
43     % updates conv
44     conv(t) = fmin;
45     conv_md(t) = mean(fitness);
46
47     % Obtain donor vectors by permutation
```

```
48
49 % Random permutation
50
51 index1 = randperm(n);
52 pop1 = pop(index1,:);
53
54 index2 = randperm(n);
55 pop2 = pop(index2,:);
56
57 K = rand(n,nvar) < cr;
58
59 % DE/Rand/1
60
61 V = pop + F*(pop1 - pop2); % donor vector
62 V = pop.*(1-K) + V.*K; % crossover with donor
   vector
63
64 % Bounds handling
65
66 for i=1:n
67
68     for j=1:nvar
69
70         if V(i,j) < LB(j)
71
72             V(i,j) = LB(j);
73
74         end
75
76         if V(i,j) > UB(j)
77
78             V(i,j) = UB(j);
```

```
79
80         end
81
82     end
83
84 end
85
86 % Evaluate new solutions and update the population
87
88 for i=1:n
89
90     fnew = fitness_function(faults,V(i,:),ctr,a,b,cti,
91                             w1,w2);
92
93     if fnew <= fitness(i)
94
95         pop(i,:) = V(i,:);
96         fitness(i) = fnew;
97
98     end
99
100 end
101
102 % Update the current best
103
104 [fmin,index] = min(fitness);
105 best = pop(index,:);
106
107 c_div(t) = diversity(pop,best);
108 c_erro(t) = erro(fitness,fmin);
109 end % while
```

```

110
111 counters.div = c_div;
112 counters.error = c_erro;
113
114 progressbar(1)
115
116 end

```

B.4 Algoritmo do Morcego

```

1 function [best, fmin, conv, conv_md, counters] = ba_oc(n, nvar,
    Ao, ro, alfa, gamma, Qmin, Qmax, niter, LB, UB, w1, w2,
    ff_parameters)
2
3 % Fitness Function Parameters
4
5 a = ff_parameters.a;
6 b = ff_parameters.b;
7 faults = ff_parameters.faults;
8 cti = ff_parameters.cti;
9 ctr = ff_parameters.ctr;
10
11 r = 0.01*ones(n,1);
12 A = Ao*ones(n,1);
13
14
15 v = zeros(n, nvar); % initial velocity
16 x = zeros(n, nvar); % initial positions
17 fitness = zeros(1, n); % initial fitness
18
19 for i=1:n
20

```

```
21     x(i,:) = LB + (UB-LB).*rand(1,nvar);
22     fitness(i) = fitness_function(faults,x(i,:),ctr,a,b,cti
        ,w1,w2);
23
24 end
25
26 [fmin,index] = min(fitness);
27 best = x(index,:);
28
29 Q = Qmin+(Qmax-Qmin)*rand(1,n);           % initial frequency
30
31 conv = zeros(1,niter);
32 conv_md = zeros(1,niter);
33
34 progressBar(0)
35
36 c_div = zeros(1,niter);
37 c_erro = zeros(1,niter);
38
39 for t=1:niter
40
41     conv(t) = fmin;
42     conv_md(t) = mean(fitness);
43
44     progressBar(t/niter)
45
46     % Etapa 1
47
48     for i=1:n
49
50         % global search
51
```

```
52     Q(i) = Qmin+(Qmax-Qmin)*rand(); %
        updates frequency
53     v(i,:) = (best - x(i,:))*Q(i); %
        updates velocity
54     xnew = x(i,:) + v(i,:);
55
56     % local search
57
58     if rand > r(i)
59
60         xnew = x(i,:) + 0.1*mean(A)*(-1+2*rand(1,nvar)
61             );
62     end
63
64     xnew = simplebounds(xnew, LB, UB);
65
66     fnew = fitness_function(faults, xnew, ctr, a, b, cti, w1,
67         w2);
68
69     if fnew <= fitness(i) && rand < A(i)
70
71         x(i,:) = xnew;
72         fitness(i) = fnew;
73
74         A(i) = alfa*A(i);
75         r(i) = ro*(1-exp(-gamma*t));
76     end
77
78     if fnew <= fmin
79
```

```

80         best = xnew;
81         fmin = fnew;
82
83     end
84
85 end
86
87     c_div(t) = diversity(x,best);
88     c_erro(t) = erro(fitness,fmin);
89
90 end
91
92 counters.div = c_div;
93 counters.error = c_erro;
94
95 progressBar(1)
96
97 end

```

B.5 Algoritmo do Morcego Direcional

```

1 function [best,fmin,conv,conv_md,counters] = dba_oc(n,nvar,
      A0,Af,r0,rf,Qmin,Qmax,niter,LB,UB,w1,w2,ff_parameters)
2
3 % Fitness Function Parameters
4
5 a = ff_parameters.a;
6 b = ff_parameters.b;
7 faults = ff_parameters.faults;
8 cti = ff_parameters.cti;
9 ctr = ff_parameters.ctr;
10

```



```
11
12 r = r0*ones(n,1);
13 A = A0*ones(n,1);
14
15 v = zeros(n,nvar);           % initial velocity
16 x = zeros(n,nvar);         % initial positions
17 fitness = zeros(1,n);      % initial fitness
18
19 w = zeros(n,nvar);
20 w0 = (UB-LB)/12;
21 wf = w0/100;
22
23 for i=1:n
24
25     x(i,:) = LB + (UB-LB).*rand(1,nvar);
26     fitness(i) = fitness_function(faults,x(i,:),ctr,a,b,cti
27         ,w1,w2);
28 end
29
30 [fmin,index] = min(fitness);
31 best = x(index,:);
32
33 conv = zeros(1,niter);
34 conv_md = zeros(1,niter);
35
36 progressBar(0)
37
38 c_div = zeros(1,niter);
39 c_erro = zeros(1,niter);
40
41 for t=1:niter
```

```
42
43     conv(t) = fmin;
44     conv_md(t) = mean(fitness);
45
46     progressbar(t/niter)
47
48     % Etapa 1
49
50     for i=1:n
51
52         % global search
53
54         i_randx = i;
55
56         while i_randx == i
57
58             i_randx = randi(n,1);
59
60         end
61
62         randx = x(i_randx,:);
63
64         q1 = Qmin + (Qmax - Qmin).*rand(1,nvar);
65         q2 = Qmin + (Qmax - Qmin).*rand(1,nvar);
66
67         if fitness(i_randx) < fitness(i)
68
69             v(i,:) = (best - x(i,:)).*q1 + (randx - x(i,:))
70                 .*q2;
71
72         else
```

```
73         v(i,:) = (best - x(i,:)).*q1;
74
75     end
76
77     xnew = x(i,:) + v(i,:);
78
79     % local search
80
81     if rand > r(i)
82
83         w(i,:) = (w0 - wf)*((t-niter)/(1-niter)) + wf;
84
85         v(i,:) = w(i,:).*(mean(A)*(-1+2*rand(1,nvar)));
86
87         xnew = best + v(i,:);
88
89     end
90
91     xnew = simplebounds(xnew, LB, UB);
92
93     fnew = fitness_function(faults, xnew, ctr, a, b, cti, w1,
94                             w2);
95
96     if fnew <= fitness(i) && rand < A(i)
97
98         x(i,:) = xnew;
99         fitness(i) = fnew;
100
101         A(i) = (A0 - Af)*((t-niter)/(1-niter)) + Af;
102         r(i) = (r0 - rf)*((t-niter)/(1-niter)) + rf;
103     end
```

```

104
105     if fnew <= fmin
106
107         best = xnew;
108         fmin = fnew;
109
110     end
111
112 end
113
114     c_div(t) = diversity(x,best);
115     c_erro(t) = erro(fitness,fmin);
116
117 end
118
119 counters.div = c_div;
120 counters.error = c_erro;
121
122 progressbar(1)
123
124 end

```

B.6 Algoritmo do Morcego Direcional *Fuzzy* Adaptativo

```

1 function [best,fmin,conv,conv_md,counters] = fdba2_oc(n,
2     nvar,niter,LB,UB,w1,w2,ff_parameters)
3 % Atualiza os parametros A, r, Q e w_f individualmente para
4     cada individuo
5 % Fitness Function Parameters
6

```

```
7 a = ff_parameters.a;
8 b = ff_parameters.b;
9 faults = ff_parameters.faults;
10 cti = ff_parameters.cti;
11 ctr = ff_parameters.ctr;
12
13 v = zeros(n,nvar); % initial velocity
14 x = zeros(n,nvar); % initial positions
15 fitness = zeros(1,n); % initial fitness
16
17 for i=1:n
18
19     x(i,:) = LB + (UB-LB).*rand(1,nvar);
20     fitness(i) = fitness_function(faults,x(i,:),ctr,a,b,cti
21         ,w1,w2);
22
23
24 [fmin,index] = min(fitness);
25 best = x(index,:);
26
27 conv = zeros(1,niter);
28 conv_md = zeros(1,niter);
29
30 progressBar(0)
31
32 c_div = zeros(1,niter);
33 c_erro = zeros(1,niter);
34 c_r = zeros(1,niter);
35 c_A = zeros(1,niter);
36
37 for t=1:niter
```

```
38
39     conv(t) = fmin;
40     conv_md(t) = mean(fitness);
41
42     progressbar(t/niter)
43
44     div_fis = diversity3(x,best);
45     %     it_fis = ones(n,1)*(t/niter);
46
47     %     param = evalfis([it_fis,div_fis],fis);
48
49     param = zeros(length(div_fis(:,1)),4);
50
51     for i=1:length(div_fis(:,1))
52
53         param(i,:) = [pulseRate(t/niter,div_fis(i)),
54                     loudness(t/niter,div_fis(i)), wf(t/niter,div_fis
55                     (i)), frequency(t/niter,div_fis(i))];
56
57     end
58
59     % Etapa 1
60
61     for i=1:n
62
63         r = param(i,1); A = param(i,2); w_f = param(i,3); Q
64         = param(i,4);
65
66         % global search
67
68         i_randx = i;
```

```
67     while i_randx == i
68
69         i_randx = randi(n,1);
70
71     end
72
73     randx = x(i_randx,:);
74
75     q1 = Q*rand(1,nvar);
76     q2 = Q*rand(1,nvar);
77
78     if fitness(i_randx) < fitness(i)
79
80         v(i,:) = (best - x(i,:)).*q1 + (randx - x(i,:))
81             .*q2;
82
83     else
84
85         v(i,:) = (best - x(i,:)).*q1;
86
87     end
88
89     xnew = x(i,:) + v(i,:);
90
91     % local search
92
93     if rand > r
94
95         w = (UB-LB)/w_f;
96
97         v(i,:) = w.*(mean(A)*(-1+2*rand(1,nvar)));
```

