

João Vitor Dias

**Evolução diferencial híbrida com o método
BFGS para a resolução do problema de
despacho econômico com o efeito de ponto
de carregamento de válvula**

Bauru - SP

Junho de 2021

João Vitor Dias

**Evolução diferencial híbrida com o método BFGS
para a resolução do problema de despacho econômico
com o efeito de ponto de carregamento de válvula**

Dissertação de Mestrado

Universidade Estadual Paulista "Júlio de Mesquita Filho"

Faculdade de Engenharia de Bauru

Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Prof^a. Dr^a. Edméa Cássia Baptista

Coorientador: Prof^a. Dr^a. Edilaine Martins Soler

Bauru - SP

Junho de 2021

Dias, João Vitor.

Evolução diferencial híbrida com o método BFGS para a resolução do problema de despacho econômico com o efeito de ponto de carregamento de válvula / João Vitor Dias, 2021

113 f. : il.

Orientadora: Edméa Cássia Baptista

Coorientadora: Edilaine Martins Soler

Dissertação (Mestrado)-Universidade Estadual Paulista. Faculdade de Engenharia, Bauru, 2021

1. Problema de despacho econômico com o efeito de ponto de carregamento de válvula. 2. Método evolução diferencial. 3. Método BFGS. 4. Otimização não linear. I. Universidade Estadual Paulista. Faculdade de Engenharia. II. Título.

ATA DA DEFESA PÚBLICA DA DISSERTAÇÃO DE MESTRADO DE JOÃO VITOR DIAS, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA, DA FACULDADE DE ENGENHARIA - CÂMPUS DE BAURU.

Aos 08 dias do mês de junho do ano de 2021, às 09:00 horas, por meio de Videoconferência, realizou-se a defesa de DISSERTAÇÃO DE MESTRADO de JOÃO VITOR DIAS, intitulada **EVOLUÇÃO DIFERENCIAL HÍBRIDA COM O MÉTODO BFGS PARA A RESOLUÇÃO DO PROBLEMA DE DESPACHO ECONÔMICO COM O EFEITO DE PONTO DE CARREGAMENTO DE VÁLVULA**. A Comissão Examinadora foi constituída pelos seguintes membros: Prof^a. Dr^a. EDILAINE MARTINS SOLER (Participação Virtual) do(a) Departamento de Matemática / Faculdade de Ciências de Bauru - UNESP, Prof. Dr. ANTONIO ROBERTO BALBO (Participação Virtual) do(a) Departamento de Matemática / Faculdade de Ciências de Bauru - UNESP, Prof. Dr. DIEGO NUNES DA SILVA (Participação Virtual) do(a) Departamento de Matemática / Instituto Federal de São Paulo - IFSP - Câmpus Presidente Epitácio. Após a exposição pelo mestrando e arguição pelos membros da Comissão Examinadora que participaram do ato, de forma presencial e/ou virtual, o discente recebeu o conceito final: APROVADO. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelo(a) Presidente(a) da Comissão Examinadora.



Prof^a. Dr^a. EDILAINE MARTINS SOLER

*À minha mãe que tanto lutou pela minha educação.
Seus braços que tanto trabalharam para que minhas mãos pudessem um dia escrever esse
trabalho.*

À minha luz e motivação, minha mãe, Marcia Andreia Gabriel.

AGRADECIMENTOS

Agradeço à minha irmã, Cassiane Lorena Dias, por todo apoio e pelo amor incondicional. Obrigado por sempre acreditar em mim.

Ao meu pai, Almir Malaquias Dias, pela minha educação e proteção.

À minha orientadora, professora Edméa Cássia Baptista, por ter me concedido a primeira oportunidade ainda na iniciação científica e, não só por todo conhecimento compartilhado, mas também por ser inspiração para minha vida pessoal e profissional.

À minha coorientadora, professora Edilaine Martins Soler, por toda dedicação e disposição em me ajudar sempre, além de ser uma grande inspiração pessoal e profissional.

Aos amigos e colegas da pós-graduação, em especial à minha "madrinha de pós", Letícia Maria Miquelin, por toda ajuda, risadas, companheirismo e por se tornar uma pessoa tão importante na minha vida; à Amanda Nerger pela grande amizade que construímos, companheirismo e ajuda; ao Tiago Gomes Cabana (Cabana) que muito me ajudou com programação; ao Mateus Gomes Cabana (Cabaninha) por toda ajuda, risadas e amizade; ao Gustavo Tanaka (Japa) pela amizade e companheirismo; ao Daniel Garcia (tio) e Ana Raquel Faccioli por estarem sempre comigo me ajudando nesta caminhada.

À minha amiga, Beatriz Ceschim, que foi peça fundamental nesta caminhada, sempre me dando muito carinho e discernimento. Agradeço por essa amizade e companheirismo que temos um para com o outro.

À minha eterna professora, Marisa Nogueira, por ter sido minha inspiração desde o ensino médio e me mostrado a beleza da matemática. Agradeço por toda sua ajuda e de seu esposo Marcos Roberto Mariano nos dias do vestibular. Serei eternamente grato por terem acreditado em mim.

À minha amiga, Juliana Ferreira, por sempre ter me motivado, acreditado e estado ao meu lado.

Aos amigos de fora da universidade, que sempre estiveram na torcida por mim.

À Faculdade de Engenharia de Bauru, com seus professores, técnicos e demais funcionários, que me permitiram concluir essa importante etapa.

À CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, pelo apoio financeiro concedido.

“Um dia, nalgum lugar, uma eternidade após, eu lembraria tudo isto num suspiro: Dois caminhos divergiam numa floresta de outono, e eu, eu escolhi o menos percorrido, e isto fez toda a diferença!”
(Robert Frost)

RESUMO

Problemas de despacho econômico (PDE) têm por finalidade determinar a geração de potência ativa de energia nas unidades geradoras, minimizando o custo de combustível ao mesmo tempo que as restrições do sistema são atendidas. Uma variante do PDE é o PDE com o efeito de ponto de carregamento de válvula (PDE-PCV). Este, considera o efeito de ponto de carregamento de válvula na função objetivo do problema e é formulado como um problema de otimização restrito, não convexo e não diferenciável. Isto dificulta a aplicação de métodos determinísticos de otimização para sua resolução. Por esta razão, a maior parte dos métodos encontrados na literatura para a resolução do PDE-PCV são baseados em heurísticas. Neste trabalho, propõe-se utilizar um método heurístico híbrido que integra o método evolução diferencial (ED) com o método Quase-Newton desenvolvido por Broyden, Fletcher, Goldfarb e Shanno (BFGS) para a resolução do PDE-PCV. A modificação proposta visa melhorar o desempenho do método ED quanto a sua busca local utilizando o método BFGS. A ED possui estrutura simples, velocidade rápida de convergência, propriedade de busca global, enquanto o método BFGS pode explorar o espaço de busca rapidamente com uma direção de gradiente e não requer recalcular a matriz hessiana a cada iteração. Uma implementação computacional dessa abordagem foi desenvolvida no *software* Matlab, e testes mostraram a eficiência do método híbrido proposto na resolução do PDE-PCV, obtendo soluções de boa qualidade quando comparadas com o método ED clássico e alguns métodos encontrados na literatura.

Palavras-chave: Problema de Despacho Econômico com o Efeito de Ponto de Carregamento de Válvula, Método Evolução Diferencial, Método BFGS, Otimização Não Linear.

ABSTRACT

Economic dispatch problems (EDP) are designed to determine the generation of active power in the generating units, minimizing the fuel cost at the same time that the system restrictions are met. A variant of the EDP is the EDP with the valve point loading effect (EDP-VLP). This considers the effect of the valve point loading on the objective function of the problem and is formulated as a constrained, non-convex and non-differentiable optimization problem. This makes it impossible to apply deterministic optimization methods directly for their resolution. For this reason, most methods found in the literature to solve the EDP-VLP are based on heuristics. In this work, it is proposed a hybrid heuristic method that integrates the differential evolution (DE) method with the Quasi-Newton method developed by Broyden, Fletcher, Goldfarb and Shanno (BFGS) to solve the EDP-VLP. The proposed approach aims to improve the performance of the DE method in terms of its local search using the BFGS method. The DE has a simple structure, fast convergence speed, global search property, while the BFGS method can explore the search space quickly with a gradient direction and does not require recalculating the Hessian matrix at each iteration. A computational implementation of this approach was developed in Matlab software, and tests showed the efficiency of the proposed hybrid method in the resolution of the EDP-VLP, obtaining good quality solutions when compared with the classic DE method and some methods found in the literature.

Keywords: Economic Dispatch Problem with the Valve Point Loading Effect, Differential Evolution Method, BFGS Method, Non-Linear Optimization.

LISTA DE ILUSTRAÇÕES

Figura 2.1 – Curva de entrada e saída com o efeito de ponto de carregamento de válvula.	26
Figura 2.2 – Curva de entrada e saída com zonas proibidas.	28
Figura 2.3 – Três possíveis estados de operação para uma unidade geradora.	29
Figura 4.1 – Exemplo bidimensional das curvas de nível de uma função objetivo e o processo para gerar o vetor mutante a partir de um vetor escolhido aleatoriamente.	49
Figura 4.2 – Exemplo bidimensional das curvas de nível de uma função objetivo e o proceso para gerar o vetor mutante a partir de um vetor escolhido aleatoriamente.	50
Figura 4.3 – a) Distribuição espacial da população na geração $t = 1$. b) Distribuição dos vetores-diferença na geração $t = 1$	51
Figura 4.4 – a) Distribuição espacial da população na geração $t = 10$. b) Distribuição dos vetores-diferença na geração $t = 10$	51
Figura 4.5 – a) Distribuição espacial da população na geração $t = 20$. b) Distribuição dos vetores-diferença na geração $t = 20$	52
Figura 4.6 – a) Distribuição espacial da população na geração $t = 10$. b) Distribuição dos vetores-diferença na geração $t = 10$	53
Figura 4.7 – a) Distribuição espacial da população na geração $t = 20$. b) Distribuição dos vetores-diferença na geração $t = 20$	53
Figura 4.8 – Ilustração do procedimento do cruzamento binomial em um vetor com 7 dimensões.	55
Figura 4.9 – Ilustração do procedimento do cruzamento exponencial em um vetor com 7 dimensões.	55
Figura 4.10 – Fluxograma do algoritmo de Evolução Diferencial.	57
Figura 5.1 – Fluxograma do algoritmo de ED-BFGS.	61
Figura 6.1 – Fluxograma do algoritmo de VSHDE.	67
Figura 6.2 – Fluxograma do algoritmo de HDE-MU.	76
Figura 6.3 – Fluxograma do algoritmo de IDE.	79
Figura 7.1 – a) Convergência da melhor solução do método ED para o caso 1. b) Convergência da melhor solução do método ED-BFGS para o caso 1.	82
Figura 7.2 – a) Convergência da melhor solução do método ED para o caso 2. b) Convergência da melhor solução do método ED-BFGS para o caso 2.	86
Figura 7.3 – a) Convergência da melhor solução do método ED para o caso 3. b) Convergência da melhor solução do método ED-BFGS para o caso 3.	89

Figura 7.4 – a) Convergência da melhor solução do método ED para o caso 4. b)
Convergência da melhor solução do método ED-BFGS para o caso 4. . 92

LISTA DE TABELAS

Tabela 4.1 – Estratégias de mutação.	48
Tabela 7.1 – Função custo para o PDE-PCV com 3 unidades geradoras.	82
Tabela 7.2 – Soluções para o PDE-PCV com 3 unidades geradoras.	82
Tabela 7.3 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 3 unidades geradoras.	83
Tabela 7.4 – Função custo para o PDE-PCV com 13 unidades geradoras.	84
Tabela 7.5 – Soluções para o PDE-PCV com 13 unidades geradoras.	85
Tabela 7.6 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 13 unidades geradoras.	85
Tabela 7.7 – Função custo para o PDE-PCV com 19 unidades geradoras.	87
Tabela 7.8 – Soluções para o PDE-PCV com 19 unidades geradoras.	87
Tabela 7.9 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 19 unidades geradoras.	88
Tabela 7.10–Função custo para o PDE-PCV com 40 unidades geradoras.	90
Tabela 7.11–Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 40 unidades geradoras.	90
Tabela 7.12–Soluções para o PDE-PCV com 40 unidades geradoras.	91
Tabela A.1–Dados para o caso 1.	100
Tabela A.2–Dados para o caso 2.	100
Tabela A.3–Dados para o caso 3.	101
Tabela A.4–Dados para o caso 4.	102

LISTA DE ABREVIATURAS E SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
AEs	Algoritmos Evolutivos
AGs	Algoritmos Genéticos
ACO	<i>Ant Colony Optimization</i>
CED	<i>Cogeneration Economic Dispatch</i>
EEs	<i>Estratégias Evolutivas</i>
ED	Evolução Diferencial
ED-SQP	Evolução Diferencial Híbrida com Programação Quadrática Sequencial
HDE	<i>Hybrid Differential Evolution</i>
HDE-MU	<i>Hybrid Differential Evolution with Multiplier Updating</i>
HGA	<i>Hybrid Genetic Algorithm</i>
MPI	Método de Ponto Interior
PP	Pesquisa de Padrão
PDED	Problema de Despacho Econômico Dinâmico
PSO	<i>Particle Swarm Optimization</i>
PDE	Problema de Despacho Econômico
PDE-PCV	Problema de Despacho Econômico com o efeito de Ponto de Carregamento de Válvula
PCV	Ponto de Carregamento de Válvula
SQP	Programação Quadrática Sequencial
SA	Simulated Annealing
SIN	Sistema Interligado Nacional
TS	<i>Tabu Search</i>
VSHDE	<i>Variable Scaling Hybrid Differential Evolution</i>
ZOP	Zonas de Operação Proibidas

SUMÁRIO

1	INTRODUÇÃO	21
1.1	Organização do trabalho	23
2	PROBLEMAS DE DESPACHO	25
2.1	Problema de Despacho Econômico	25
2.1.1	Modelo de otimização para o problema de despacho econômico com ponto de carregamento de válvula	30
3	MÉTODOS DE OTIMIZAÇÃO	32
3.1	Métodos Quase-Newton	32
3.1.1	Método BFGS (NOCEDAL; WRIGHT, 2006)	34
3.1.2	Método da Função Penalidade (BAZARAA; SHERALI; SHETTY, 1979)	40
4	COMPUTAÇÃO EVOLUTIVA	43
4.1	Métodos bio-inspirados	43
4.2	Métodos não bio-inspirados	45
4.2.1	Evolução diferencial	45
4.2.2	Mutação	48
4.2.3	Comportamento da mutação diferencial para otimizar uma função convexa	50
4.2.4	Função multimodal	52
4.2.5	<i>Crossover</i>	53
5	METODOLOGIA DE RESOLUÇÃO PROPOSTA: EVOLUÇÃO DIFERENCIAL HÍBRIDA COM O MÉTODO BFGS	58
5.1	Evolução diferencial híbrida com o método BFGS	58
6	MÉTODOS HÍBRIDOS BASEADOS NO ALGORITMO DE EVOLUÇÃO DIFERENCIAL PARA A RESOLUÇÃO DO PROBLEMA DE DESPACHO ECONÔMICO	63
6.1	O trabalho de Coelho e Mariani (2006a)	63
6.2	O trabalho de Chiou (2007)	65
6.3	O trabalho de Wang, Chiou e Liu (2007)	68
6.4	O trabalho de He, Wang e Zhizhong (2008)	70
6.5	O trabalho de Yuan et al. (2009)	72
6.6	O trabalho de Chiang e Wang (2010)	75

6.7	O trabalho de Duvvuru e Swarup (2011)	76
6.8	O trabalho de Zou et al. (2016)	77
7	TESTES NUMÉRICOS	81
7.1	Caso 1: 3 unidades geradoras	81
7.2	Caso 2: 13 unidades geradoras	84
7.3	Caso 3: 19 unidades geradoras	86
7.4	Caso 4: 40 unidades geradoras	89
8	CONCLUSÕES	93
	REFERÊNCIAS	95
A	DADOS DOS PROBLEMAS DE DESPACHO ECONÔMICO	100
A.1	Caso de teste 1: 3 unidades geradoras	100
A.2	Caso de teste 2: 13 unidades geradoras	100
A.3	Caso de teste 3: 19 unidades geradoras	101
A.4	Caso de teste 4: 40 unidades geradoras	102
B	REVISÃO TEÓRICA DE OTIMIZAÇÃO	103
B.1	Conceitos básicos	103
B.1.1	Matriz Hessiana	103
B.1.2	Matriz definida positiva ou semidefinida positiva	103
B.1.3	Matriz definida negativa ou semidefinida negativa	104
B.1.4	Matriz singular ou não singular	104
B.1.5	Fórmula de Taylor	104
B.1.6	Funções convexas	104
B.2	Otimização não linear	105
B.2.1	Otimização irrestrita	105
B.2.2	Condições de otimalidade para otimização irrestrita	106
B.2.3	Otimização restrita	107
B.2.3.1	Restrição de igualdade	107
B.2.3.2	Restrição de desigualdade	108
B.2.3.3	Restrições de igualdade e desigualdade	110

1 INTRODUÇÃO

A energia elétrica é um recurso de extrema importância na vida dos seres humanos, desde suas atividades cotidianas mais simples como acender uma lâmpada em suas casas, até as mais complexas como permitir o funcionamento de máquinas de grande porte. Para essa energia elétrica chegar até nossas casas e trabalhos, ela percorre um longo caminho que vai desde sua geração e transmissão até a distribuição. A geração de energia pode ser feita através da força da água, como o que acontece nas hidrelétricas; pela força dos ventos, chamada de energia eólica; pela luz do sol, chamada de energia solar ou também pela queima de biomassa e combustíveis fósseis feita nas termoeletricas. A energia gerada por essas fontes circula pelo Sistema Interligado Nacional (SIN) que conecta as diversas usinas às redes de distribuição de todo o país. E por fim, quem é responsável para que essa energia elétrica chegue até nossas casas são as distribuidoras de energia elétrica ([ANEEL, 2010](#)).

Ainda que o Brasil disponha de uma matriz energética elétrica amparada nas usinas hidrelétricas, os níveis de água dos reservatórios não são sempre os mesmos devido à sazonalidade. Exemplo disso é a escassez de chuvas que ocorre em diferentes épocas do ano. Esse fato fez com que o Brasil investisse mais em usinas que não dependessem somente do ciclo da água, que é o caso das usinas termoeletricas. Como essas usinas vêm se tornando cada vez mais relevantes em relação a geração de energia elétrica no país, é de extrema importância que estratégias de minimização dos custos de geração sejam desenvolvidas. Para tal, tem-se o problema de despacho econômico (PDE). O PDE tem como objetivo reduzir o custo total para geração de energia determinando o despacho de cada unidade vinculada à rede. Sendo assim, o propósito é gerar energia com o menor custo de combustível possível atendendo à demanda dos consumidores e respeitando todas as restrições do sistema de geração ([ABIDO, 2003](#)). Logo, justifica-se a importância do uso do despacho econômico na otimização da operação de sistemas de energia termoeletrica.

Existem diversos modelos de otimização na área de Engenharia que são difíceis e até mesmo impossíveis de serem resolvidos através de métodos exatos de otimização. Essa dificuldade ocorre devido à modelagem matemática do problema, descontinuidades ou não diferenciabilidade, entre outras características ([SERAPIÃO, 2009](#)). Desde que o PDE foi introduzido, vários métodos têm sido utilizados para resolvê-lo, como técnicas baseadas em gradiente ([WOOD; WOLLENBERG; SHEBLE, 2013](#)), método dos pontos interiores ([GONÇALVES, 2015](#)), entre outros. Nesse segmento, algoritmos bio-inspirados baseados em populações vêm sendo usados para resolver esses tipos de problemas. Os algoritmos bio-inspirados utilizam um método construtivo para obter uma população inicial, a qual é

composta das possíveis soluções para o problema alvo. A partir desta população inicial, aplicam-se técnicas de busca local e global para melhorar as soluções da população de forma a encontrar soluções de boa qualidade (SERAPIÃO, 2009).

Encontram-se na literatura muitos estudos que utilizam métodos heurísticos para a resolução do PDE, tais como busca tabu (do inglês *tabu search* ou TS) (LIN; CHENG; TSAY, 2002), enxame de partículas (do inglês *particle swarm optimization* ou PSO) (EBERHART; KENNEDY, 1995), recozimento simulado (do inglês *simulated annealing* ou SA) (WONG; FUNG, 1993), evolução diferencial (do inglês *differential evolution* ou DE) (STORN; PRICE, 1997), entre outros. Dentre esses métodos, destaca-se o método evolução diferencial (ED), o qual tem tido sucesso na resolução do PDE, e utilizado em trabalhos como de Coelho e Mariani (2006a), Andrade (2019), Zou et al. (2016), Duvvuru e Swarup (2011), Zhang et al. (2019), entre outros. Esse método utiliza o vetor de diferença de dois vetores populacionais escolhidos aleatoriamente para perturbar um vetor existente e essa perturbação é feita para cada vetor populacional (STORN; PRICE, 1997). A ideia contrasta com o método usado pelos algoritmos evolutivos (AEs) tradicionais, em que funções predeterminadas de distribuição de probabilidade determinam perturbações vetoriais (STORN; PRICE, 1997).

Neste trabalho, a atenção será dirigida a uma variante do PDE, denominada problema de despacho econômico com o efeito de ponto de carregamento de válvula (PDE-PCV). Esta variante é obtida a partir do PDE ao introduzir na função objetivo o efeito de ponto de carregamento de válvula (PCV) das unidades geradoras. O efeito de PCV está relacionado ao momento em que são abertas as válvulas da unidade geradora, liberando pressão e conseqüentemente diminuindo a eficiência energética da unidade geradora (JERONYMO, 2011). Na modelagem matemática, o efeito de PCV é representado por um valor absoluto senoidal (WALTERS; SHEBLE, 1993). O ponto de válvula torna a função objetivo não convexa e não diferenciável, tornando o modelo difícil de ser solucionado através de métodos de otimização matemáticos determinísticos diretamente. Dessa forma, justifica-se o uso de heurísticas de otimização, como a ED, para solucionar esse tipo de problema.

Como visto, o PDE-PCV é não convexo e não diferenciável, não sendo possível resolvê-lo por métodos de otimização matemáticos determinísticos diretamente. Isto acontece porque geralmente estes métodos utilizam informações associadas ao gradiente e à matriz Hessiana das funções com a finalidade de calcular as direções de busca. A não convexidade também limita o uso de métodos determinísticos, visto que estes normalmente convergem para mínimos locais. Por isso, o PDE-PCV tem sido explorado na literatura predominantemente utilizando métodos heurísticos ou heurísticos híbridos com métodos determinísticos. Essa hibridização tem a finalidade de melhorar o desempenho dos métodos heurísticos.

Desse modo, o objetivo deste trabalho é apresentar um método heurístico híbrido de otimização que integra o método ED proposto por [Storn e Price \(1997\)](#) para a etapa de busca global, com o método determinístico desenvolvido por Broyden, Fletcher, Goldfarb e Shanno (BFGS) para a etapa de busca local. A rotina BFGS aplicada neste trabalho é proporcionada pelo pacote *Matlab Optimization Toolbox*, utilizando a função *fminunc*. Visto que a função objetivo do PDE-PCV é não diferenciável e o método BFGS necessita do valor do gradiente, a função *fminunc* utiliza um método de diferença finita para fornecer esse valor ([COLEMAN; BRANCH; GRACE, 1999](#)).

As principais vantagens da ED são sua estrutura simples, velocidade, propriedade de busca global e não exige esforço computacional adicional para a representação binária, como é o caso dos algoritmos genéticos (AGs) ([HOLLAND, 1992](#)) clássicos. O método BFGS pode explorar a região local rapidamente, mas esta técnica de busca é sensível aos pontos de partida iniciais e pode convergir para uma solução ótima local. Outra vantagem do método BFGS está em não precisar recalcular a matriz Hessiana a cada iteração. Sendo assim, na metodologia proposta, o método BFGS utilizará a melhor solução obtida pelo método ED como ponto de partida.

A contribuição deste trabalho é descrever e avaliar uma nova metodologia híbrida para a resolução do PDE-PCV. A vantagem da utilização de um método híbrido de otimização de ED e BFGS está na melhoria da velocidade de convergência do algoritmo de otimização em relação à busca na vizinhança de um ponto promissor, bem como na obtenção de soluções melhores que a ED clássica.

Testes numéricos envolvendo o PDE-PCV com restrições de balanço de potência e limites de operação das unidades geradoras foram realizados. Foram realizados testes numéricos com os sistemas elétricos contendo 3, 13, 19 e 40 unidades geradoras. Os resultados demonstram a eficiência do método híbrido proposto na resolução do PDE-PCV quando comparados com o método ED clássico e outros métodos encontrados na literatura, tais como [Silva \(2014\)](#), [Zhao et al. \(2018\)](#), [He, Rao e Huang \(2016\)](#), [Coelho e Mariani \(2006a\)](#), [Elsayed et al. \(2017\)](#) e [Zou et al. \(2016\)](#).

1.1 Organização do trabalho

O presente trabalho encontra-se organizado da seguinte forma:

No Capítulo 2, o PDE com e sem os efeitos de PCV é descrito, bem como suas respectivas formulações matemáticas.

No Capítulo 3, são apresentados métodos numéricos utilizados para a resolução de problemas de programação não linear e um resumo sobre os mesmos, os quais serão importantes para o desenvolvimento do método híbrido proposto neste trabalho.

No capítulo 4, são abordadas as características de métodos inspirados em mecanismos de adaptação dos seres vivos, conforme observados na natureza, eficazes para resolver diversos problemas de otimização. É destacado o método ED utilizado neste trabalho.

No Capítulo 5, é proposta a metodologia para a resolução do PDE-PCV.

No capítulo 6, é realizado um levantamento de trabalhos que tratam do PDE-PCV através de métodos híbridos baseados no algoritmo de ED encontrados na literatura.

No Capítulo 7, são apresentados os resultados obtidos da aplicação do método proposto à resolução do PDE-PCV.

Por fim, no Capítulo 8 são feitas as conclusões sobre o trabalho desenvolvido. Em seguida, são apresentadas as referências bibliográficas.

2 PROBLEMAS DE DESPACHO

Os problemas de despacho são encontrados na área de sistemas de energia elétrica, cuja finalidade é obter a geração de energia termoelétrica que minimize os custos de combustíveis utilizados na geração, o que é conhecido na literatura como PDE. Neste capítulo, será descrito o PDE-PCV e seu respectivo modelo matemático.

2.1 Problema de Despacho Econômico

O PDE é um problema matemático de otimização não linear que consiste na distribuição da demanda total às unidades geradoras de energia a fim de minimizar os custos de produção (GOMEZ-EXPOSITO; CONEJO; CANIZARES, 2011). O PDE também deve representar a restrição de atendimento da demanda total de carga do sistema e os limites operacionais dos geradores, que incluem pontos de válvula, zonas proibidas, limites de operação e limites de rampa (GAING, 2003). Portanto, ao resolver um PDE, será determinado o despacho ótimo de potência ativa de cada unidade geradora, minimizando os custos de combustíveis ao mesmo tempo que as restrições do sistema são atendidas. Dependendo do tipo da fonte de energia que é utilizada nas unidades geradoras para gerar eletricidade, os custos de produção variam (GOMEZ-EXPOSITO; CONEJO; CANIZARES, 2011).

Dessa forma, o PDE pode ser considerado como um problema de otimização não linear, no qual deseja-se minimizar a função objetivo que determina o custo mínimo de combustível. Para cada uma das unidades geradoras é definida uma função custo $F_e(P)$ proposta por Steinberg e Smith (2011) que caracteriza os custos de geração em termos da potência produzida. O custo de geração total é igual ao somatório da função custo de combustível de cada gerador:

$$F_e(P) = \sum_{i=1}^n F_i(P_i) = \sum_{i=1}^n a_i P_i^2 + b_i P_i + c_i, \quad (2.1)$$

em que:

$F_e(P)$: é a função custo de combustível total;

$F_i(P_i)$: é a função custo de combustível individual da i -ésima unidade geradora;

a_i , b_i e c_i : parâmetros da função custo de produção da unidade de geração i ;

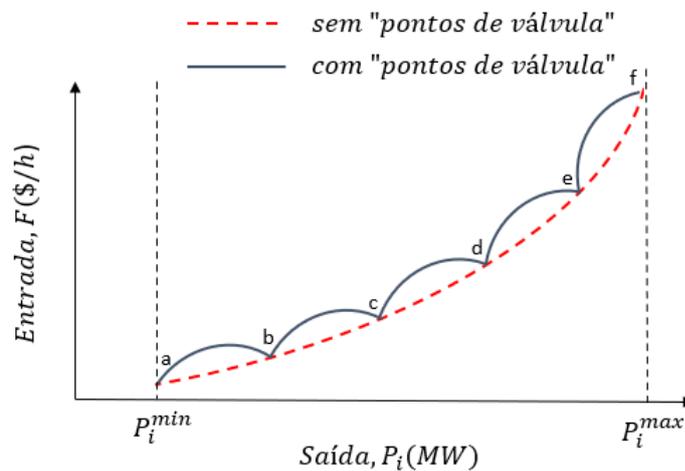
P_i : é a potência gerada pela unidade geradora i ;

n : número de unidades geradoras.