```
98         xnew = best + v(i,:);
99
100     end
101
102     xnew = simplebounds(xnew, LB, UB);
103
104     fnew = fitness_function(faults, xnew, ctr, a, b, cti, w1,
105                             w2);
106
107     if fnew <= fitness(i) && rand < A
108
109         x(i,:) = xnew;
110         fitness(i) = fnew;
111
112     end
113
114     if fnew <= fmin
115
116         best = xnew;
117         fmin = fnew;
118
119     end
120 end
121
122 c_div(t) = diversity(x, best);
123 c_erro(t) = erro(fitness, fmin);
124 c_r(t) = r;
125 c_A(t) = A;
126
127 end
128
```



```
129 counters.div = c_div;
130 counters.error = c_erro;
131 counters.r = c_r;
132 counters.A = c_A;
133
134 progressbar(1)
135
136 end
```

ANEXO C – TESTE ANOVA DOS PARÂMETROS DOS ALGORITMOS

Neste anexo são apresentadas os resultados do teste ANOVA aplicado ao delineamento fatorial dos parâmetros dos algoritmos, para determinar o nível de significância de cada parâmetro, e assim determinar o valor a ser adotado, na minimização da função objetivo.

C.1 Valores adotados para cada parâmetro

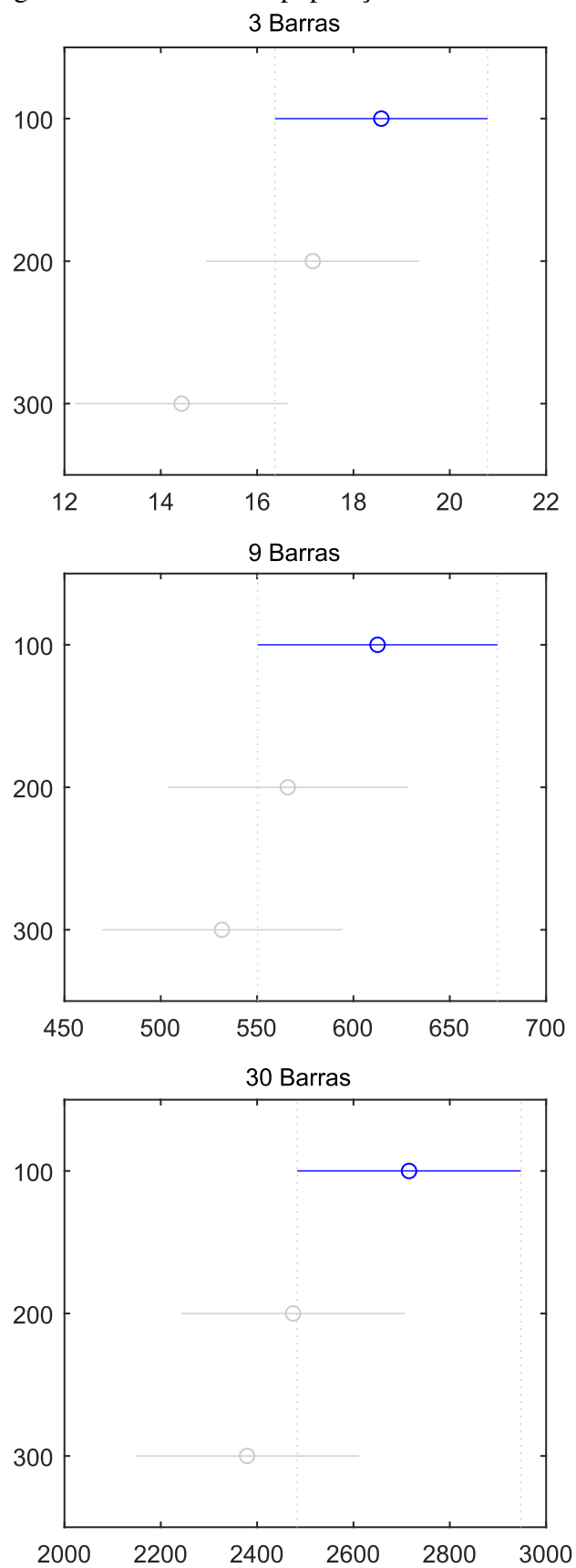
Os parâmetros a serem testados e os valores considerados para cada parâmetro no delineamento fatorial realizado foram os seguintes:

- AG: Tamanho da população = 100; 200; 300; Taxa de mutação = 0,01; 0,05; 0,01, Taxa de cruzamento = 1, Critério de seleção = 0,5; 0,8; 1,0;
- OEP: Tamanho da população = 100; 200; 300; Fator de inercia inicial = 0,9; 0,7; 0,6; Fator de inercia final = 0,1; 0,3; 0,5; $c_1 = 1; 1,5; 2,5$; Obs.: c_2 foi adotado como $c_2 = 4 - c_1$
- ED: Tamanho da população = 100; 200; 300; $F = 0,7; 0,8; 0,9$, Taxa de cruzamento = 0,5; 0,75; 0,9;
- AM: Tamanho da população = 150; 300; $A_0 = 1,0$, $\alpha = 0,992; 0,996$, $r_0 = 1,0$, $\gamma = 0,003; 0,006$, $f_{min} = 0; 0,5$ e $f_{max} = 1; 2$
- AMD: Tamanho da população = 150; 300; $A_0 = 0,6; 0,9$, $A_\infty = 0,3; 0,5$, $r_0 = 0,3; 0,5$, $r_\infty = 0,6; 0,9$, $f_{min} = 0; 0,5$ e $f_{max} = 1; 2$