Grande parte dos trabalhos encontrados na literatura utilizam a função custo definida em (2.1). Essa função é uma aproximação quadrática das funções custo dos geradores, o que evita dificuldades referentes a problemas não convexos. Todavia, para a realização de testes que retratem a realidade nas usinas termoeletricas, esta função não é adequada. No caso de geradores com turbinas à vapor, existem válvulas de admissão de vapor, cuja abertura interfere na saída da unidade geradora. Esse fenômeno é chamado de efeito de PCV, e ocorre quando cada válvula em uma turbina começa a se abrir (WALTERS; SHEBLE, 1993). A Figura 2.1 apresenta a curva de entrada e saída de uma unidade geradora i , onde o eixo das ordenadas contém a entrada como sendo a função custo (\$/h) e no eixo das abscissas é representada a saída como a potência gerada (em MW).

Na Figura 2.1, podemos observar que a válvula se abriu nos pontos a , b , c , d , e e f , os quais são conhecidos como pontos de válvula (WALTERS; SHEBLE, 1993). Isso acontece porque ao ser preciso aumentar a potência de saída de uma unidade geradora, uma quantidade maior de vapor deverá ser injetada na turbina. Esse efeito provoca ondulações na função custo da unidade geradora, pois há uma variação na quantidade de combustível injetado durante esse processo, ou também pela possível mudança de combustível injetado para aquele nível de geração.

Figura 2.1 – Curva de entrada e saída com o efeito de ponto de carregamento de válvula.



Fonte: Adaptado de De-Oliveira, Nascimento-Jr e Saavedra (2008).

O efeito de PCV no custo total é modelado por um valor absoluto senoidal. Então, a equação (2.1) para o cálculo do custo de combustível pode ser modificada para considerar os efeitos de PCV de modo que (WALTERS; SHEBLE, 1993):

$$F_e(P) = \sum_{i=1}^n F_i(P_i) = \sum_{i=1}^n \left[a_i P_i^2 + b_i P_i + c_i + \left| e_i \text{sen}(f_i(P_i^{min} - P_i)) \right| \right], \quad (2.2)$$

onde e_i e f_i são constantes associadas ao efeito de PCV dos geradores. Conseqüentemente, o custo mínimo se dará minimizando (2.2), em que o efeito de PCV estará sendo considerado.

A função módulo de seno que aparece na função custo (2.2) é responsável por modelar os pontos de carregamento de válvula. Ao mesmo tempo em que esta nova função objetivo é mais completa, ela também introduz novas dificuldades ao problema de otimização, tais como a não convexidade e não diferenciabilidade.

No PDE-PCV, a potência demandada deve satisfazer o balanço de potência, isto é, a soma das potências das unidades geradoras devem atender à demanda total do sistema, denominada P_D . O balanço de potência é definido através da seguinte equação:

$$\sum_{i=1}^n P_i = P_D, \quad (2.3)$$

onde P_D representa a potência demandada pela carga, a qual em geral adota-se como constante.

Em todo sistema elétrico de potência ocorrem perdas de potência nas linhas de transmissão. Essa potência desperdiçada no atendimento às cargas também é fornecida pelos geradores que estão sendo despachados. Logo, essas perdas podem ser inseridas ao problema como um tipo de demanda da rede. Deste modo, além da potência gerada pelas unidades geradoras ter de satisfazer a demanda de potência, será preciso também considerar as perdas de potência nas linhas de transmissão. Ou seja, a potência gerada deve ser igual à soma da potência demandada pelo sistema e das perdas de potência nas linhas de transmissão, conforme a equação:

$$\sum_{i=1}^n P_i = P_D + P_L(P), \quad (2.4)$$

sendo que P_L é a perda de potência nas linhas de transmissão do sistema, a qual é calculada como função das saídas das unidades geradoras usando a matriz de coeficiente de perda B a partir da fórmula de perdas de Kron (SERAPIÃO, 2009):

$$P_L(P) = \sum_{i=1}^n \sum_{j=1}^n B_{ij} P_i P_j + \sum_{i=1}^n B_{0i} P_i + B_{00}, \quad (2.5)$$

em que:

B_{ij} é o ij -ésimo elemento da matriz de coeficiente de perda;

B_{0i} é o i -ésimo elemento do vetor de coeficiente de perda;

B_{00} é a constante do coeficiente de perda.

Uma restrição muito comum e necessária ao PDE-PCV é a restrição de desigualdade referente aos limites da capacidade de geração de potência de cada uma das unidades geradoras, representada na Figura 2.1 como P_i^{min} e P_i^{max} . As restrições de desigualdade referentes aos limites mínimo e máximo de geração de potência são descritas por (WOOD; WOLLENBERG; SHEBLE, 2013):

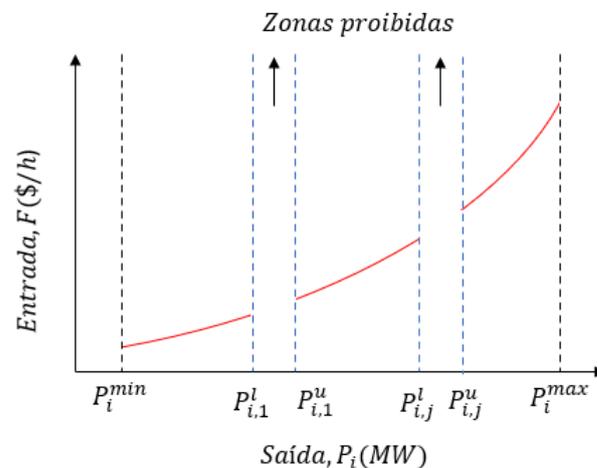
$$P_i^{min} \leq P_i \leq P_i^{max}, \quad \forall i \in G, \quad G = \{1, 2, \dots, n\} \quad (2.6)$$

onde P_i^{min} e P_i^{max} são, respectivamente, os limites mínimo e máximo de potência da unidade geradora i , e G é o conjunto de geradores do sistema. Essas restrições garantem que os geradores operem somente dentro de suas capacidades de geração.

Por motivos físicos de operação dos geradores, nem sempre é possível operar em todas as faixas devido à oscilações das válvulas à vapor no eixo da máquina ou nos serviços auxiliares como caldeiras, bombas de alimentação, entre outras causas (DE-OLIVEIRA; NASCIMENTO-JR; SAAVEDRA, 2008). Esses fatores influenciam a origem de zonas de operação proibidas, logo deve-se evitar operar nessas áreas. Na Figura 2.2 é apresentada a curva de entrada e saída de uma unidade geradora i com zonas de operação proibidas, juntamente com as restrições sobre a potência gerada de cada unidade geradora P_i com limite de operação mínimo e máximo.

Observa-se na Figura 2.2 que os limites de operação são divididos em várias regiões. Isso implica que, além de considerar os limites de operação mínimo e máximo, haverá também os limites intermediários caracterizando a restrição de zonas de operação proibidas.

Figura 2.2 – Curva de entrada e saída com zonas proibidas.



Fonte: Adaptado de De-Oliveira, Nascimento-Jr e Saavedra (2008) e GAING (2003).

Segundo GAING (2003), a melhor maneira de tratar o problema com zonas de operação proibidas é ajustar a saída da geração P_i de forma a evitar a operação da unidade

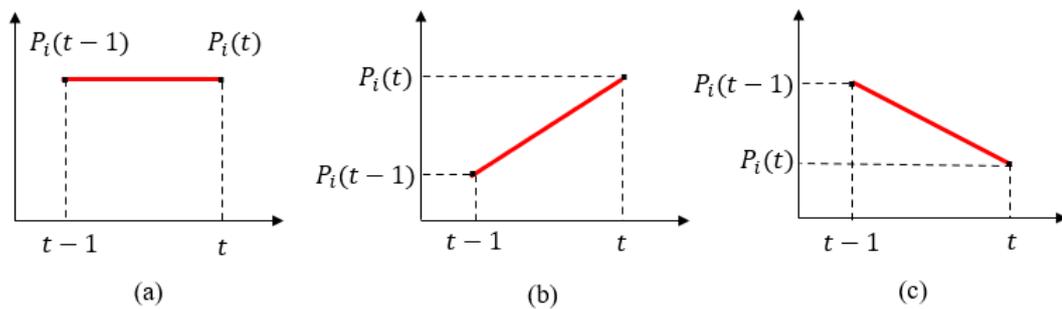
nas zonas proibidas. Isso é feito modificando as restrições de limites mínimos e máximos de operação. As zonas possíveis de operação são descritas da seguinte forma:

$$P_i \in \left([P_i^{min}, P_i^{max}] - \cup_{j=1}^{n_i} [P_{i,j}^t, P_{i,j}^u] \right), \quad (2.7)$$

sendo n_i o número de zonas de operação proibidas; $P_{i,j}^l$ é o limite inferior da j -ésima zona proibida da i -ésima unidade geradora; e $P_{i,j}^u$ é o limite superior da j -ésima zona proibida da i -ésima unidade geradora.

Segundo [GAING \(2003\)](#), por conveniência na solução do PDE, a potência gerada pela unidade geradora é considerada como sendo ajustada de forma suave e instantânea. Na prática, a saída do gerador não pode ser ajustada instantaneamente. As variações na potência de saída de uma unidade geradora ocorrem constantemente e cada unidade geradora possui valores máximos de acréscimo e de decréscimo de potência por hora. Esses valores são conhecidos como limites de rampa, em razão dessas variações de potência seguirem a forma de rampa, conforme pode ser observado na [Figura 2.3](#). Logo, a faixa de operação de todas as unidades geradoras é restrita por seus limites de rampa para forçar a operação das unidades continuamente entre dois períodos de operação específicos ([GAING, 2003](#)).

Figura 2.3 – Três possíveis estados de operação para uma unidade geradora.



Fonte: Adaptado de [De-Oliveira, Nascimento-Jr e Saavedra \(2008\)](#).

Conforme a [Figura 2.3](#), existem três possíveis situações considerando os limites de rampa quando uma unidade geradora está funcionando mudando de um instante $t-1$ para um instante t . Uma unidade geradora pode operar em estado constante ([Figura 2.3 \(a\)](#)), aumentar sua geração de potência ([Figura 2.3 \(b\)](#)) ou diminuir sua geração de potência ([Figura 2.3 \(c\)](#)).

As restrições de desigualdades para os limites de rampa são dadas da seguinte forma ([GAING, 2003](#)):

- Se a geração aumenta:

$$P_i - P_i^0 \leq UR_i. \quad (2.8)$$

- Se a geração diminui:

$$P_i^0 - P_i \leq DR_i. \quad (2.9)$$

A inclusão de limites de taxa de rampa das restrições (2.8) e (2.9) modifica os limites de potência de saída conforme pode ser observado na seguinte expressão (GAING, 2003):

$$\max(P_i^{\min}, P_i^0 - DR_i) \leq P_i \leq \min(P_i^{\max}, P_i^0 + UR_i), \quad (2.10)$$

onde P_i é a potência de saída atual da unidade geradora i e P_i^0 é a potência de saída anterior da unidade geradora i . UR_i é o limite de rampa de subida do i -ésimo gerador (MW/período de tempo) e DR_i é o limite de rampa de descida do i -ésimo gerador (MW/período de tempo).

2.1.1 Modelo de otimização para o problema de despacho econômico com ponto de carregamento de válvula

Tendo em vista a função custo apresentada em (2.2) e as restrições apresentadas, o modelo de otimização para o PDE-PCV é expresso por:

$$\begin{aligned} \text{minimizar } F_e(P) &= \sum_{i=1}^n F_i(P_i) = \sum_{i=1}^n \left[a_i P_i^2 + b_i P_i + c_i + |e_i \text{sen}(f_i(P_i^{\min} - P_i))| \right] \\ \text{sujeito a: } \sum_{i=1}^n P_i &= P_D + P_L(P) \\ P_i^{\min} &\leq P_i \leq P_i^{\max}, \quad \forall i \in G \\ P_i &\in \left([P_i^{\min}, P_i^{\max}] - \cup_{j=1}^{n_i} [P_{i,j}^t, P_{i,j}^u] \right), \quad \forall i \in G, \quad j = 2, 3, \dots, n_i \\ \max(P_i^{\min}, P_i^0 - DR_i) &\leq P_i \leq \min(P_i^{\max}, P_i^0 + UR_i), \quad \forall i \in G \end{aligned} \quad (2.11)$$

em que:

$F_e(P)$: é a função custo de combustível total;

$F_{e_i}(P_i)$: é a função custo de combustível individual da i -ésima unidade geradora;

a_i, b_i, c_i, e_i e f_i : parâmetros da função custo de produção da unidade de geração i ;

P_i : é a potência ativa gerada pela unidade geradora i ;

P_D : potência ativa demandada;

P_L : perda de potência ativa transmissão do sistema;

P_i^{min} e P_i^{max} : limites mínimo e máximo, respectivamente, de saída de potência da unidade geradora i ;

n_i : número de zonas de operação proibidas;

$P_{i,j}^l$: limite inferior da zona proibida j da i -ésima unidade geradora;

$P_{i,j}^u$ é o limite superior da j -ésima zona proibida da i -ésima unidade geradora;

P_i^0 : potência de saída anterior da unidade geradora i ;

UR_i : limite de rampa subida do i -ésimo gerador;

DR_i : limite de rampa de descida do i -ésimo gerador.

Ressalta-se que, neste trabalho, o modelo para o PDE-PCV abordado nos casos do Capítulo 7, considera apenas as restrições de balanço de potência (2.3) e limites de operação das unidades geradoras (2.6).

No próximo capítulo, são apresentados os métodos de otimização que serão fundamentais neste trabalho.

3 MÉTODOS DE OTIMIZAÇÃO

Neste capítulo, apresentaremos alguns métodos que podem ser utilizados para otimização restrita e irrestrita. De interesse particular para este trabalho, são os métodos BFGS e Função Penalidade, que compõem o método híbrido proposto que será apresentado no Capítulo 5.

3.1 Métodos Quase-Newton

A fim de discutir sobre os métodos Quase-Newton, apresenta-se primeiro o desenvolvimento do método de Newton. Este método, também conhecido como método de Newton multidimensional, baseia-se em aproximar localmente a função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ a ser minimizada por uma função quadrática e então minimizar essa função. O método de Newton possui convergência quadrática, porém, esta convergência está diretamente ligada à escolha da solução inicial. Para definirmos a função que aproxima f vamos utilizar a série de Taylor até 2ª ordem. Assim, temos (BAZARAA; SHERALI; SHETTY, 1979):

$$f(x) \approx q(x) = f(x_k) + \nabla f^T(x_k)(x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k), \quad (3.1)$$

onde $x \in \mathbb{R}^n$, $\nabla f(x_k)$ é o vetor gradiente de f no ponto $x_k \in \mathbb{R}^n$ e $\nabla^2 f(x_k)$ é a matriz Hessiana de f no ponto x_k . A condição necessária para existência de ponto mínimo é que

$$\nabla f(x) = 0, \quad (3.2)$$

logo, considerando a aproximação quadrática,

$$\nabla q(x) = 0. \quad (3.3)$$

Então temos que,

$$\nabla q(x) = \nabla f(x_k) + \nabla^2 f(x_k)(x - x_k) = 0, \quad (3.4)$$

$$\nabla^2 f(x_k)(x - x_k) = -\nabla f(x_k). \quad (3.5)$$

Assumindo que a matriz Hessiana de f é não singular e fazendo $x = x_{k+1}$:

$$\nabla^2 f(x_k)(x_{k+1} - x_k) = -\nabla f(x_k), \quad (3.6)$$

$$(x_{k+1} - x_k) = -(\nabla^2 f(x_k))^{-1} \nabla f(x_k). \quad (3.7)$$

Logo, a expressão que caracteriza o método de Newton é definida por:

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k). \quad (3.8)$$

O critério de parada para o método de Newton pode ser estabelecido como:

$$\|\nabla f(x_k)\| < \epsilon, \quad (3.9)$$

sendo ϵ o valor do erro pré-estabelecido.