- AMDFA: Tamanho da população = 100; 200; 300.

Todos os algoritmos foram testados com três níveis, com exceção do AM e do AMD, devido a grande quantidade de fatores a serem analisados. Para cada experimento, foram coletadas 25 amostras, com exceção também dos testes com AM e AMD, em que foram coletadas 10 amostras, devido ao grande esforço computacional que 25 amostras exigiriam.

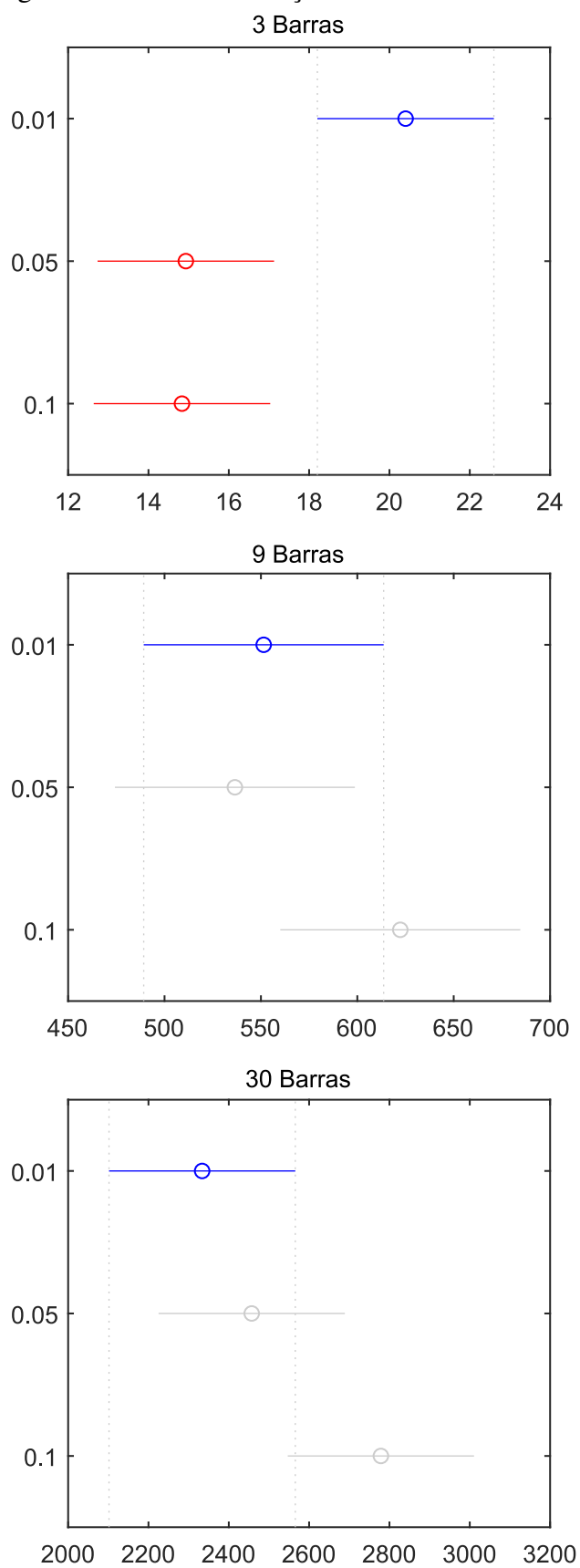
Os gráficos de significância obtidos para cada caso teste (3 barras, 9 barras e 30 barras) de cada algoritmo são apresentados a seguir.

Figura 45 – Tamanho da população - AG.



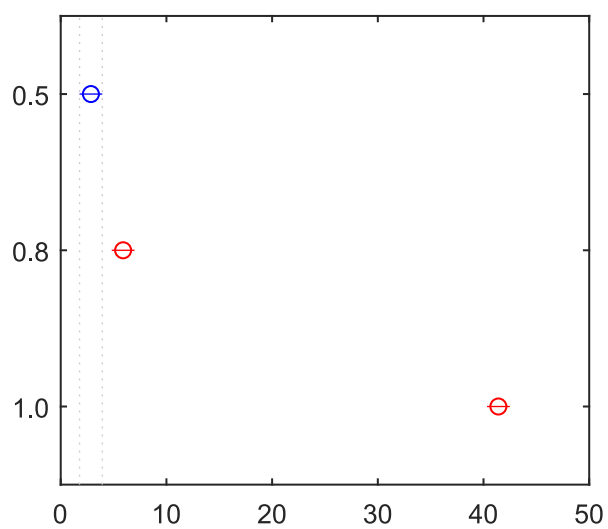
Fonte: O próprio autor.

Figura 46 – Taxa de mutação - AG.

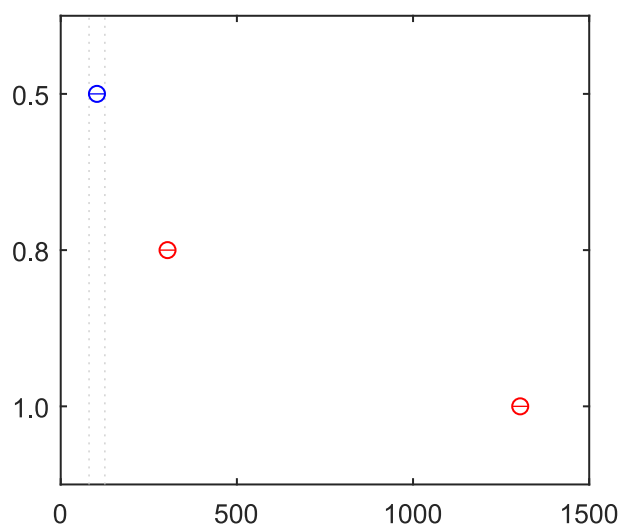


Fonte: O próprio autor.

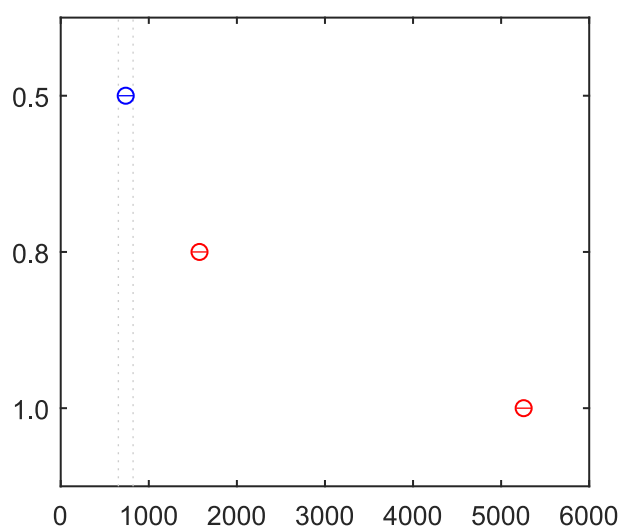
Figura 47 – Critério de seleção - AG.
3 Barras



9 Barras

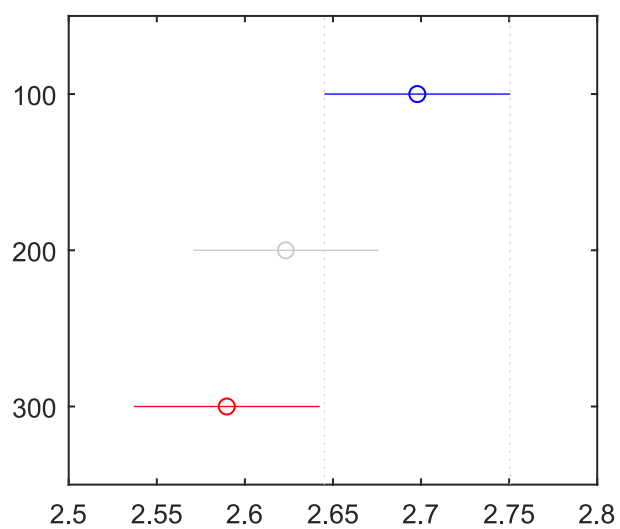


30 Barras

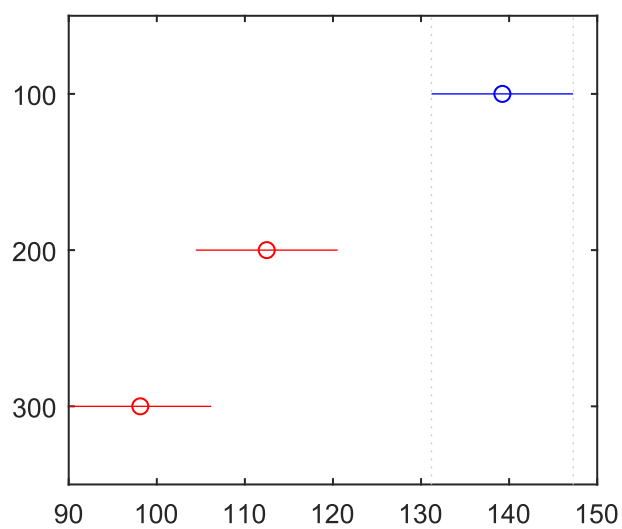