A convergência do método de Newton é garantida se a matriz Hessiana for definida positiva e não singular, e isto deve ser assegurado em cada iteração do método. Algumas modificações podem ser feitas no método de Newton com o intuito de melhorar a sua convergência. Uma destas modificações consiste em introduzir um parâmetro de busca α_k , o qual é determinado por uma busca unidimensional, isto é:

$$\text{minimizar}_{\alpha} f(x_k + \alpha d_k). \quad (3.10)$$

Deste modo, o método de Newton com busca unidimensional é dado por:

$$x_{k+1} = x_k + \alpha_k d_k, \quad (3.11)$$

com $d_k = -(\nabla^2 f(x_k))^{-1} \nabla f(x_k)$ sendo uma direção de descida.

O algoritmo do método de Newton para minimizar f , é apresentado a seguir.

Algoritmo 1: Método de Newton.

Entrada: $f : \mathbb{R}^n \rightarrow \mathbb{R}$ continuamente diferenciável, ϵ a precisão desejada e x_0 o ponto inicial;

$k \leftarrow 0$;

Saída: Ponto de mínimo;

enquanto $\|\nabla f(x_k)\| \geq \epsilon$, **faça**

$$\begin{aligned}d_k &\leftarrow -[\nabla^2 f(x_k)]^{-1} \nabla f(x_k); \\x_{k+1} &= x_k + d_k; \\k &\leftarrow k + 1;\end{aligned}$$

fm

Retorna x_k

Os métodos Quase-Newton, ao invés de calcularem a inversa da matriz Hessiana, como no método de Newton mostrado anteriormente, aproxima esta inversa em um processo iterativo finito, utilizando apenas derivadas de primeira ordem. São analiticamente os métodos mais sofisticados para os problemas irrestritos, pois aplicar um método de Newton é computacionalmente inviável devido ao custo de se avaliar a matriz Hessiana. Os métodos Quase-Newton são uma alternativa para resolver este impasse, pois constroem uma aproximação da inversa da Hessiana utilizando apenas informações da função erro.

A seguir, apresenta-se os métodos Quase-Newton Davidon-Fletcher-Powell (DFP) e BFGS.

3.1.1 Método BFGS (NOCEDAL; WRIGHT, 2006)

Um método Quase-Newton bastante popular encontrado na literatura é o método BFGS, nomeado por seus descobridores Broyden, Fletcher, Goldfarb e Shanno. Analisaremos primeiro o método Davidon-Fletcher-Powell (DFP), do qual se originou o método BFGS.

Davidon apresentou o método DFP no ano de 1959 e posteriormente, Fletcher e Powell o aprimoraram em 1963. O método DFP se enquadra na classe geral de procedimentos Quase-Newton, onde as direções de busca são da forma $d_k = -B_k \nabla f(x_k)$ ao invés de $-(\nabla^2 f(x_k))^{-1} \nabla f(x_k)$, como no método de Newton. A matriz B_k é uma matriz simétrica definida positiva $n \times n$ que se aproxima da inversa da matriz Hessiana e é atualizada em cada iteração do método. A matriz ser simétrica e definida positiva garante que d_k seja uma direção de descida sempre que $\nabla f(x_k) \neq 0$, desde que $d_k^k = -\nabla f(x_k)$ (BAZARAA; SHERALI; SHETTY, 1979).

Considere o seguinte modelo quadrático para a função objetivo no ponto x_k :

$$m_k(d) = f(x_k) + \nabla f(x_k)^T d + \frac{1}{2} d^T B_k d. \quad (3.12)$$

Note que, no ponto $d = 0$, o valor do modelo e seu gradiente coincidem com o valor da função objetivo e de seu gradiente.

O minimizador d_k para o modelo quadrático é dado por:

$$d_k = -B_k^{-1}\nabla f(x_k), \quad (3.13)$$

e o novo iterando será expresso por:

$$x_{k+1} = x_k + \alpha_k d_k, \quad (3.14)$$

com α_k satisfazendo as condições de Wolfe (ver [Bazaraa, Sherali e Shetty \(1979\)](#)).

A diferença desta iteração com a iteração do método de Newton, é a aproximação B_k no lugar da Hessiana. Ou seja, ao invés de recalcular B_k a cada nova iteração, esta será atualizada a cada iteração utilizando a curvatura obtida na iteração atual e na anterior.

Na iteração x_{k+1} teremos o modelo quadrático:

$$m_{k+1}(d) = f(x_{k+1}) + \nabla f(x_{k+1})^T d + \frac{1}{2} d^T B_{k+1} d. \quad (3.15)$$

Uma condição que B_{k+1} deve satisfazer é que o gradiente de m_{k+1} coincida com o gradiente da função f em x_k e x_{k+1} . Mas como $\nabla m_{k+1}(0) = \nabla f(x_{k+1})$, a segunda dessas condições é satisfeita automaticamente. A primeira condição pode ser escrita matematicamente como:

$$\nabla m_{k+1}(-\alpha_k d_k) = \nabla f(x_{k+1}) - \alpha_k B_{k+1} d_k = \nabla f(x_k), \quad (3.16)$$

ou seja,

$$B_{k+1}(\alpha_k d_k) = \nabla f(x_{k+1}) - \nabla f(x_k). \quad (3.17)$$

Definem-se os seguintes vetores para simplificar a expressão acima:

$$s_k = x_{k+1} - x_k, \quad y_k = \nabla f(x_{k+1}) - \nabla f(x_k). \quad (3.18)$$

Com isso, de (3.17) tem-se a seguinte equação, chamada de equação secante:

$$B_{k+1} s_k = y_k. \quad (3.19)$$

Realizadas as mudanças em s_k e em seus gradientes y_k , a equação secante exige que a matriz simétrica definida positiva mapeie s_k em y_k . Para isso ocorrer, s_k e y_k devem satisfazer a seguinte condição de curvatura:

$$s_k^T y_k > 0. \quad (3.20)$$

Quando f é fortemente convexa, a desigualdade $s_k^T y_k > 0$ é satisfeita para quaisquer dois pontos x_k e x_{k+1} . No entanto, esta condição nem sempre valerá para funções não-convexas. Neste caso, teremos de forçar que ela valha explicitamente, impondo restrições à busca linear que calcula α_k . Quando α_k satisfaz as condições de Wolfe ou as condições fortes de Wolfe, temos que $s_k^T y_k > 0$. Para verificar este fato, basta notar que, nas condições de Wolfe, para qualquer direção de descida, tem-se:

$$\nabla f(x_{k+1})^T s_k \geq c_2 \nabla f(x_k)^T s_k, \quad 0 < c_2 < 1, \quad (3.21)$$

ou seja,

$$y_k^T s_k \geq c_2 \nabla f(x_k)^T s_k - \nabla f(x_k)^T s_k \quad (3.22)$$

$$y_k^T s_k \geq (c_2 - 1) \alpha_k \nabla f(x_k)^T s_k. \quad (3.23)$$

Sendo assim, segue que

$$s_k^T y_k \geq (c_2 - 1) \alpha_k \nabla f(x_k)^T s_k. \quad (3.24)$$

Como d_k é uma direção de descida em relação a x_k , $c_2 < 1$ e $\alpha_k > 0$, a condição de curvatura é satisfeita e com isso a equação secante tem solução B_{k+1} . De fato, infinitas soluções são possíveis, já que o grau de liberdade em uma matriz simétrica é $n(n+1)/2$ e a equação secante representa apenas n restrições. A condição de que B_{k+1} deve ser definida positiva impõe mais n restrições, mas estas condições não absorvem todo o grau de liberdade. Então, para determinar B_{k+1} de maneira única, impomos mais uma condição, onde a matriz simétrica na equação secante, B_{k+1} , deve ser, em algum sentido, a mais próxima da matriz atual B_k . Portanto, queremos encontrar a matriz solução para o seguinte problema:

$$\begin{aligned} & \text{minimizar}_B \quad \|B - B_k\| \\ & \text{sujeito a: } B = B^T \\ & \quad \quad \quad B s_k = y_k \end{aligned} \quad (3.25)$$

com B_k simétrica definida positiva e $y_k^T s_k > 0$.

Uma norma de matriz na qual resolve-se facilmente o problema (3.25), é a norma de Frobenius com peso:

$$\|A\|_W = \left\| W \frac{1}{2} A W \frac{1}{2} \right\|_2. \quad (3.26)$$

A matriz W pode ser qualquer matriz definida positiva que satisfaça a relação $W y_k = s_k$. Para sermos mais objetivos, podemos supor que $W = \bar{G}_k^{-1}$, em que \bar{G}_k é a Hessiana média definida por:

$$\bar{G}_k = \int_0^1 \nabla^2 f(x_k + \tau \alpha_k d_k) d\tau. \quad (3.27)$$

Segue do teorema de Taylor a propriedade $y_k = \bar{G}_k \alpha_k d_k = \bar{G}_k s_k$. Com essa escolha de W , a norma (3.26) é adimensional, e como queremos que a matriz B_{k+1} não dependa das unidades do problema, isso é necessário. Com esta matriz W de peso e a norma de Frobenius com peso, a solução única para o problema (3.25) é dada por

$$B_{k+1} = (I - \rho_k y_k s_k^T) B_k (I - \rho_k s_k y_k^T) + \rho_k y_k y_k^T, \quad (3.28)$$

onde

$$\rho_k = \frac{1}{y_k^T s_k}. \quad (3.29)$$

A expressão (3.28) é chamada de fórmula de atualização DFP, já que foi originalmente proposta por Davidon e depois estudada, implementada e popularizada por Fletcher e Powell.

A inversa de B_k , dada por $H_k = B_k^{-1}$, é bastante útil na implementação do método, já que permite que a direção d_k seja calculada por um produto matriz-vetor. A atualização da inversa da aproximação da Hessiana H_k é definida utilizando a fórmula de Sherman-Morrison-Woodbury e esta atualização corresponde à atualização DFP de B_k :

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{y_k^T s_k}. \quad (3.30)$$

O algoritmo do método DFP consiste no seguinte:

Algoritmo 2: Método DFP.

Entrada: Sejam $x_0 \in \mathbb{R}^n$ arbitrário e $H_0 \in \mathbb{R}^{n \times n}$ uma matriz simétrica e definida positiva.

Saída: Ponto de mínimo;

enquanto $\|\nabla f(x_k)\| \neq 0$, **faça**

$$d_k = -H_k \nabla f(x_k);$$

Determinar α_k através de uma busca linear e definir $x_{k+1} = x_k + \alpha_k d_k$;

Definir $s_k = x_{k+1} - x_k$, $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ e calcular

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{y_k^T s_k};$$

$$k \leftarrow k + 1;$$

fim

Retorna *Ponto de mínimo*

O que caracteriza o método DFP é a fórmula recursiva para atualizar H_k . A matriz H_{k+1} é obtida a partir de uma correção de H_k que consiste em somar duas matrizes simétricas. A vantagem em termos de esforço computacional é que o número de operações para determinar d_k é da ordem de n^2 , em lugar de n^3 como no método de Newton (FRIEDLANDER, 1994).

A fórmula de atualização DFP é muito eficaz, mas foi rapidamente substituída pela fórmula BFGS. Para chegar à fórmula de atualização BFGS, basta fazer uma pequena mudança nos argumentos que levaram à fórmula DFP, ou seja, em vez de impor condições à aproximação da Hessiana B_k , são impostas condições similares à sua inversa H_k . A aproximação atualizada H_{k+1} deve ser simétrica definida positiva e deve satisfazer a equação secante:

$$H_{k+1} y_k = s_k. \quad (3.31)$$

De forma análoga a (3.25), a condição de proximidade a H_k é dada da seguinte forma:

$$\begin{aligned} & \text{minimizar}_H \quad \|H - H_k\| \\ & \text{sujeito a: } H = H^T \\ & \quad \quad \quad H y_k = s_k \end{aligned} \quad (3.32)$$

Novamente utiliza-se a norma de Frobenius com peso e a matriz W satisfazendo agora $W s_k = y_k$. Novamente usaremos $W = \bar{G}_k$. A única solução H_{k+1} de (3.32) é dada

por:

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T, \quad (3.33)$$

em que:

$$\rho_k = \frac{1}{y_k^T s_k}. \quad (3.34)$$

Para que possamos definir a fórmula BFGS, precisamos apenas definir a aproximação inicial H_0 . Como nem toda aproximação funciona bem em todos os casos, podemos utilizar algumas alternativas, como por exemplo, definir H_0 como a inversa da aproximação por diferenças finitas da Hessiana de f em x_0 ou utilizar a matriz identidade para refletir o escalamento das variáveis. E uma vantagem do método BFGS em relação ao método de Newton, é o uso apenas das primeiras derivadas.

Podemos derivar uma versão do algoritmo BFGS que funciona com a aproximação da Hessiana B_k em vez de $(B_k)^{-1}$ simplesmente aplicando a fórmula de Sherman-Morrison-Woodbury em (3.33) para obter:

$$B_{k+1} = B_k - \frac{B_k s_k^T s_k B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}. \quad (3.35)$$

Modificando a atualização de H_k dada por B_{k+1} , temos o seguinte algoritmo para o método BFGS:

Algoritmo 3: Método BFGS.

Entrada: Sejam $x_0 \in \mathbb{R}^n$ arbitrário e $H_0 \in \mathbb{R}^{n \times n}$ uma matriz simétrica e definida positiva.

Saída: Ponto de mínimo;

enquanto $\|\nabla f(x_k)\| \neq 0$, **faça**

$$d_k = -H_k \nabla f(x_k);$$

Determinar α_k através de uma busca linear e definir $x_{k+1} = x_k + \alpha_k d_k$;

Definir $s_k = x_{k+1} - x_k$, $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ e calcular

$$H_{k+1} = H_k - \frac{H_k s_k^T s_k H_k}{s_k^T H_k s_k} + \frac{y_k y_k^T}{y_k^T s_k};$$

$k \leftarrow k + 1$;

fim

Retorna *Ponto de mínimo*

3.1.2 Método da Função Penalidade (BAZARAA; SHERALI; SHETTY, 1979)

O método da função penalidade que será descrito a seguir, é utilizado neste trabalho para tratar as restrições de igualdade do problema de despacho econômico.