Fonte: O próprio autor.

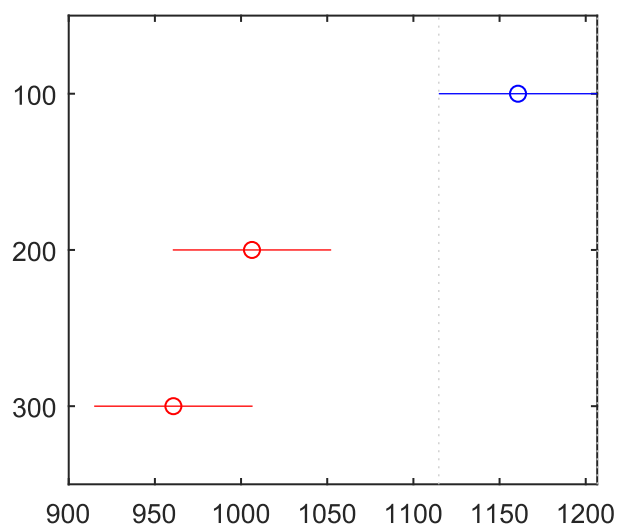
Figura 48 – Tamanho da população - EOP.
3 Barras



9 Barras

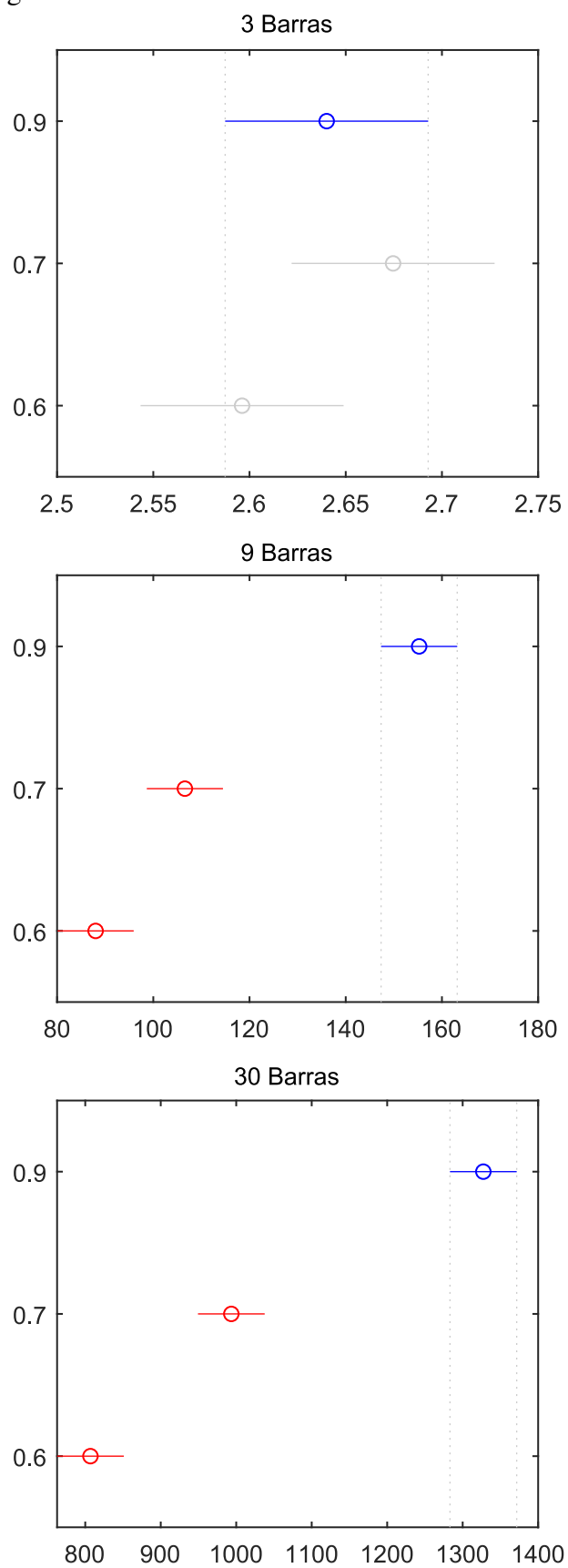


30 Barras



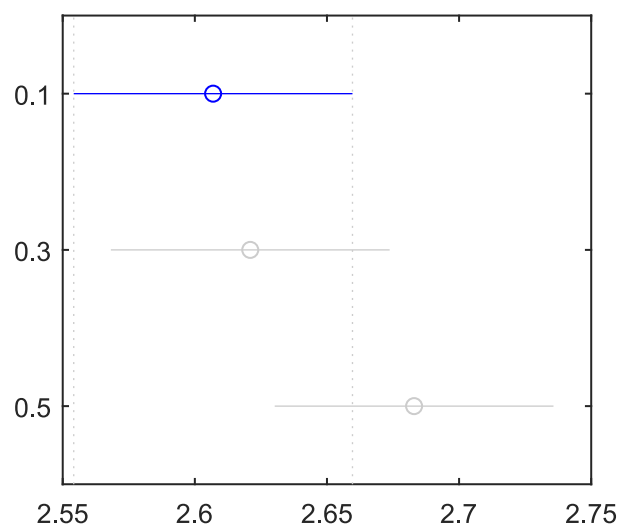
Fonte: O próprio autor.

Figura 49 – Fator de inércia inicial - EOP.

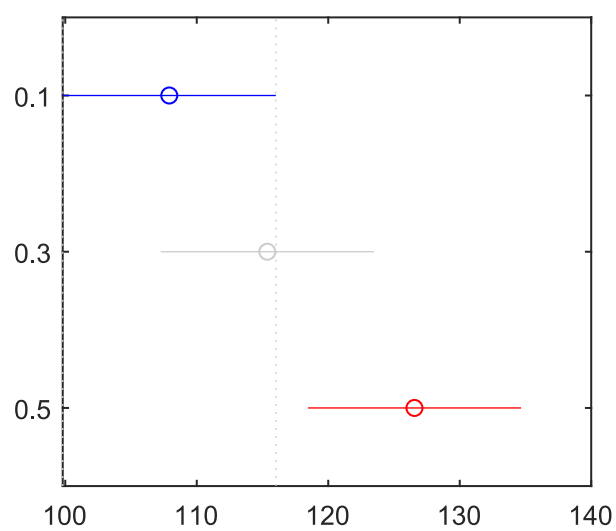


Fonte: O próprio autor.

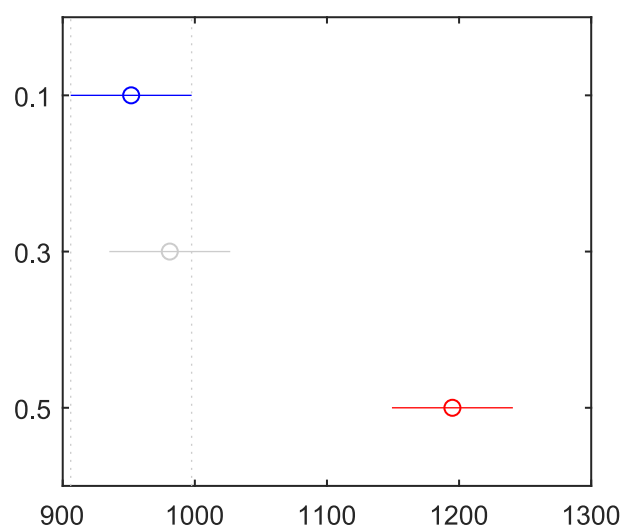
Figura 50 – Fator de inércia final - EOP.
3 Barras



9 Barras

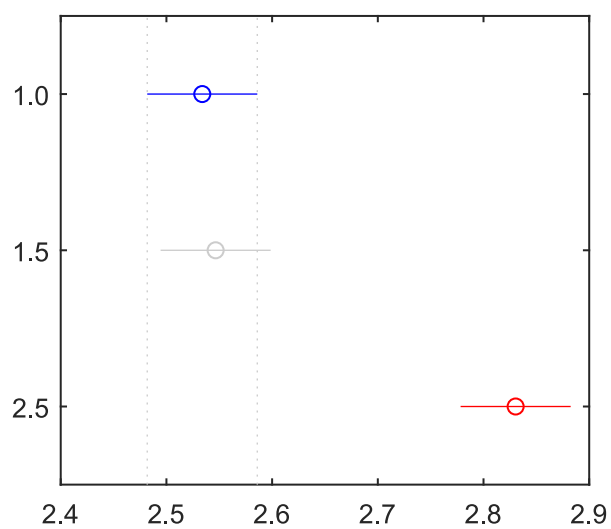


30 Barras

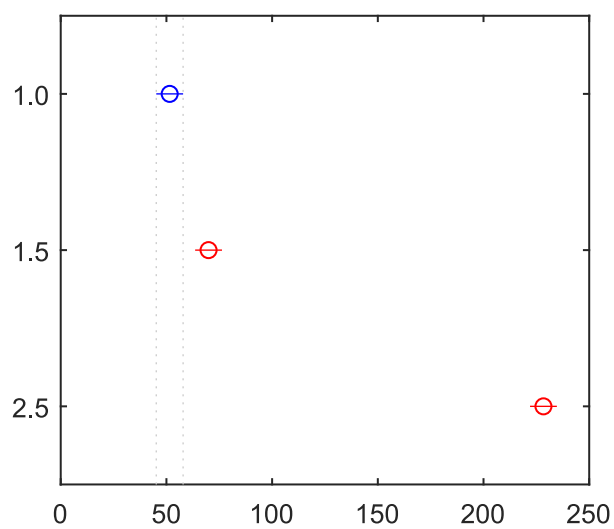


Fonte: O próprio autor.

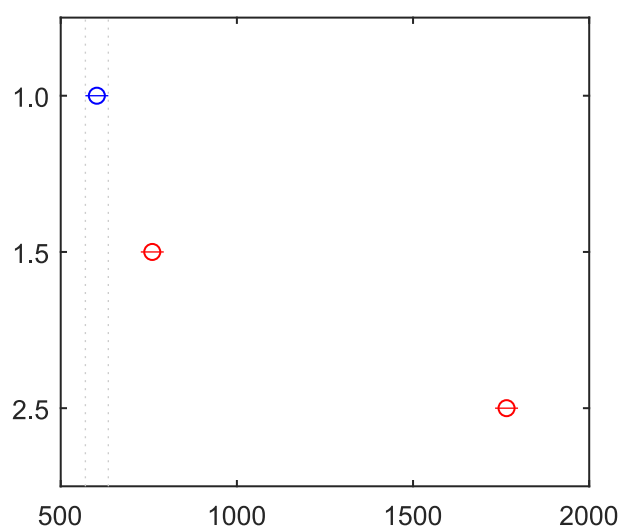
Figura 51 – Constante c_1 - EOP.
3 Barras



9 Barras

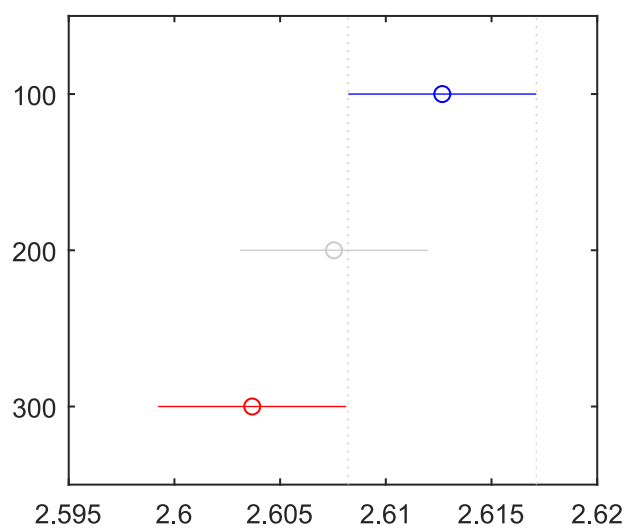


30 Barras

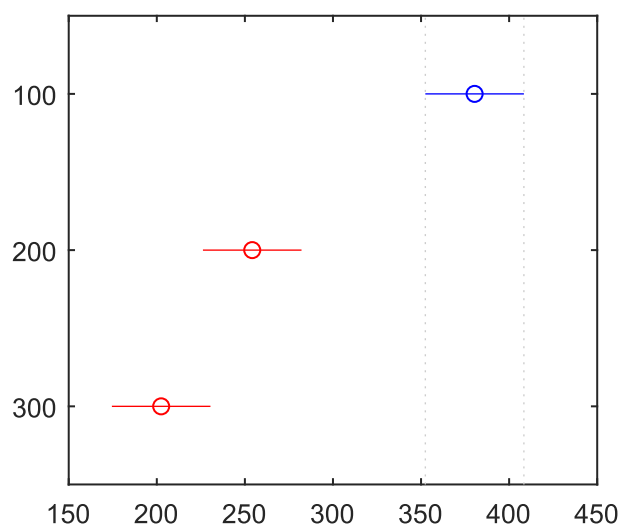


Fonte: O próprio autor.

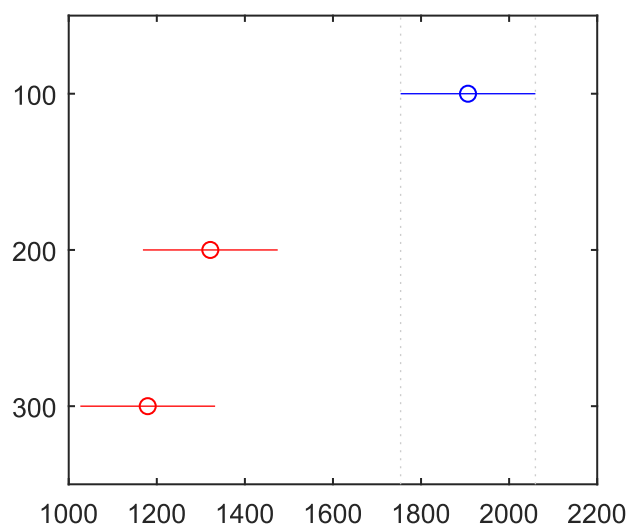
Figura 52 – Tamanho da população - ED.
3 Barras



9 Barras

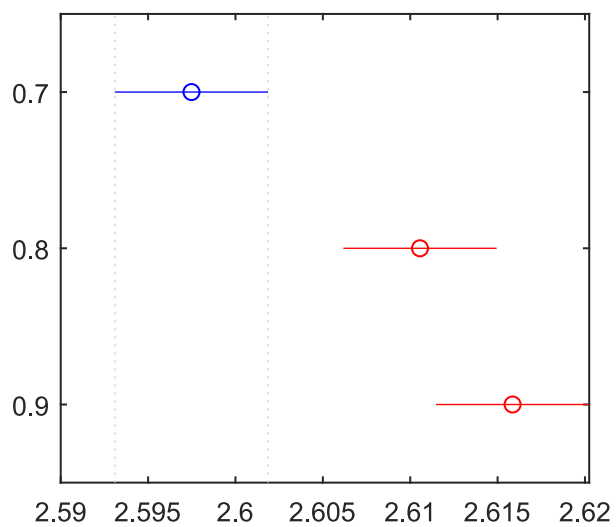


30 Barras

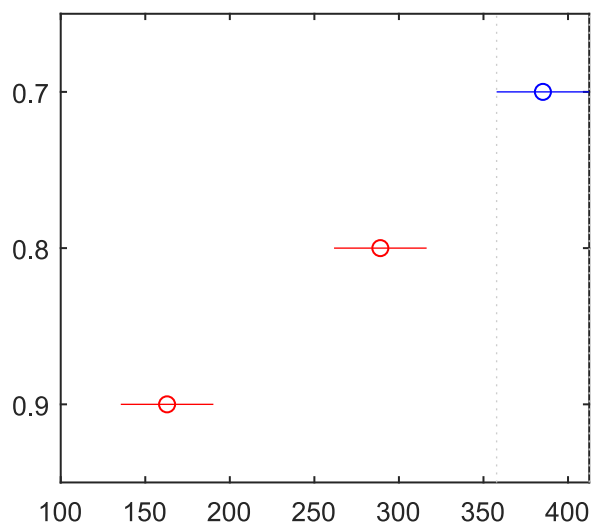


Fonte: O próprio autor.

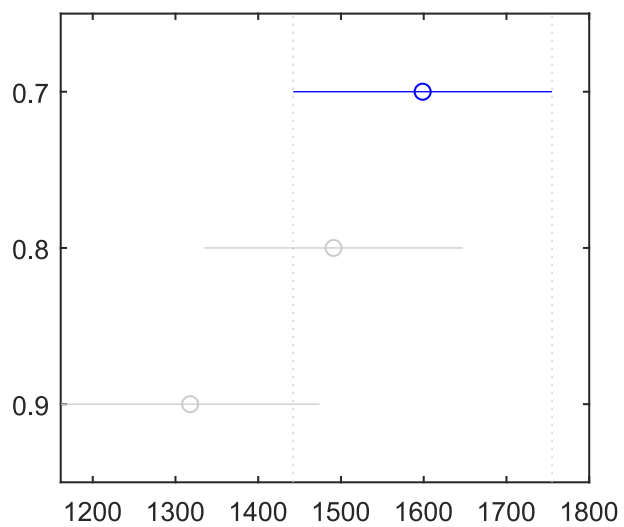
Figura 53 – Constante F - ED.
3 Barras



9 Barras

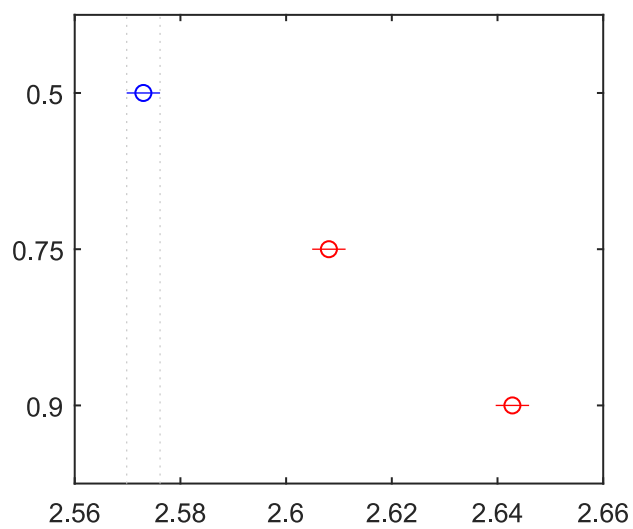


30 Barras

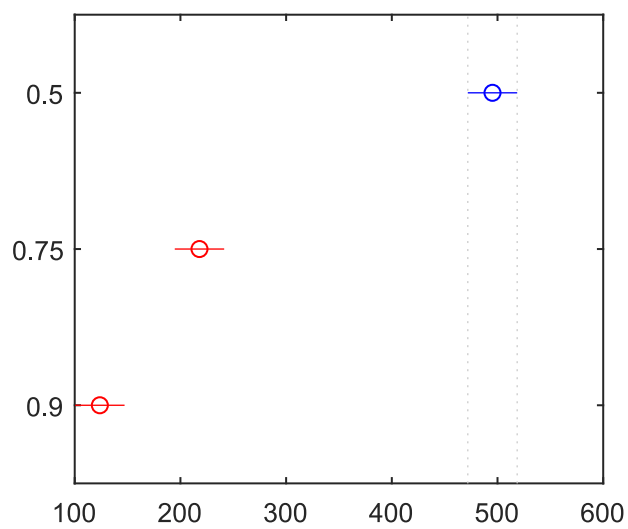


Fonte: O próprio autor.

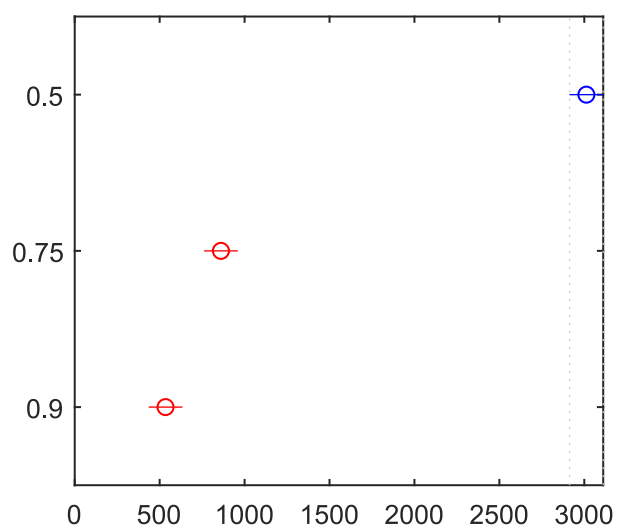
Figura 54 – Taxa de cruzamento - ED.
3 Barras



9 Barras

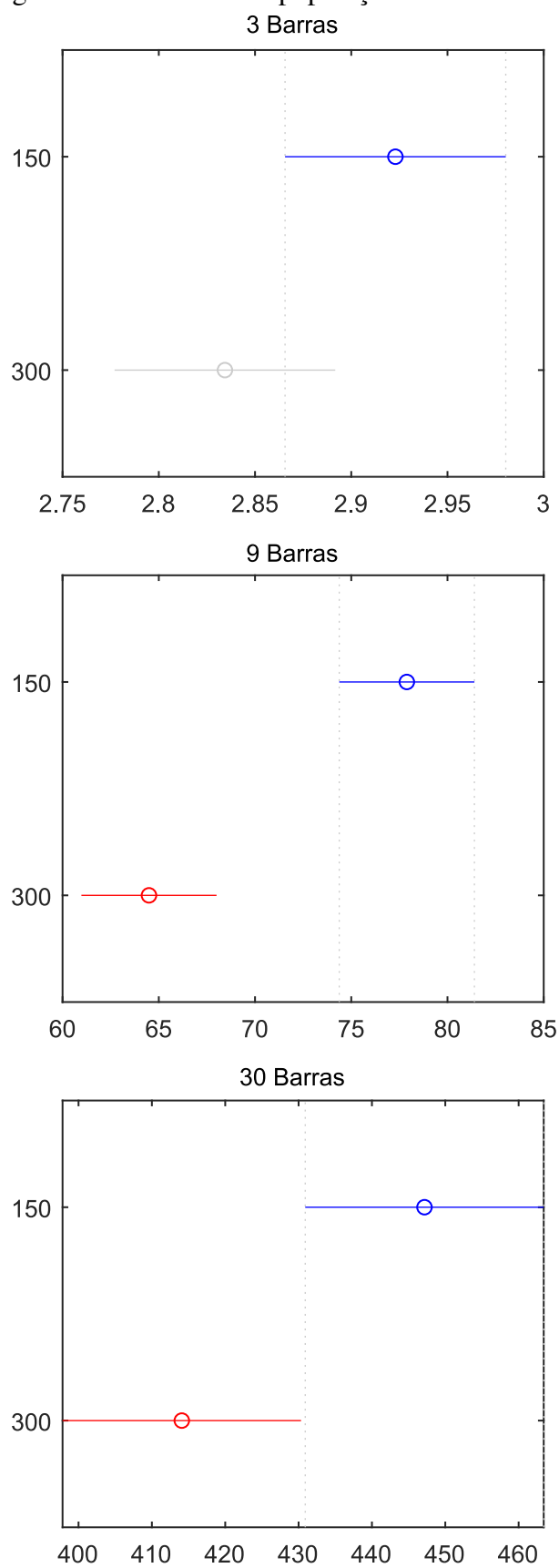


30 Barras



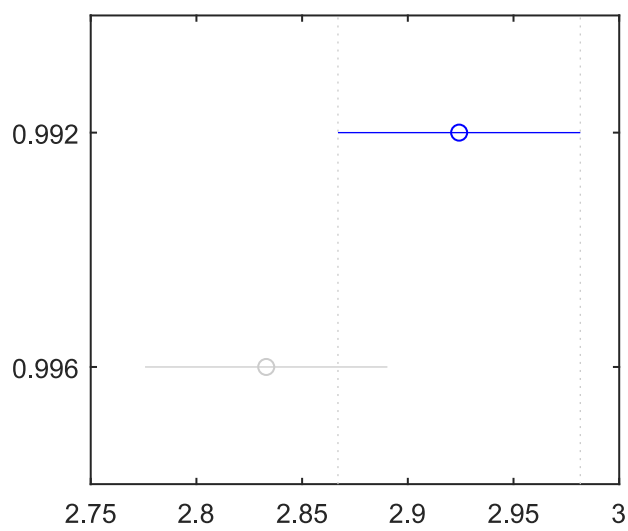
Fonte: O próprio autor.

Figura 55 – Tamanho da população - AM.

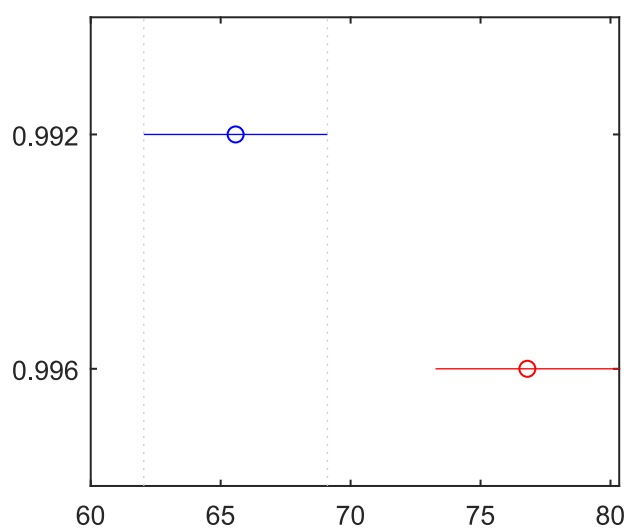


Fonte: O próprio autor.

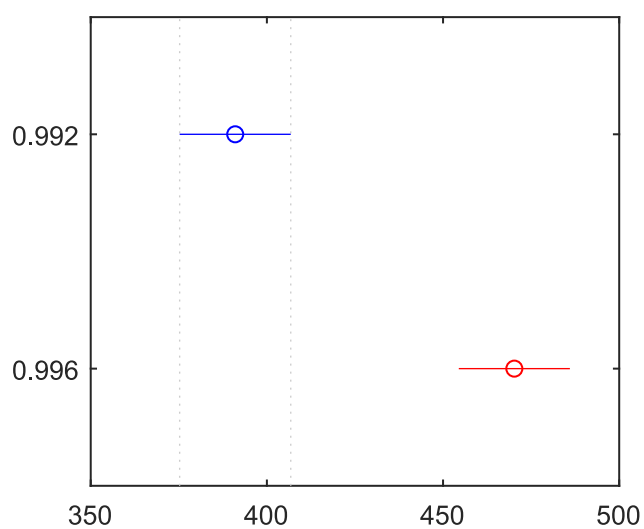
Figura 56 – Constante α - AM.
3 Barras



9 Barras

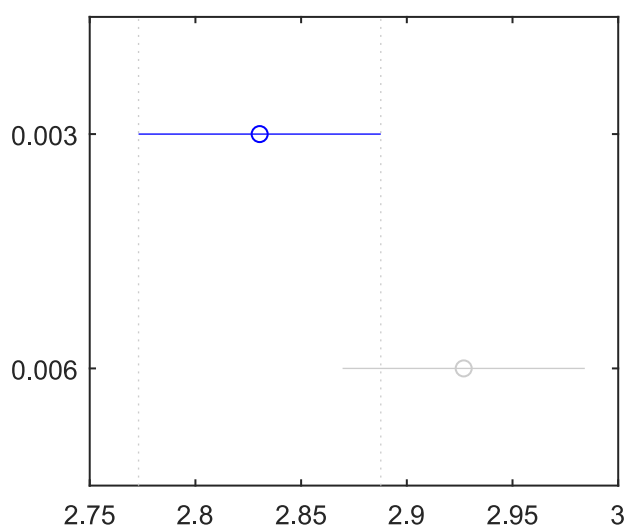


30 Barras

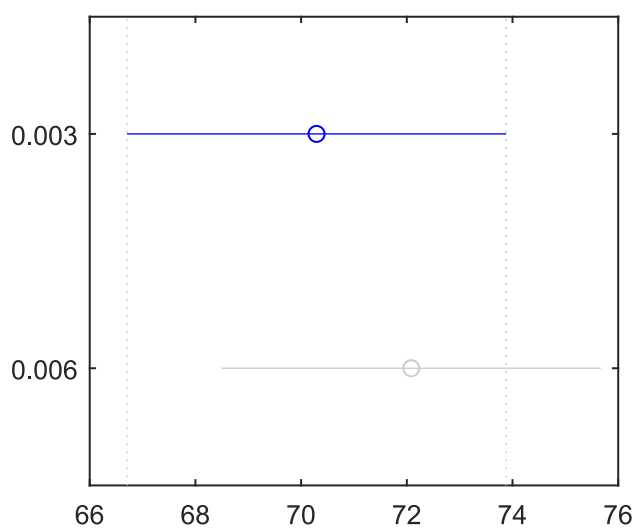


Fonte: O próprio autor.

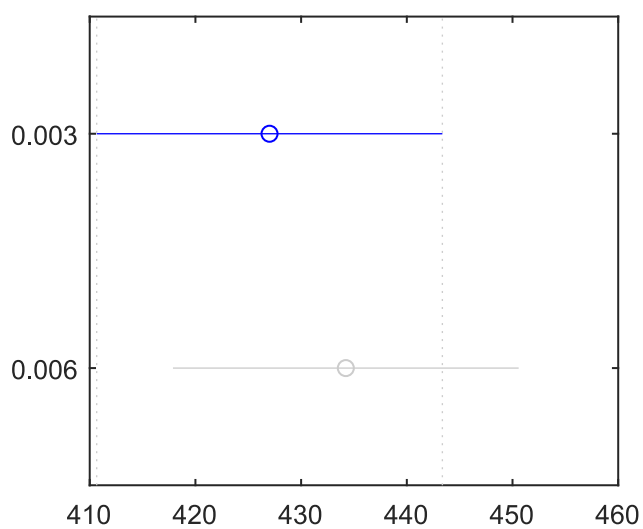
Figura 57 – Constante γ - AM.
3 Barras



9 Barras

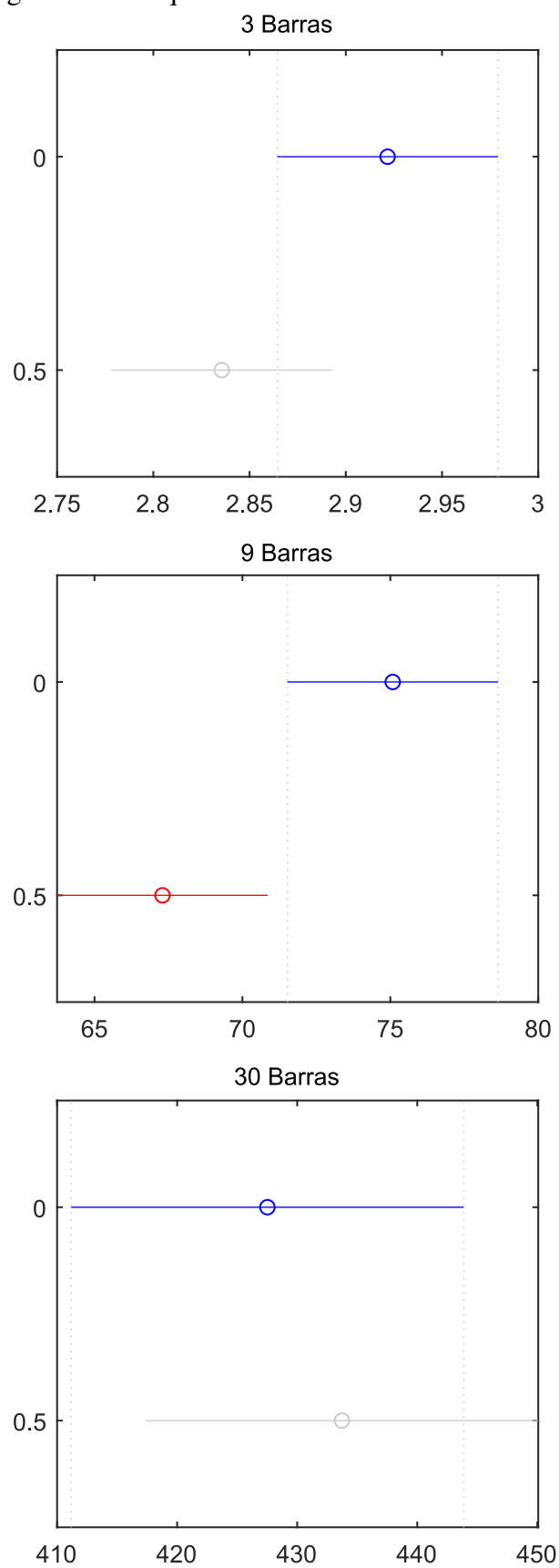


30 Barras



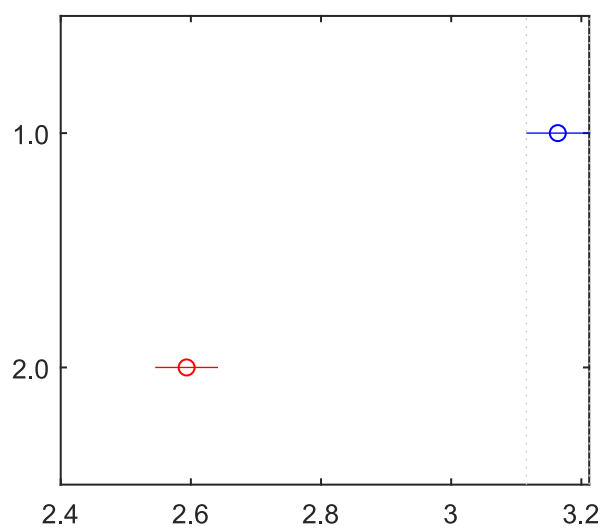