Um problema de otimização no qual há apenas restrições de igualdade pode ser escrito, genericamente como:

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } g(x) = 0 \end{aligned} \tag{3.36}$$

em que:

$g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, representa o conjunto das restrições de igualdade do problema;
 $x \in \Omega \subseteq \mathbb{R}^n$, sendo Ω um conjunto aberto.

Os métodos que utilizam a função penalidade, transformam um problema restrito em um problema irrestrito. As restrições são adicionadas à função objetivo através de um parâmetro de penalidade de modo a penalizar qualquer violação das restrições. A fim de ilustrarmos o método, considere o problema restrito (3.36) com restrições de igualdade. Suponha que este problema seja substituído pelo seguinte problema irrestrito, onde $\mu > 0$ é um número muito grande:

$$\begin{aligned} & \text{minimizar } f(x) + \mu g^2(x) \\ & \text{sujeito a: } x \in \mathbb{R}^n \end{aligned} \tag{3.37}$$

Nota-se que uma solução ótima para o problema descrito acima deve ter $g^2(x)$ um valor bem próximo de zero, assim, apenas $f(x)$ estará sendo minimizada com um erro muito pequeno. Caso $g^2(x)$ não tenha um valor próximo de zero, será aplicada uma grande penalidade $\mu g^2(x)$.

Ao problema (3.36) também podem ser inseridas restrições de desigualdade e sua formulação é dada por:

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } h(x) \leq 0 \end{aligned} \tag{3.38}$$

em que:

$h : \mathbb{R}^n \rightarrow \mathbb{R}^p$, representa o conjunto das restrições de desigualdade do problema;
 $x \in \Omega \subseteq \mathbb{R}^n$, sendo Ω um conjunto aberto.

Agora considere o problema restrito (3.38) com restrições de desigualdade $h(x) \leq 0$. Neste caso, a forma $f(x) + \mu g^2(x)$ não é apropriada, já que conseqüentemente para $h(x) < 0$ ou $h(x) > 0$ também estará ocorrendo a penalização, e o intuito é penalizar apenas se o ponto x não for viável, ou seja, para $h(x) > 0$. Um modelo de problema irrestrito adequado é descrito da seguinte forma:

$$\begin{aligned} & \text{minimizar } f(x) + \mu \max \{0, h(x)\} \\ & \text{sujeito a: } x \in \mathbb{R}^n \end{aligned} \quad (3.39)$$

Isso nos diz que, se $h(x) \leq 0$, então o $\max \{0, h(x)\} = 0$ e a penalidade não ocorre. Contudo, se $h(x) > 0$, então o $\max \{0, h(x)\} > 0$ e o termo de penalidade $\mu h(x)$ é adicionado à função objetivo. Mas nos pontos x onde $h(x) = 0$, a função objetivo anterior pode não ser diferenciável, e se a diferenciabilidade é uma condição necessária para o método de resolução, podemos utilizar um termo de penalidade do tipo $\mu [\max \{0, h(x)\}]^2$.

Portanto, uma função penalidade deve incidir em uma penalidade positiva para pontos não factíveis e nenhuma penalidade para pontos factíveis. Sejam as restrições da forma $h_i(x) \leq 0$ para $i \in \{1, 2, \dots, p\}$ e $g_i(x) = 0$ para $i \in \{1, 2, \dots, m\}$ então a função penalidade α é definida como:

$$\alpha(x) = \sum_{i=1}^p \phi[h_i(x)] + \sum_{i=1}^m \psi[g_i(x)], \quad (3.40)$$

onde ϕ e ψ são funções contínuas satisfazendo

$$\phi(y) = 0 \text{ se } y \leq 0 \text{ e } \phi(y) > 0 \text{ se } y > 0, \quad (3.41)$$

$$\psi(y) = 0 \text{ se } y = 0 \text{ e } \psi(y) > 0 \text{ se } y \neq 0. \quad (3.42)$$

As funções ϕ e ψ podem ser definidas das seguintes formas:

$$\phi(y) = [\max \{0, y\}]^q, \quad (3.43)$$

$$\psi(y) = |y|^q, \quad (3.44)$$

sendo q um número inteiro positivo. A função α é definida comumente como se segue:

$$\alpha(x) = \sum_{i=1}^p [\max \{0, h_i(x)\}]^q + \sum_{i=1}^m |g_i(x)|^q, \quad (3.45)$$

e tem-se então a função auxiliar P definida como:

$$P(x) = f(x) + \mu\alpha(x). \quad (3.46)$$

Os passos do algoritmo da função penalidade são apresentados a seguir.

Algoritmo 4: Função Penalidade.

Entrada: Estabelecer o erro de convergência ($\epsilon > 0$), o ponto inicial x_0 , o parâmetro de penalidade $\mu_0 > 0$ e o fator de incremento da penalidade $\beta > 1$;

$k \leftarrow 0$;

Saída: Solução ótima;

enquanto $\mu_k\alpha(x_{k+1}) > \epsilon$, **faça**

 Resolva o seguinte problema utilizando um método de minimização irrestrita para μ_k fixo, obtendo x_{k+1} :

 minimizar $f(x) + \mu_k\alpha(x)$

$x \in \mathbb{R}^n$

$\mu_{k+1} = \beta\mu_k$;

$k \leftarrow k + 1$;

fim

Retorna x_k

O capítulo seguinte apresenta as características de métodos bio-inspirados e não bio-inspirados, destacando o método ED utilizado como base neste trabalho.

4 COMPUTAÇÃO EVOLUTIVA

Neste capítulo, abordaremos as características de métodos inspirados em mecanismos de adaptação dos seres vivos, conforme observados na natureza, eficazes para resolver diversos problemas de otimização.

4.1 Métodos bio-inspirados

Os métodos bio-inspirados são métodos que baseiam-se na evolução das espécies. A evolução biológica realiza uma pesquisa num espaço de busca de grande dimensão e complexidade, constituído por todas as possíveis combinações genéticas que podem ser geradas (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). No processo evolutivo, é obtido um conjunto de diferentes vetores de variáveis de decisão, que seria o análogo computacional de uma “população” de seres vivos. Seguindo a analogia, esses vetores de variáveis de decisão seriam o mesmo que “indivíduos” ou cromossomos. Os genes são os constituintes básicos dos cromossomos, podendo ser encarados como cada uma das características elementares de uma solução. Estão localizados em uma determinada posição e cada um deles pode tomar um valor de um conjunto de possibilidades, denominadas alelos (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Cada uma das iterações do processo de otimização é designada de geração e a sequência das várias populações reflete a evolução ao longo do tempo.

Conforme as hipóteses apresentadas por Darwin, o processo de evolução natural recorre a dois mecanismos básicos: a seleção e a reprodução com variação (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Fazendo uma analogia com o mecanismo de seleção natural darwiniana, os indivíduos pertencentes a uma população são selecionados para passar para a geração seguinte de acordo com o seu valor de função de aptidão, denominada de função *fitness*, conduzindo assim à convergência para valores sempre melhores. A seleção garante que os indivíduos mais aptos têm maior probabilidade de sobreviver e originar novos descendentes (filhos), propagando assim as suas características ao longo das gerações. A variação genética assegura que os descendentes gerados não sejam uma cópia exata dos pais, garantindo uma variabilidade entre os indivíduos. Essas variações representam novos pontos a serem analisados e possuem algum tipo de semelhança com os pontos anteriores. Assim, mais pontos são gerados próximos aos melhores pontos anteriores e mais distantes dos piores pontos, criando uma “memória” dentro do mecanismo de otimização (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Dessa maneira, ao longo das gerações, a população de indivíduos evolui de forma progressiva.

A formulação, no início do século XX, da teoria genética da hereditariedade permitiu integrar o conceito de gene no processo evolutivo, tornando mais claro de que forma é que as alterações são introduzidas na população. A variabilidade das espécies passa a ser explicada através da aplicação de um conjunto de operadores genéticos, como o operador de mutação (ou recombinação), o operador de *crossover* (ou cruzamento) e o operador de seleção (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). O operador de *crossover* efetua a troca de material genético originando um ou mais descendentes. Já o operador de mutação é responsável por fazer alterações aleatórias nos genes dos indivíduos. O operador de seleção atua na conservação ou exclusão dos indivíduos nas gerações futuras, permitindo manter ou não a informação descoberta anteriormente. Através dessas transformações efetuadas nos indivíduos selecionados, é promovida a exploração de novas regiões do espaço de busca. Assumindo que este possui algum tipo de regularidade, as referidas transformações poderão conduzir à descoberta de novas regiões, onde se encontrem soluções de qualidade superior às anteriores.

Adotando a inspiração natural, os métodos bio-inspirados processam conjuntos de elementos do espaço de busca. Estes conjuntos são evoluídos ao longo de sucessivas gerações. O objetivo é encontrar uma solução melhor que todas as outras ao final das gerações, idealmente chamada de solução ótima. No início são escolhidas aleatoriamente várias soluções. A partir deste conjunto inicial e de acordo com os operadores de mutação e *crossover*, as evoluções na população de soluções são efetuadas selecionando os indivíduos mais aptos de uma determinada geração para servirem de progenitores das soluções que irão aparecer na geração seguinte. Também são aplicados os operadores genéticos, que atuam sobre os elementos selecionados originando as novas soluções. A seleção dos indivíduos é realizada de acordo com o valor de suas aptidões, calculado pela função *fitness*. A função *fitness* associa a cada solução uma medida de qualidade, representando a sua capacidade para resolver determinado problema.

Devido ao mecanismo de seleção, a qualidade média dos elementos que constituem a população tem tendência para aumentar ao longo do tempo. Os operadores genéticos são os responsáveis pela obtenção de novas soluções, ao mesmo tempo que tentam garantir que o processo mantenha um nível adequado de diversidade. À medida que o número de gerações aumenta, um método bio-inspirado converge gradualmente para regiões do espaço de busca onde se encontram soluções promissoras. A otimização termina quando um determinado critério de parada preestabelecido é atingido. Os critérios de parada mais comuns são: número máximo de gerações, descoberta de uma solução com qualidade pretendida ou inexistência de melhoria durante um determinado período de tempo.

Os princípios básicos da evolução biológica serviram de inspiração para o desenvolvimento de algoritmos de pesquisa e adaptação em ambientes computacionais, tendo em vista aplicações, tanto no campo da engenharia, como no da biologia (GASPAR-CUNHA;

TAKAHASHI; ANTUNES, 2012). Os AGs, estratégias evolutivas (EEs) (RECHENBERG, 1978) e colônias de formigas (do inglês *ant colony optimization*, ou ACO) (DORIGO; DI-CARO; GAMBARDELLA, 1999) são exemplos de métodos inspirados nos princípios da seleção natural e da genética. Esses métodos realizam o processamento simultâneo de uma população de potenciais soluções para o problema. Esta particularidade permite que diferentes áreas do espaço de busca (potenciais soluções para o problema) sejam analisadas (EIBEN; SMITH, 2003), reduzindo a probabilidade de convergência prematura para um ótimo local. Possuem componentes probabilísticos e não recorrem a informação específica sobre o problema que estão a resolver, sendo que a única informação de que necessitam é o resultado da aplicação da função de aptidão. Estes dois fatores, combinados com a existência de uma população de soluções, tornam estes métodos particularmente versáteis e robustos.

4.2 Métodos não bio-inspirados

Embora a inspiração biológica ainda seja uma fonte de idéias para as pesquisas na área da Computação Evolutiva, já se tornou frequente o uso de mecanismos cuja justificativa é essencialmente matemática e computacional. O crescente entendimento dos mecanismos que fundamentam os algoritmos da Computação Evolutiva tem permitido, de certa forma, um afastamento dos novos algoritmos que vêm sendo desenvolvidos em relação à estrita inspiração biológica. Desde meados dos anos 90, diversos algoritmos foram desenvolvidos sem que houvesse a preocupação de que de alguma forma fosse imitado algum mecanismo de adaptação observado na natureza. Exemplos desses algoritmos são o SA, TS e ED. Daremos destaque ao algoritmo de ED que será utilizado neste trabalho.

4.2.1 Evolução diferencial

Encontra-se na literatura a primeira publicação sobre o algoritmo de ED no ano de 1995, em um relatório técnico de Rainer Storn e Kenneth Price (STORN; PRICE, 1995). Trata-se de um algoritmo de otimização simples e eficiente para resolver problemas de otimização não linear com variáveis contínuas (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Após o algoritmo ser apresentado no *International Contest on Evolutionary Optimization da IEEE International Conference on Evolutionary Computation* (IEEE ICEC) em 1996 e 1997, o algoritmo ganhou destaque pelo seu excelente desempenho. Neste concurso, na edição de 1996, a ED ficou em terceiro lugar (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012).

Na edição de 1997, com um novo conjunto de funções de teste, a ED apresentou o melhor desempenho entre os algoritmos classificados (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Em dezembro de 1997, Storn e Price publicaram um artigo no *Journal*

of *Global Optimization* apresentando testes experimentais mais amplos e resultados que ilustravam a robustez do algoritmo. Em 1999, os pesquisadores David Corne, Marco Dorigo e Fred Glover incluíram uma seção de capítulo no livro *New Ideas in Optimization* sobre o algoritmo de evolução diferencial (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012).

Com o passar dos anos, esse algoritmo foi se mostrando versátil e eficaz em muitas aplicações práticas. Exemplos dessas aplicações são encontrados no trabalho de Storn (1999) com o projeto de filtros digitais; em Reddy e Kumar (2007) com a otimização de sistemas de reservatório de água, e em Ruzek e Kvasnicka (2001) com a localização do foco sísmico de terremotos. No treinamento de redes neurais a ED também tem se mostrado eficaz (ILONEN; KAMARAINEN; LAMPINEN, 2003), assim como na solução de problemas inversos (MICHALSKI, 2001) e para o projeto de dispositivos de engenharia elétrica (KIM et al., 2007). Melhorias do algoritmo têm surgido ao longo do tempo, como no trabalho de Gong e Cai (2008), em que os autores propõem técnicas para o tratamento de restrições; e em Batista, Guimarães e Ramirez (2009) com versões para a resolução de problemas de otimização multiobjetivo.

A forma como o operador de mutação gera perturbações nos valores dos indivíduos da população, também conhecida como processo de mutação, se sustenta em argumentos matemáticos e heurísticos que indicam sua adequabilidade para a otimização de funções e não exatamente em argumentos derivados de metáforas da natureza. Contudo, a ED segue uma linha histórica de algoritmos e métodos que evoluem uma população de soluções candidatas segundo operadores heurísticos inspirados em mecanismos bastante gerais de adaptação natural, e por essa razão, a ED é classificada como uma variante dos algoritmos evolutivos (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012).

Iremos apresentar uma visão geral sobre o algoritmo de ED e seu mecanismo de busca, sustentado no operador de mutação diferencial, que dá nome ao algoritmo. Embora o algoritmo de ED seja classificado como um algoritmo evolutivo, e se enquadre em um esquema geral de um algoritmo evolutivo, a mutação diferencial não tem base ou inspiração em nenhum processo natural (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012).

Cada solução factível encontrada pelo algoritmo é avaliada e aplicada ao processo de evolução, que se baseia em mutar os indivíduos com uma certa probabilidade, cruzar o indivíduo mutante com o indivíduo atual gerando um novo indivíduo, e selecionar os melhores indivíduos. A seguir é descrito o método ED.

Inicialmente, é gerada uma população inicial de forma aleatória com distribuição uniforme representada por $P_g = \{P_{g,i}/i = 1, \dots, NP\}$, em que NP é o tamanho da população, g é o índice da geração corrente e i é o índice do indivíduo na população. Cada indivíduo da população é representado pelo vetor (GASPAR-CUNHA; TAKAHASHI;

ANTUNES, 2012):

$$P_{g,i} = \begin{bmatrix} P_{g,i,1} \\ P_{g,i,2} \\ \vdots \\ P_{g,i,j} \\ \vdots \\ P_{g,i,D} \end{bmatrix}, \quad (4.1)$$

em que D é o número de variáveis do problema. Considerando um problema com dimensão D , cada solução $P_{g,i}$ é um vetor D -dimensional onde cada posição é gerada aleatoriamente a partir da expressão:

$$P_{g,i,j} = P_{g,i,j}^{min} + rand[0, 1](P_{g,i,j}^{max} - P_{g,i,j}^{min}), \quad (4.2)$$

em que os valores $P_{g,i,j}^{min}$ e $P_{g,i,j}^{max}$ são os limites inferiores e superiores, respectivamente, da j -ésima variável e $rand[0, 1]$ gera um valor aleatório no intervalo $[0,1]$ (STORN; PRICE, 1997). Após essa população ser gerada, para cada indivíduo é avaliada a função *fitness*. Se o problema for de minimização, o indivíduo que possuir o menor valor de função *fitness* será o melhor, o mesmo acontece se o problema for de maximização, onde o melhor indivíduo será aquele que fornece o maior valor de função *fitness*. Realizada esta avaliação, inicia-se o processo evolutivo do método.

Um indivíduo $P_{g,i}$ é escolhido de forma aleatória para ser substituído e outros três diferentes indivíduos $P_{g,r1}$, $P_{g,r2}$ e $P_{g,r3}$ são selecionados também de forma aleatória. Com isso, é iniciada a próxima etapa, onde é feita a modificação de um vetor através da diferença ponderada entre dois indivíduos da população. Este processo recebe o nome de mutação e no método ED há variações desse processo como será apresentado na Tabela 4.1. Entre os indivíduos $P_{g,r1}$, $P_{g,r2}$ e $P_{g,r3}$ serão escolhidos aleatoriamente dois, e realizada a diferença entre esses dois indivíduos, criando assim um vetor-diferença. Este vetor-diferença é multiplicado por um fator de mutação F e somado ao terceiro indivíduo que não foi escolhido inicialmente. Assim, tem-se o vetor doador $v_{g,i}$, que é o resultado de uma perturbação no valor de algum indivíduo da população.

Para cada indivíduo $P_{g,i}$ da população é gerado um vetor mutante $v_{g,i}$ (ou vetor doador) definido na Tabela 4.1. Logo após finalizar o processo de mutação, acontece o *crossover*, também chamado de cruzamento, que segundo Storn e Price (1997) é uma forma de aumentar a diversidade dos parâmetros dos vetores com os valores perturbados, sendo que esse cruzamento ocorre de acordo com uma probabilidade. Os indivíduos $P_{g,i}$

Tabela 4.1 – Estratégias de mutação.

Estratégia	Expressão
$ED/rand/1/bin$	$v_{g,i} = P_{g,r1} + F(P_{g,r2} - P_{g,r3})$ (1)
$ED/best/1/bin$	$v_{g,i} = P_{g,best} + F(P_{g,r1} - P_{g,r2})$ (2)
$ED/rand/2/bin$	$v_{g,i} = P_{g,r1} + F(P_{g,r2} - P_{g,r3}) + F(P_{g,r4} - P_{g,r5})$ (3)
$ED/best/2/bin$	$v_{g,i} = P_{g,best} + F(P_{g,r1} + P_{g,r2} - P_{g,r3} - P_{g,r4})$ (4)
$ED/rand - to best/1/bin$	$v_{g,i} = P_{g,i} + \lambda(P_{g,best} - P_{g,i}) + F(P_{g,r2} - P_{g,r3})$ (5)

Fonte: Adaptada de [Storn e Price \(1995\)](#), [Storn \(1996\)](#) e [Storn e Price \(1997\)](#).

da população são recombinaados com os indivíduos da população mutante, produzindo a descendência ou população de soluções testes $u_{g,i}$.

É possível implementar a ED utilizando algumas estratégias que variam de acordo com o tipo de indivíduo a ser modificado na formação do vetor doador, o número de indivíduos considerados para a perturbação e o tipo de crossover a ser utilizado. Para definir a mutação e o cruzamento utiliza-se a notação $ED/x/y/z$, onde x caracteriza se elegeremos um vetor aleatório (*rand*) para mutar ou o vetor que produz a melhor solução (*best*), y indica o número de vetores-diferenças empregados e z mostra se o tipo de cruzamento a ser realizado será binomial ou exponencial ([STORN; PRICE, 1997](#)).

Após o cruzamento, é gerado um novo indivíduo $u_{g,i}$, denominado vetor teste, e também é calculado o seu valor de função *fitness*. É realizada a seleção gulosa, que consiste na seleção dos indivíduos mais aptos para serem inseridos na nova população. Então, compara-se o valor de função *fitness* do indivíduo atual $P_{g,i}$ com o valor de função *fitness* do novo indivíduo $u_{g,i}$. Se o valor de função *fitness* do indivíduo atual $P_{g,i}$ for melhor que o valor de função *fitness* do novo indivíduo $u_{g,i}$, $P_{g,i}$ é inserido na nova população e $u_{g,i}$ é descartado. Caso contrário $u_{g,i}$ é inserido na nova população e $P_{g,i}$ é descartado ([STORN; PRICE, 1997](#)). A cada geração do algoritmo uma nova população é gerada, substituindo a anterior. No final, a ED deve retornar a melhor solução encontrada. O critério de parada do método é dado pelo número máximo de gerações ou qualidade das soluções.

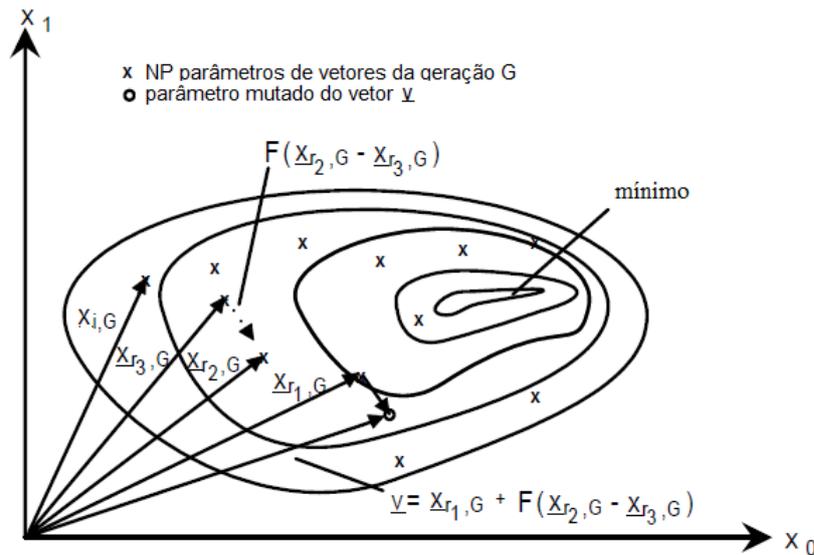
4.2.2 Mutação

Como dito anteriormente, existem diversas estratégias para definir a mutação de um vetor. A Tabela 4.1 apresenta algumas dessas estratégias de mutação da ED.

Chamada de $ED/rand/1$, a primeira estratégia de mutação da ED citada na Tabela 4.1, é a estratégia mais utilizada ([DE-OLIVEIRA; NASCIMENTO-JR; SAAVEDRA, 2008](#)). É caracterizada por *rand*, assim determina um vetor escolhido de forma aleatória que será utilizado para mutar $P_{g,i}$, e 1 determina a quantidade de vetores-diferença utilizados para a perturbação ([STORN, 1996](#)). Essa mutação é realizada de acordo com a expressão (1). Com essa mutação é gerado um novo vetor $v_{g+1,i}$ a partir da diferença ponderada entre dois outros vetores da população. Esse novo vetor $v_{g+1,i}$ é denominado vetor mutante, que

no processo de cruzamento irá resultar no vetor teste $u_{g+1,i}$. A constante F é um valor real maior que zero, que determina o tamanho do passo a ser dado na direção definida pelo vetor diferença e os valores $r1, r2$ e $r3 \in \{1, 2, \dots, NP\}$. A Figura 4.1 apresenta um exemplo bidimensional que ilustra os vetores $P_{g,i}, P_{g,r1}, P_{g,r2}$ e $P_{g,r3}$, os quais são utilizados para calcular o vetor mutante $v_{g+1,i}$ (STORN; PRICE, 1995).

Figura 4.1 – Exemplo bidimensional das curvas de nível de uma função objetivo e o processo para gerar o vetor mutante a partir de um vetor escolhido aleatoriamente.



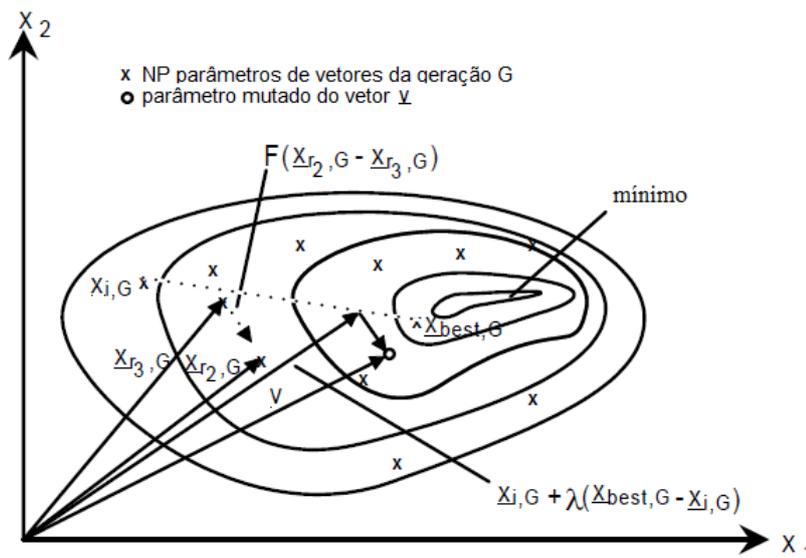
Fonte: Adaptado de (STORN; PRICE, 1995).

A segunda estratégia é denominada *ED/best/1*, onde *best* indica que o vetor escolhido com intuito de ser utilizado para mutar o vetor $P_{g,i}$, será o vetor que possui melhor solução na geração g , e 1 determina a quantidade de vetores-diferença utilizados para a perturbação. Essa mutação é realizada de acordo com a expressão (2). Com essa mutação é gerado um novo vetor $v_{g+1,i}$ a partir da diferença ponderada entre dois outros vetores da população. Esse novo vetor $v_{g+1,i}$ é denominado vetor mutante, e no processo de cruzamento irá resultar no vetor teste $u_{g+1,i}$. A constante F é um valor real maior que zero, que determina o tamanho do passo a ser dado na direção definida pelo vetor diferença e os valores $r1$ e $r2 \in \{1, 2, \dots, NP\}$ (STORN, 1996).

A terceira estratégia de mutação apresentada é denominada *ED/rand/2*. Assim como a primeira estratégia, também será escolhido um vetor aleatório para mutar $P_{g,i}$, porém são utilizadas duas diferenças ponderadas para realizar a perturbação. Essa mutação é realizada de acordo com a expressão (3). Com essa mutação é gerado um novo vetor $v_{g+1,i}$ a partir da diferença ponderada entre dois outros pares de vetores da população. A constante F é um valor real maior que zero, que determina o tamanho do passo a ser dado na direção definida pelos vetores diferença e os valores $r1, r2, r3, r4$ e $r5 \in \{1, 2, \dots, NP\}$ (STORN; PRICE, 1997).

A quarta estratégia de mutação apresentada é denominada *ED/best/2*, que assim como a segunda estratégia, também escolherá a melhor solução para mutar o vetor $P_{g,i}$, porém utilizará duas diferenças ponderadas para realizar a perturbação. Essa mutação é realizada de acordo com a expressão (4). A constante F é um valor real maior que zero, que determina o tamanho do passo a ser dado na direção definida pelo vetor diferença e os valores $r1, r2, r3$ e $r4 \in \{1, 2, \dots, NP\}$ (STORN, 1996).

Figura 4.2 – Exemplo bidimensional das curvas de nível de uma função objetivo e o processo para gerar o vetor mutante a partir de um vetor escolhido aleatoriamente.



Fonte: Adaptado de Storn e Price (1995).

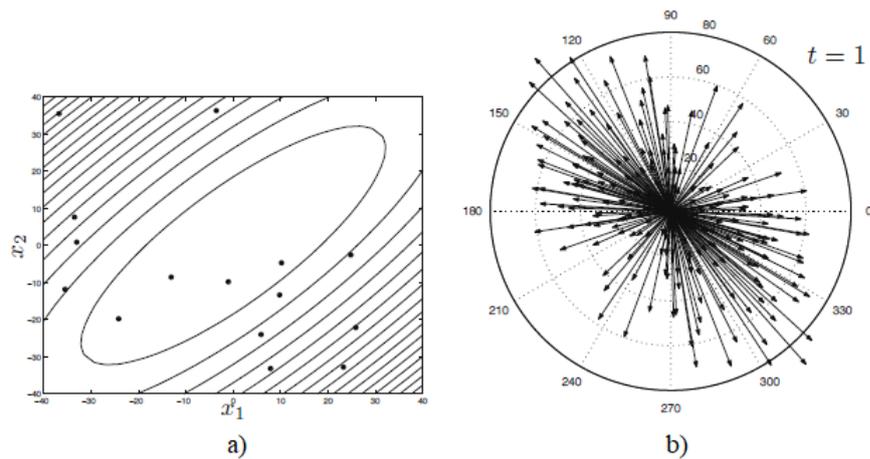
A quinta e última estratégia de mutação apresentada é denominada *ED/rand – to best/1* onde a perturbação é realizada entre o melhor membro da população $P_{g,best}$ e o indivíduo atual $P_{g,i}$ e também entre indivíduos da população escolhidos de forma aleatória. Essa mutação é realizada de acordo com a expressão (5). A constante F é um valor real maior que zero, que determina o tamanho do passo a ser dado na direção definida pelo vetor diferença e os valores $r2$ e $r3 \in \{1, 2, \dots, NP\}$ (STORN, 1996). A variável λ é utilizada para incorporar o melhor indivíduo da população atual $P_{g,i}$ (STORN; PRICE, 1995), e normalmente é definido $\lambda = F$ para reduzir o número de variáveis de controle (STORN, 1996). A Figura 4.2 apresenta um exemplo bidimensional que ilustra os vetores $P_{g,i}$, $P_{g,best}$, $P_{g,r2}$ e $P_{g,r3}$ os quais desempenham um papel na geração do vetor mutante $v_{g+1,i}$ (STORN; PRICE, 1995).