Fonte: O próprio autor.

Figura 58 – Frequência mínima - AM.

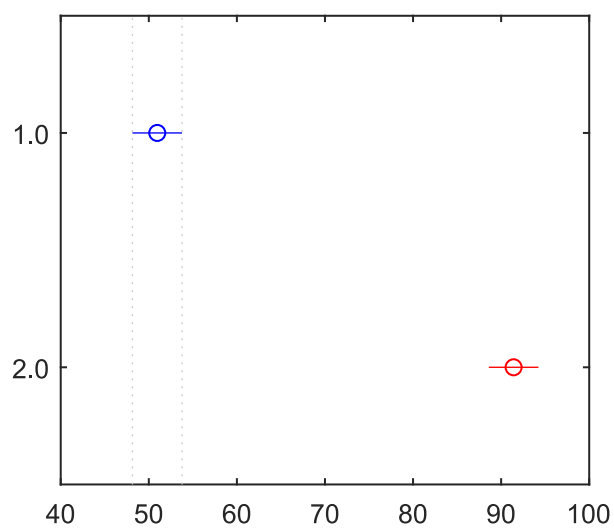


Fonte: O próprio autor.

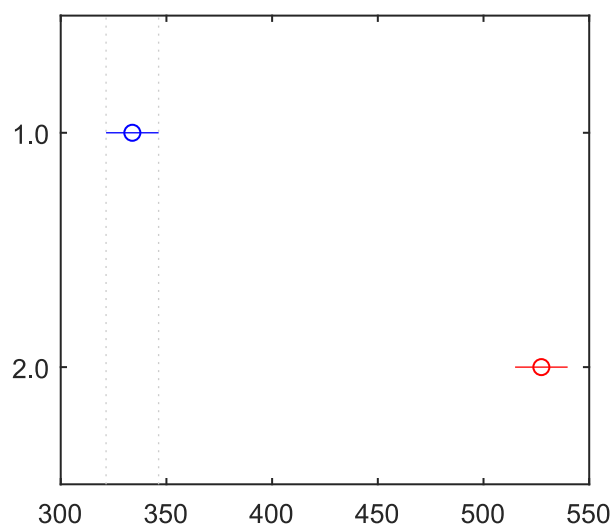
Figura 59 – Frequência máxima - AM.
3 Barras



9 Barras

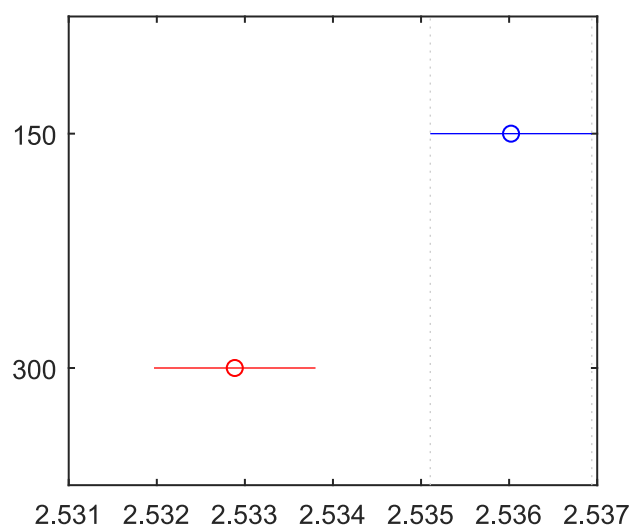


30 Barras

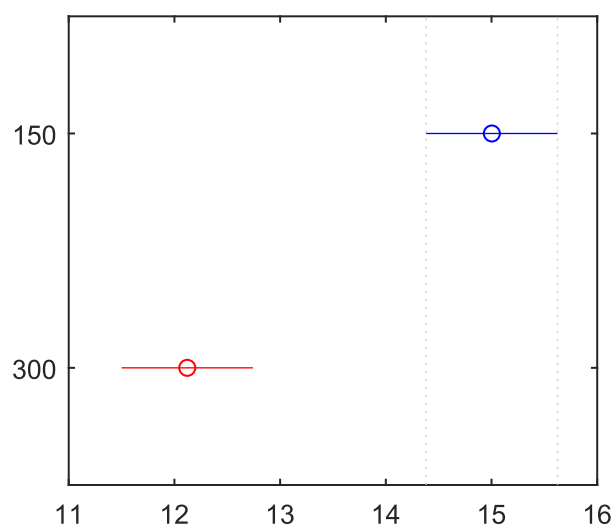


Fonte: O próprio autor.

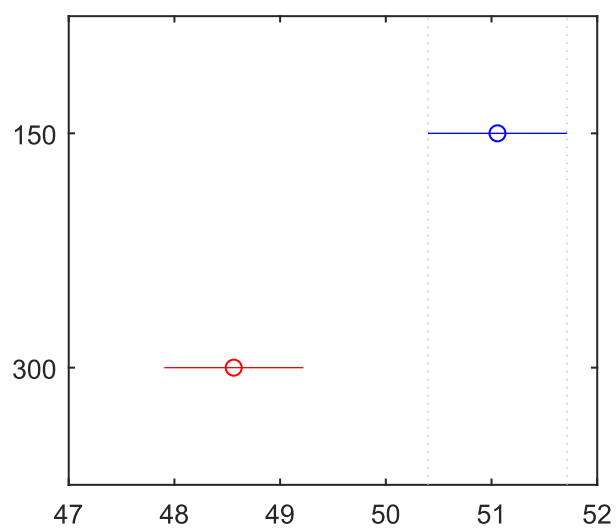
Figura 60 – Tamanho da população - AMD.
3 Barras



9 Barras

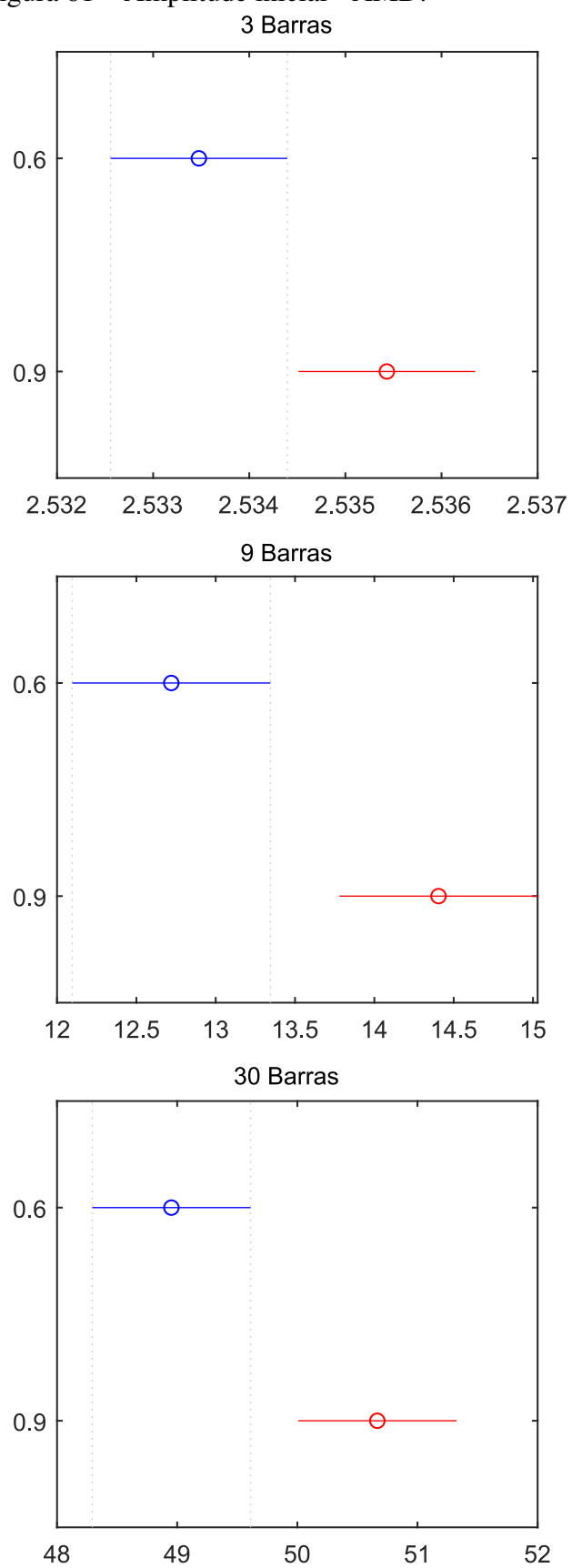


30 Barras



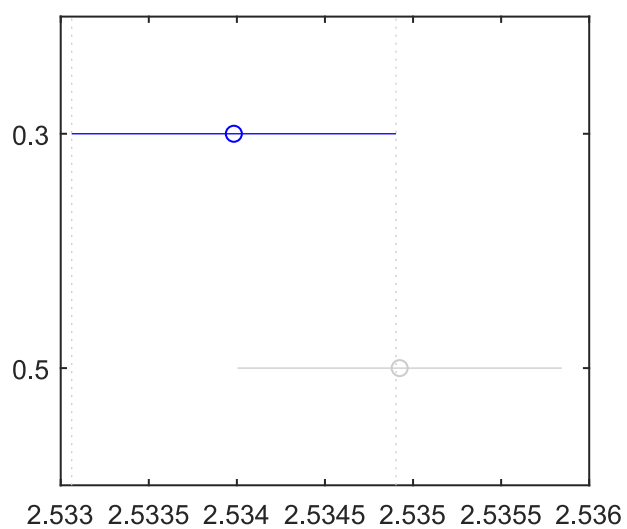
Fonte: O próprio autor.

Figura 61 – Amplitude inicial - AMD.

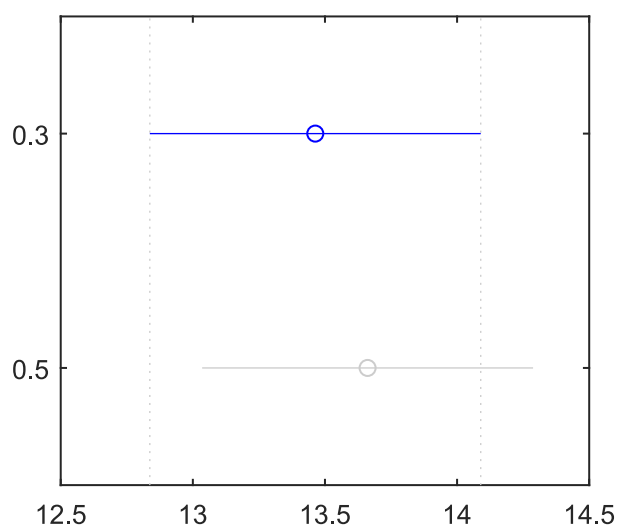


Fonte: O próprio autor.

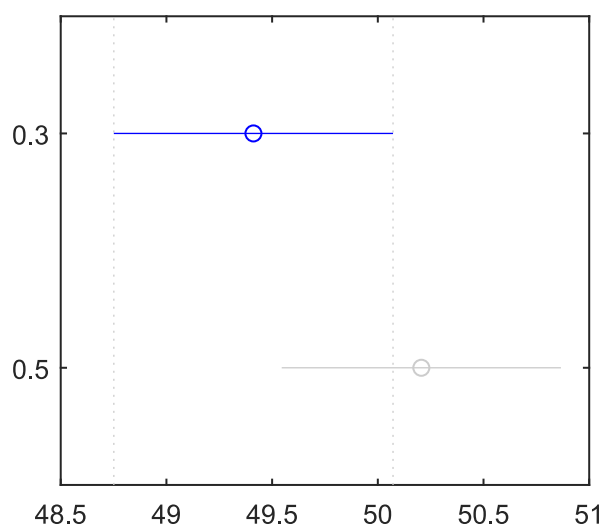
Figura 62 – Amplitude final - AMD.
3 Barras



9 Barras

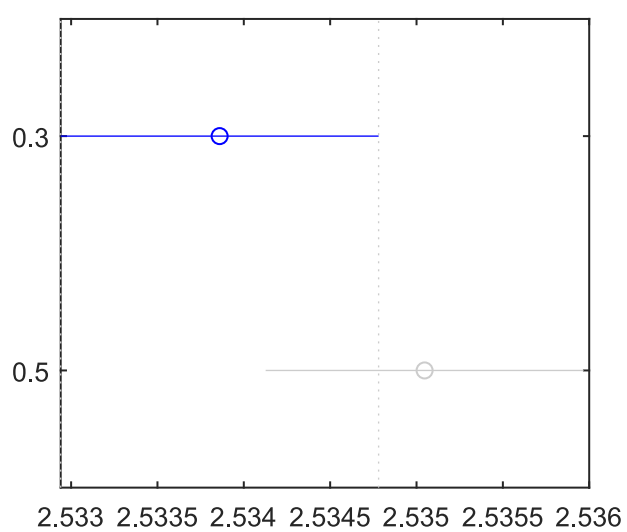


30 Barras

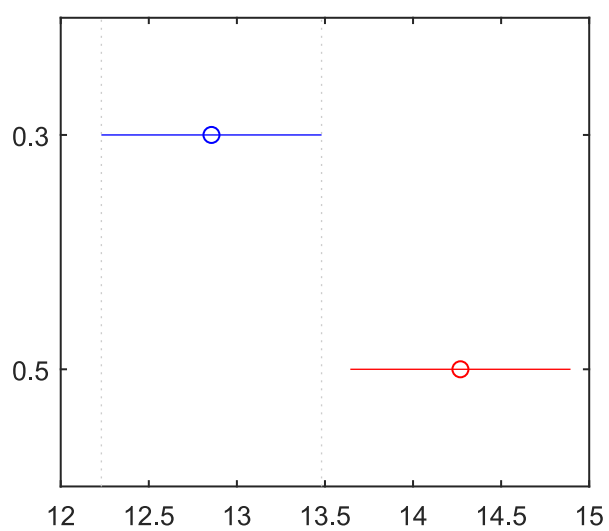


Fonte: O próprio autor.

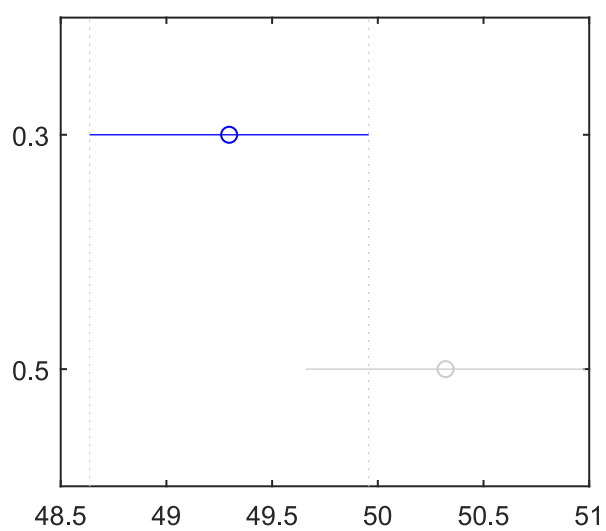
Figura 63 – Frequência de emissão inicial - AMD.