4.2.3 Comportamento da mutação diferencial para otimizar uma função convexa

No algoritmo de ED, a distribuição dos vetores-diferença, que serão utilizados na operação de mutação para perturbar indivíduos da população, depende da distribuição

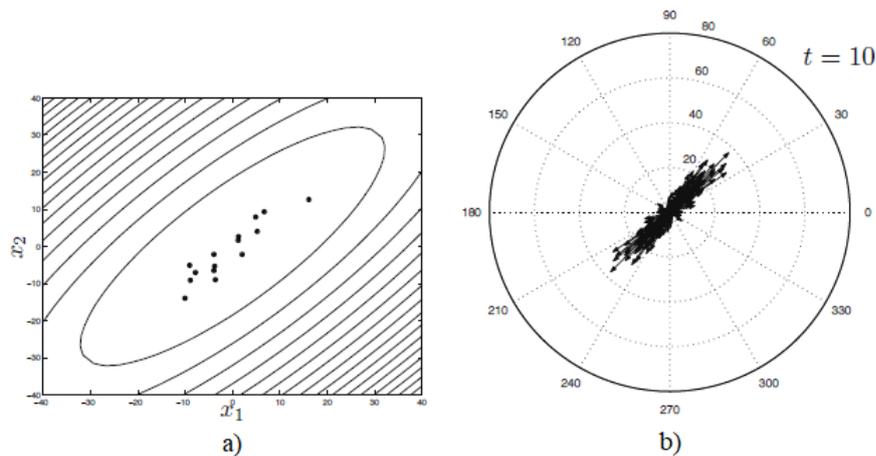
especial dos indivíduos da população do problema. À medida que a população se distribui de acordo com as curvas de nível da função objetivo, a distribuição dos vetores-diferença também se ajusta a essas curvas (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). As Figuras 4.3 a 4.5 mostram essas distribuições para uma função objetivo quadrática.

Figura 4.3 – a) Distribuição espacial da população na geração $t = 1$. b) Distribuição dos vetores-diferença na geração $t = 1$.



Fonte: Gaspar-cunha, Takahashi e Antunes (2012).

Figura 4.4 – a) Distribuição espacial da população na geração $t = 10$. b) Distribuição dos vetores-diferença na geração $t = 10$.

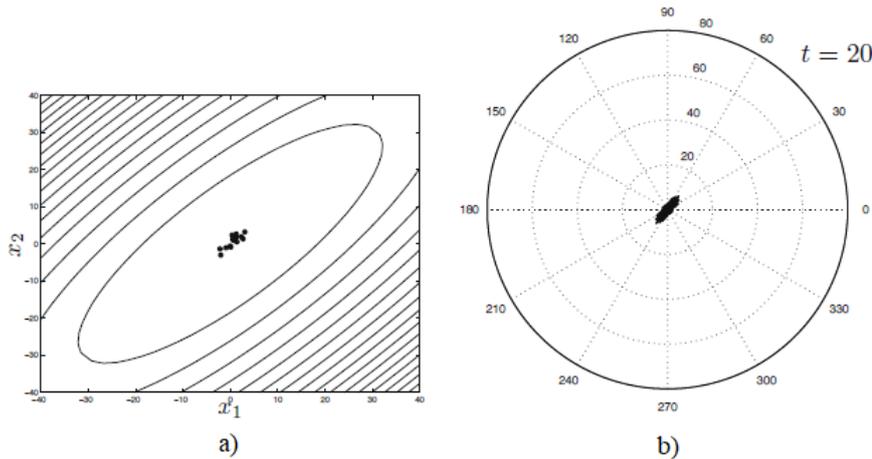


Fonte: Gaspar-cunha, Takahashi e Antunes (2012).

Nas figuras citadas, nota-se que as curvas de nível da função representam elipses. Na Figura 4.3, como a população ainda não possui nenhuma informação sobre a função objetivo em questão, vemos que os vetores-diferença possuem tamanhos variados, além de estarem apontados para várias direções. Durante o processo de evolução do algoritmo, muitas soluções irão apresentar resultados piores que outras e serão descartadas, enquanto as soluções que apresentarão resultados melhores irão substituir indivíduos na população. Isso fará com que a distribuição espacial da população se altere, e que com o passar das

gerações se alinhe com o contorno da função objetivo (GASPAR-CUNHA; TAKAHASHI; ANTUNES, 2012). Podemos observar essa tendência ilustrada pelas figuras 4.4 e 4.5, onde vemos que da décima para vigésima geração, a distribuição espacial se assemelha ao contorno da função objetivo.

Figura 4.5 – a) Distribuição espacial da população na geração $t = 20$. b) Distribuição dos vetores-diferença na geração $t = 20$.



Fonte: Gaspar-cunha, Takahashi e Antunes (2012).

As Figuras 4.3 a 4.5 exemplificam o comportamento geral do algoritmo de ED em funções convexas. Veremos na próxima seção esse mesmo comportamento, mas para uma função multimodal.

4.2.4 Função multimodal

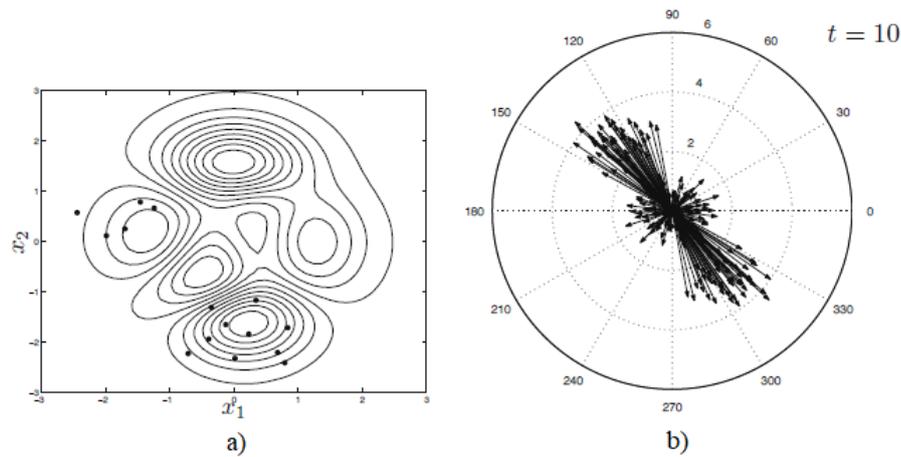
Tendo em vista a variedade de problemas que podemos encontrar, é comum que as funções objetivo desses problemas possam apresentar diversos mínimos locais. Uma função que contém essa característica, é chamada de função multimodal. Neste caso, a população avizinha-se em torno dos mínimos locais, e durante o processo iterativo tende a se concentrar no mínimo local com melhor solução. Nas figuras 4.6 e 4.7 têm-se um exemplo da distribuição espacial dos vetores-diferença para uma função objetivo multimodal.

Podemos observar na Figura 4.6 que nas primeiras gerações os vetores-diferença possuem tamanhos variados, além de estarem apontados para várias direções, consequência da aleatoriedade dos indivíduos iniciais. Conforme o processo iterativo acontece, a população vai se concentrando na região que possui melhores soluções para função objetivo, enquanto os indivíduos que estavam nas outras regiões são descartados, sendo substituídos por indivíduos com soluções melhores (Figura 4.7). Os vetores-diferença começam a ter amplitudes menores e concentram-se na mesma região de atração.

As figuras 4.6 e 4.7 demonstram o caso de função multimodal, onde a distribuição espacial da população se acumula em regiões com tendências distintas, causando a geração

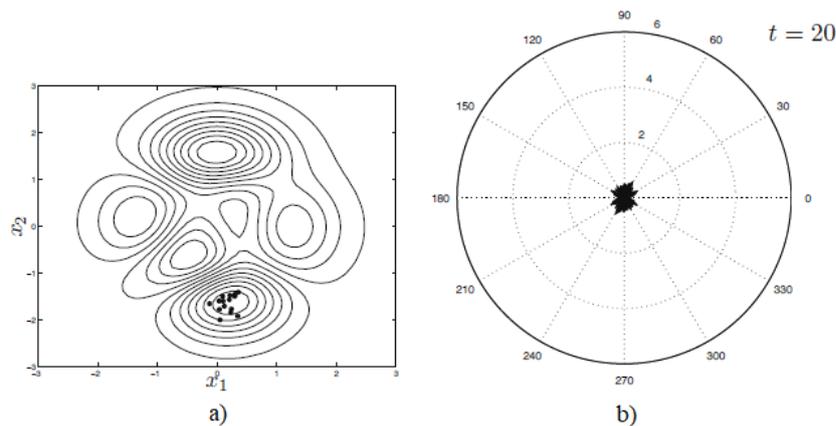
de grupos de vetores-diferença bem definidos. A perturbação causada por alguns grupos de vetores leva a saltos na região de busca em direção a uma única região, ao mesmo tempo que outros grupos de vetores provocam pequenas perturbações nos valores dos indivíduos da população, melhorando a busca local em cada região de atração. Depois de algumas gerações, a população se concentra em uma única região.

Figura 4.6 – a) Distribuição espacial da população na geração $t = 10$. b) Distribuição dos vetores-diferença na geração $t = 10$.



Fonte: Gaspar-cunha, Takahashi e Antunes (2012).

Figura 4.7 – a) Distribuição espacial da população na geração $t = 20$. b) Distribuição dos vetores-diferença na geração $t = 20$.



Fonte: Gaspar-cunha, Takahashi e Antunes (2012).

4.2.5 Crossover

O *crossover*, também chamado de operador de cruzamento, é responsável por realizar a troca de características entre os indivíduos da população e os vetores mutantes, de modo que ao final desse cruzamento é gerado um novo indivíduo modificado, denominado de solução teste $u_{g+1,i}$. Este novo indivíduo herdará as características dos indivíduos cruzados, ou seja, os pais (STORN; PRICE, 1995). Segundo Storn e Price (1997), com

esse cruzamento, aumenta-se a diversidade dos parâmetros dos vetores com os valores perturbados.

No algoritmo de ED, duas formas de cruzamento são possíveis, sendo o cruzamento binomial ou exponencial. Na versão clássica do algoritmo de ED emprega-se a recombinação discreta com probabilidade $CR \in [0, 1]$. A distribuição binomial é dada por (STORN; PRICE, 1997):

$$u_{g+1,i,j} = \begin{cases} v_{g+1,i,j}, & \text{se } randb(j) \leq CR \\ P_{g,i,j}, & \text{se } randb(j) > CR \end{cases} \quad (4.3)$$

onde $randb(j)$ é a amostragem de uma variável aleatória com distribuição uniforme no intervalo $[0, 1]$ e CR é a probabilidade de aceitação de informação da solução mutante.

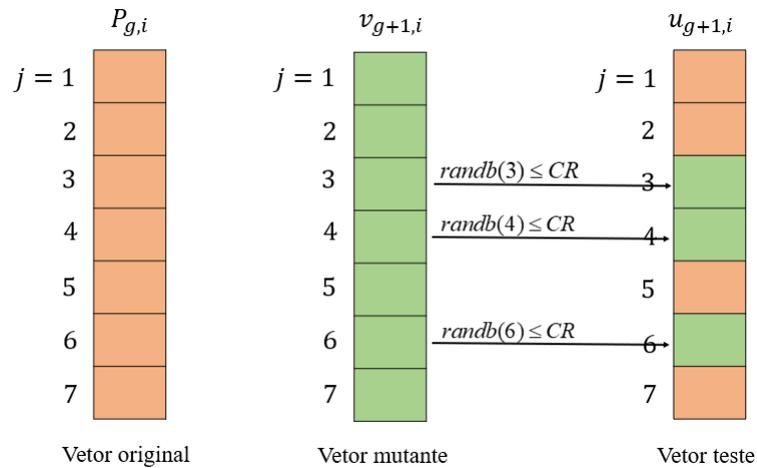
É ilustrado na Figura 4.8 o procedimento do cruzamento binomial em um vetor com dimensão igual a 7. O indivíduo $P_{g,i}$ da população é modificado através do cruzamento com um vetor mutante $v_{g+1,i}$ utilizando uma das estratégias de mutação apresentadas anteriormente. O vetor teste gerado a partir desse cruzamento, é resultado da aplicação de cada posição do vetor original $P_{g,i}$ na equação (4.3). Como vemos na Figura 4.8, as posições 3, 4 e 6 possuem $randb(j) \leq CR$, logo elas são substituídas pelos valores das posições correspondentes do vetor mutante, gerando assim o vetor teste $u_{g+1,i}$.

A segunda forma de cruzamento da ED, é o cruzamento exponencial e é definido pela seguinte equação:

$$u_{g+1,i,j} = \begin{cases} v_{g+1,i,j}, & \text{para } j = \langle n \rangle_D, \langle n+1 \rangle_D, \dots, \langle n+L-1 \rangle_D \\ P_{g,i,j}, & \text{para todos os outros } j \in [0, D-1] \end{cases} \quad (4.4)$$

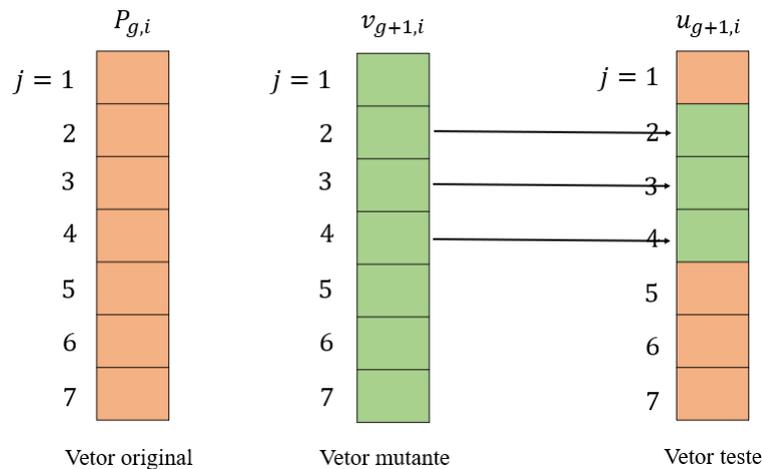
onde $\langle n \rangle_D$ representa o valor inteiro $1 \leq r \leq D$ com módulo D , tal que $n = q \times D + r$, $q \in \mathbb{Z}$ e L é um número inteiro pertencente ao intervalo $[1, D]$ que denota o número de parâmetros que serão substituídos, sendo que $rand()$ deve gerar um número aleatório pertencente ao intervalo $[0, 1]$ e CR a probabilidade de cruzamento. Os valores de n e L são gerados de forma aleatória para cada novo vetor v (STORN; PRICE, 1995). Esta forma de cruzamento fornece uma sequência, em que posições consecutivas de u serão iguais as posições correspondentes de v e os outros elementos de u adquirem os valores do vetor original $P_{g,i}$.