3 Barras



9 Barras

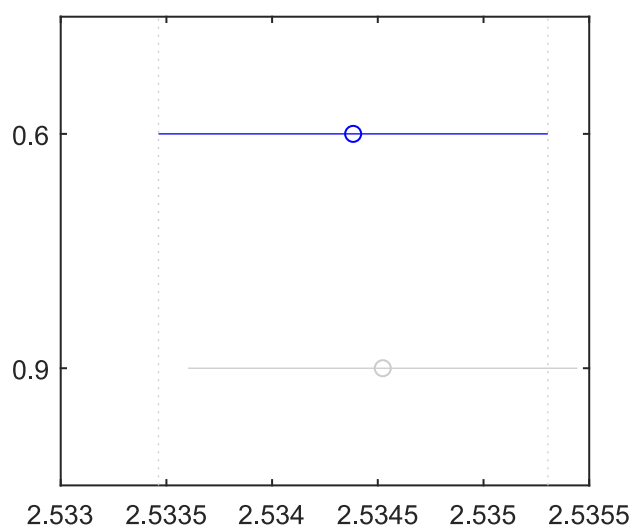


30 Barras

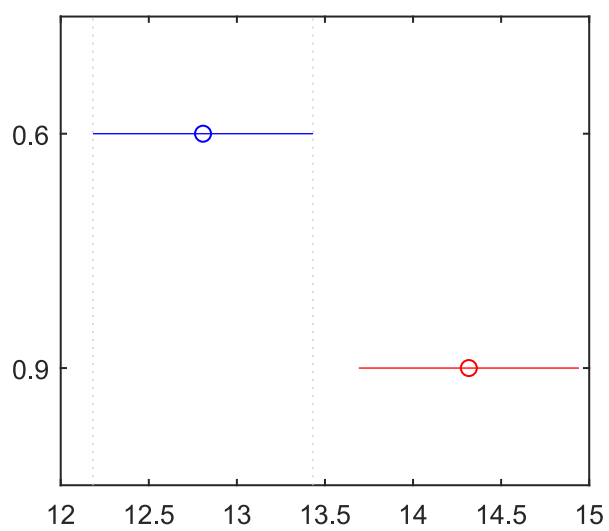


Fonte: O próprio autor.

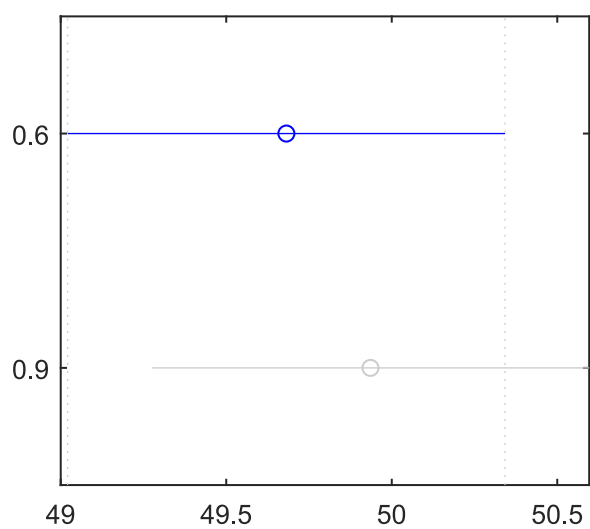
Figura 64 – Frequência de emissão final - AMD.
3 Barras



9 Barras

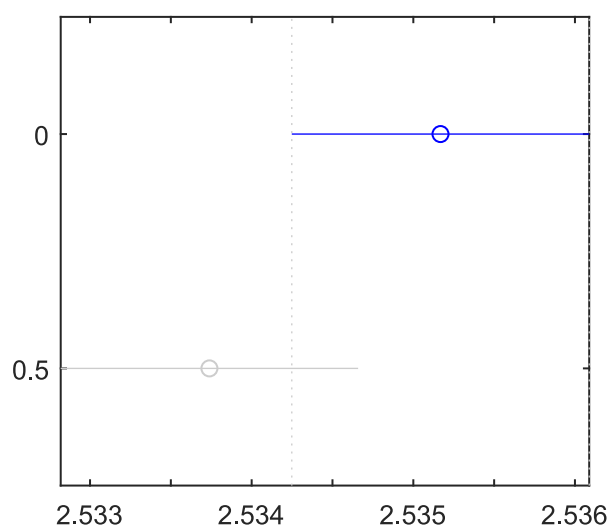


30 Barras

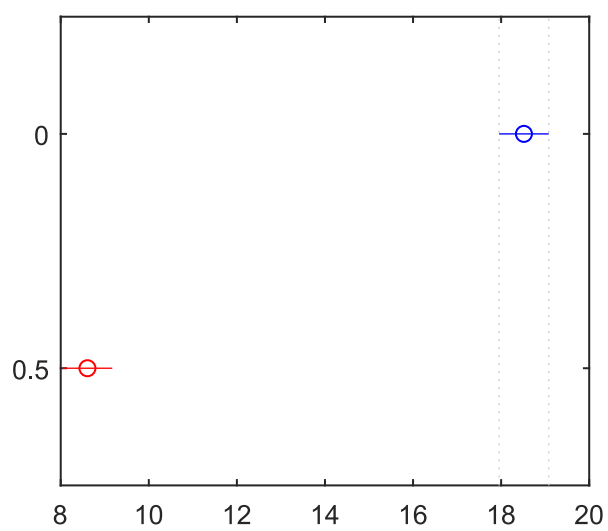


Fonte: O próprio autor.

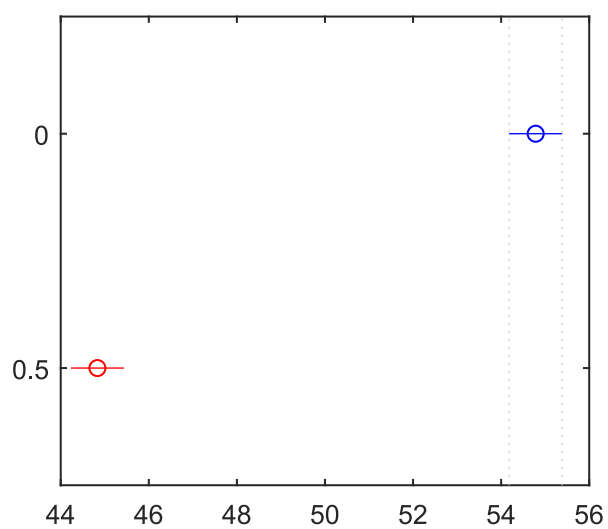
Figura 65 – Frequência mínima - AMD.
3 Barras



9 Barras

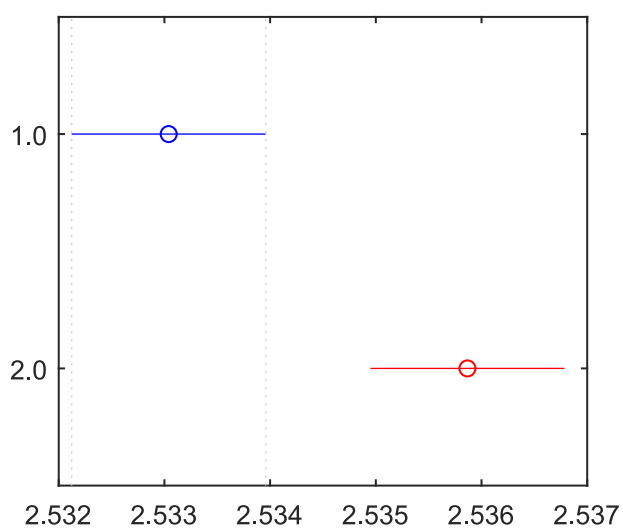


30 Barras

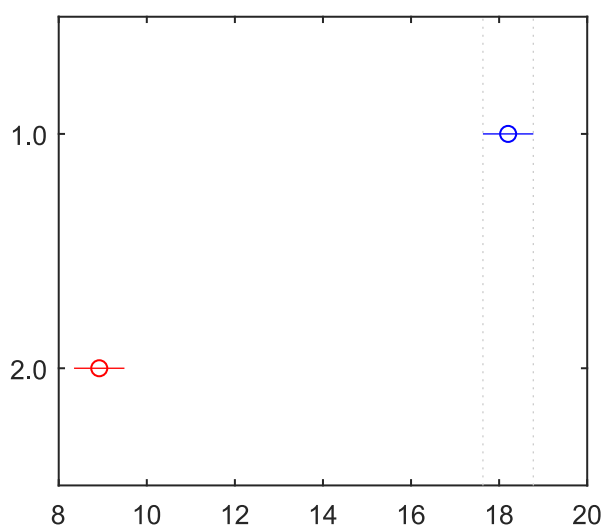


Fonte: O próprio autor.

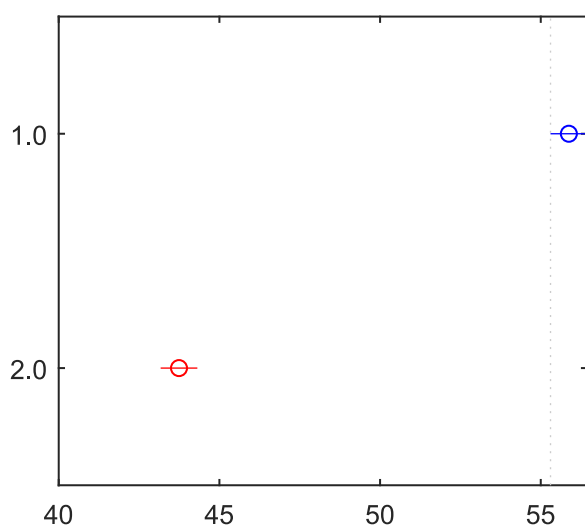
Figura 66 – Frequência máxima - AMD.
3 Barras



9 Barras

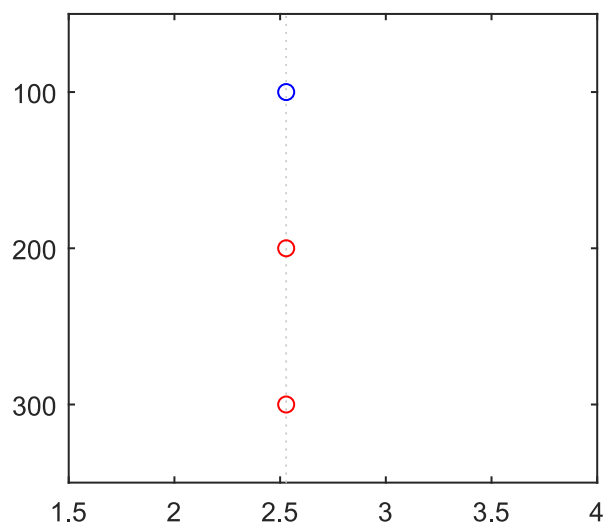


30 Barras

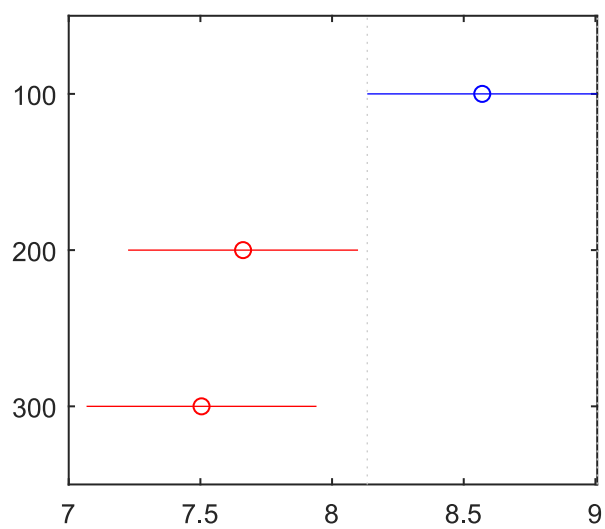


Fonte: O próprio autor.

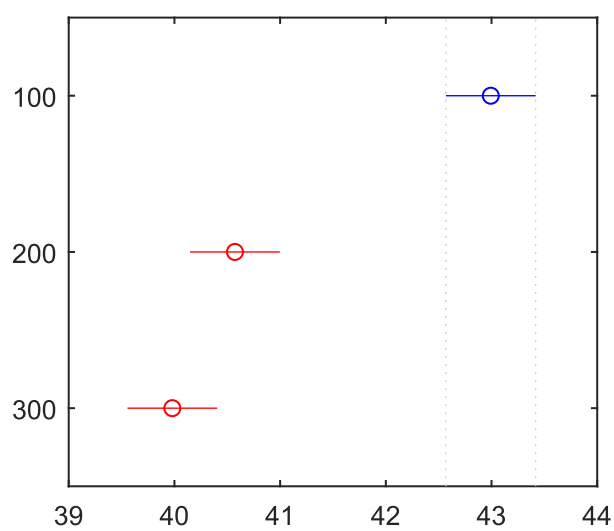
Figura 67 – Tamanho da população - AMDFA.
3 Barras



9 Barras



30 Barras



Fonte: O próprio autor.