Figura 4.8 – Ilustração do procedimento do cruzamento binomial em um vetor com 7 dimensões.



Fonte: Adaptado de [Storn e Price \(1997\)](#).

Figura 4.9 – Ilustração do procedimento do cruzamento exponencial em um vetor com 7 dimensões.



Fonte: Adaptado de [Storn e Price \(1995\)](#).

É ilustrado na Figura 4.9 o procedimento do cruzamento exponencial em um vetor com 7 dimensões, $n = 2$ e $L = 3$. Para as posições 2, 3 e 4 do vetor $u_{g+1,i}$, os valores de n são $n = 2$, $n = 3$ e $n = 4$, respectivamente, logo elas são substituídas pelos valores das posições correspondentes do vetor mutante, gerando assim o vetor teste $u_{g+1,i}$.

Realizado um desses processos de cruzamento em cada indivíduo da população, é então iniciada a seleção entre o indivíduo original $P_{g,i}$ e o vetor teste $u_{g+1,i}$, ou seja, selecionar qual indivíduo permanecerá para a próxima geração. A seleção nada mais é que o processo de produzir melhores filhos. O valor de *fitness* do vetor teste é calculado e comparado com o valor de *fitness* do vetor original. Esse processo, para um problema de

minimização, pode ser descrito como:

$$P_{g+1,i} = \begin{cases} u_{g+1,i}, & \text{se } f(u_{g+1,i}) \leq f(P_{g,i}) \\ P_{g,i}, & \text{caso contrário} \end{cases} \quad (4.5)$$

Se o valor de *fitness* do vetor original for menor que o valor de *fitness* do vetor teste, o vetor original é selecionado para a próxima geração, caso contrário, o vetor teste substitui o vetor original na geração seguinte. As operações do algoritmo de ED são apresentadas na forma do Algoritmo 5 com *crossover* binomial, e todas as etapas descritas acima são apresentadas no fluxograma da Figura 4.10.

Algoritmo 5: Algoritmo de Evolução Diferencial.

Entrada: NP, F, CR

Saída: Melhor solução encontrada

$g \leftarrow 1$

Gera população inicial $P_{g,i} = P_{g,i}^{min} + (P_{g,i}^{max} - P_{g,i}^{min}) * rand()$

Avalia cada solução da população inicial

Identifica o melhor atual

enquanto *O critério de parada não for satisfeito, faça*

para $i \in \{1, \dots, NP\}$ **faça**

 Seleciona aleatoriamente $r_1, r_2, r_3 \in \{1, \dots, NP\}$ e forma o vetor mutante

$v_{g+1,i}$

se $randb(j) \leq CR$ **então**

$u_{g+1,i,j} = v_{g+1,i,j}$

senão

$u_{g+1,i,j} = P_{g,i,j}$

fim

fim

fim

para $i \in \{1, \dots, NP\}$ **faça**

se $f(u_{g+1,i}) \leq f(P_{g,i})$

$P_{g+1,i} = u_{g+1,i}$

senão

$P_{g+1,i} = P_{g,i}$

fim

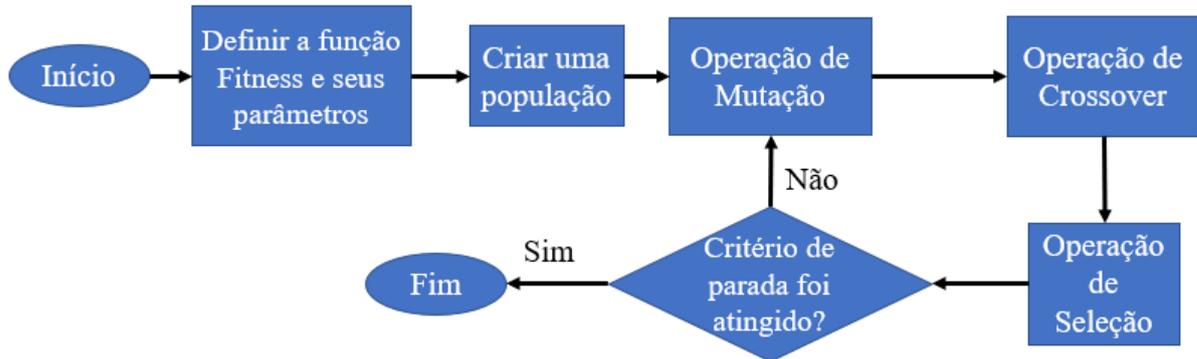
fim

$g \leftarrow g + 1$

fim

Retorna *Melhor solução encontrada*

Figura 4.10 – Fluxograma do algoritmo de Evolução Diferencial.



No próximo capítulo, será apresentada a metodologia de resolução do PDE-PCV proposta neste trabalho.

5 METODOLOGIA DE RESOLUÇÃO PROPOSTA: EVOLUÇÃO DIFERENCIAL HÍBRIDA COM O MÉTODO BFGS

Neste capítulo, será apresentada a metodologia proposta neste trabalho para a resolução do PDE-PCV. Trata-se de uma metodologia híbrida que combina o método ED com o método Quase-Newton BFGS.

5.1 Evolução diferencial híbrida com o método BFGS

Para resolver o PDE-PCV, propõe-se neste trabalho a utilização do método ED-BFGS. O algoritmo de ED é um algoritmo evolucionário, e como dito anteriormente, é baseado na teoria da seleção natural das espécies. O método ED-BFGS é uma modificação proposta ao algoritmo de ED. Nesta combinação entre ED e BFGS, a busca global é realizada pela ED, enquanto o BFGS é projetado para convergir rapidamente para uma região em torno de um mínimo local. O método BFGS explora o espaço de busca com a direção do gradiente e garante uma solução ótima local, conseguindo valores de função *fitness* melhores que a ED.

Inicialmente, é gerada pela ED uma população inicial de acordo com os parâmetros NP , F e CR seguindo as restrições como descritas em [Storn e Price \(1997\)](#). Cada indivíduo desta população é avaliado da mesma forma que na ED clássica, a fim de se saber qual é o indivíduo com melhor valor de função *fitness* e, conseqüentemente, o mais apto a prosseguir para a próxima geração e gerar novos descendentes. Este indivíduo será utilizado como solução inicial no método BFGS, realizando a busca local a partir dele, conforme proposto em [Coelho e Mariani \(2006b\)](#).

Para realizar a avaliação (valor de função *fitness*) do indivíduo pelo método BFGS, é preciso adaptar o problema restrito em um problema irrestrito, já que este método é de otimização irrestrita. Para essa modificação, é utilizada a função penalidade obtendo um problema do tipo (3.37). Após a avaliação do melhor indivíduo pelo método BFGS, é verificado se os limites de geração desta solução respeitam os limitantes de operação do sistema. Se respeitarem, esta solução é substituída na população, caso contrário, serão substituídas as posições desse vetor solução que infringem os limites pela potência mínima ou máxima correspondente, e em seguida, o indivíduo é inserido na população.

Após essas avaliações, o processo evolutivo ocorre a partir desta população inicial do algoritmo de ED, que envolve a seleção de indivíduos aleatórios, sendo os operadores de mutação e cruzamento aplicados sobre estes indivíduos da população. O processo de cruzamento é realizado com uma certa probabilidade, e em seguida, o indivíduo gerado é avaliado, realizando-se a seleção gulosa. Após isso, é realizada a atualização da população e do melhor indivíduo dessa população. Ao final das gerações, o algoritmo de ED-BFGS deve retornar a melhor solução encontrada.

A estratégia de mutação utilizada no algoritmo de ED e ED-BFGS é do tipo *ED/rand/1/bin* e é dada pela seguinte expressão:

$$v_{g+1,i} = P_{g,r1} + F(P_{g,r2} - P_{g,r3}), \quad (5.1)$$

onde $v_{g+1,i}$ é denominado vetor mutante; F é uma constante real maior que zero; $P_{g,r1}$, $P_{g,r2}$ e $P_{g,r3}$ são indivíduos escolhidos de forma aleatória da população.

A forma de cruzamento adotada em ambos os algoritmos é do tipo binomial e é dada por:

$$u_{g+1,i,j} = \begin{cases} v_{g+1,i,j}, & \text{se } randb(j) \leq CR \\ P_{g,i,j}, & \text{se } randb(j) > CR \end{cases} \quad (5.2)$$

onde $u_{g+1,i,j}$ é a j -ésima posição da solução teste; $v_{g+1,i,j}$ é a j -ésima posição do vetor mutante; $randb(j)$ é a amostragem de uma variável aleatória com distribuição uniforme no intervalo $[0, 1]$ e CR é a probabilidade de aceitação de informação da solução mutante.

O Algoritmo 6 mostra o algoritmo de ED-BFGS e todas as etapas descritas acima são apresentadas no fluxograma da Figura 5.1.

Algoritmo 6: Algoritmo de ED-BFGS.

Entrada: NP, F, CR

Saída: Melhor solução encontrada

$g \leftarrow 1$

Gera população inicial $P_{g,i} = P_{g,i}^{min} + (P_{g,i}^{max} - P_{g,i}^{min}) * rand()$

Avalia cada solução da população inicial

Identifica o melhor indivíduo

enquanto *O critério de parada não for satisfeito*, **faça**

BFGS recebe o melhor indivíduo da ED

para $j = \{1, \dots, D\}$ **faça**

se $P_{g,i,j} > P_{g,i,j}^{max}$ então

$$P_{g,i,j} = P_{g,i,j}^{max}$$

senão

$$P_{g,i,j} = P_{g,i,j}$$

fim

se $P_{g,i,j} < P_{g,i,j}^{min}$ então

$$P_{g,i,j} = P_{g,i,j}^{min}$$

senão

$$P_{g,i,j} = P_{g,i,j}$$

fim

fim

Substitui o melhor indivíduo da ED pelo novo indivíduo

para $i = \{1, \dots, NP\}$ faça

Seleciona aleatoriamente $r_1, r_2, r_3 \in \{1, \dots, NP\}$ e forma o vetor mutante $v_{g+1,i}$

para $j = \{1, \dots, D\}$ faça

se $randb(j) \leq CR$ então

$$u_{g+1,i,j} = v_{g+1,i,j}$$

senão

$$u_{g+1,i,j} = P_{g,i,j}$$

fim

fim

fim

para $i = \{1, \dots, NP\}$ faça

se $f(u_{g+1,i}) \leq f(P_{g,i})$

$$P_{g+1,i} = u_{g+1,i}$$

senão

$$P_{g+1,i} = P_{g,i}$$

fim

fim

Atualiza o melhor atual

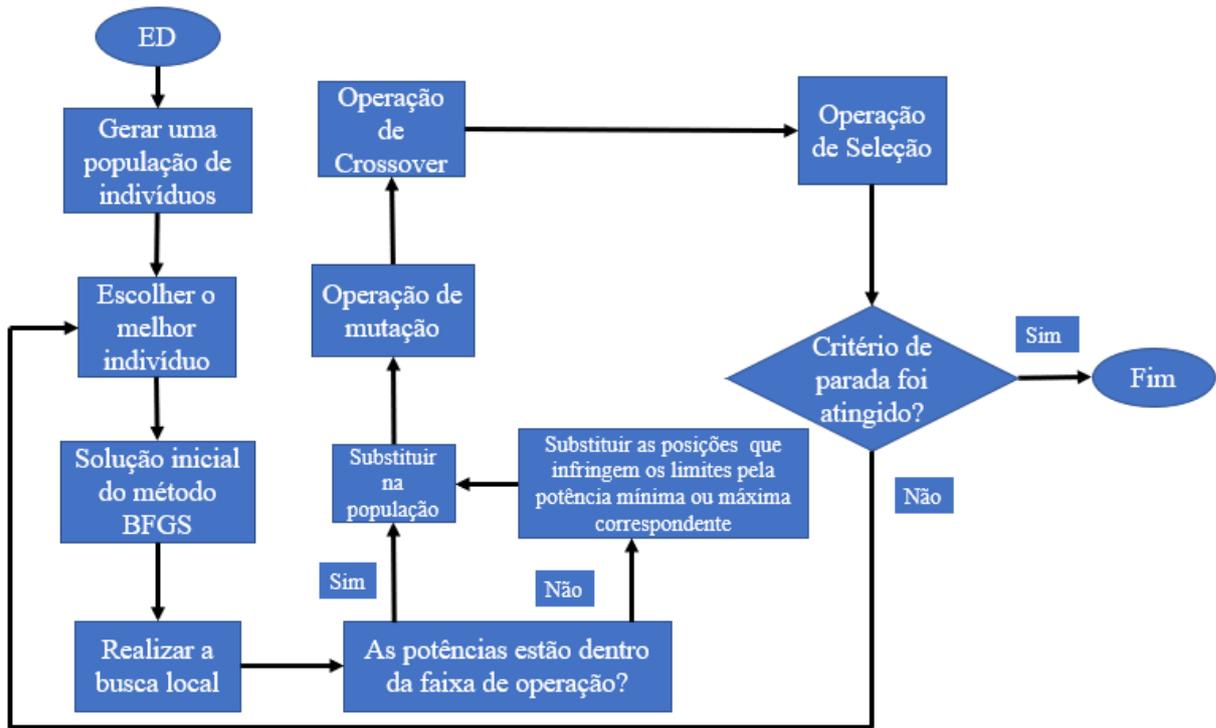
$$g \leftarrow g + 1$$

fim

Retorna Melhor solução encontrada

A rotina BFGS utilizada neste trabalho é fornecida pelo *Matlab Optimization Toolbox*, utilizando a função *fminunc*. Como a função objetivo do PDE-PCV é não diferenciável e o método BFGS exige o valor do gradiente, a função *fminunc* utiliza um método de diferença finita para fornecer o valor do gradiente (COLEMAN; BRANCH; GRACE, 1999).

Figura 5.1 – Fluxograma do algoritmo de ED-BFGS.



Na função $fminunc$, o padrão para escolher a direção de busca no algoritmo Quase-Newton é o método BFGS, que retorna uma matriz Hessiana estimada na solução. O algoritmo Quase-Newton usa o método BFGS para atualizar a aproximação da matriz Hessiana.

Na resolução do PDE-PCV, uma das questões muito importantes é como o algoritmo trata as restrições de igualdade e desigualdade do problema (QIN et al., 2017). No caso do PDE-PCV abordado neste trabalho, existem restrições que precisam ser tratadas: balanço de potência e limites de operação. Para satisfazer as restrições dos limites de operação sobre a potência P_i gerada por cada unidade geradora i (em MW), é adotada a seguinte estratégia:

$$P_i = P_i^{min} + (P_i^{max} - P_i^{min}) * rand(), \quad (5.3)$$

onde P_i^{min} e P_i^{max} são, respectivamente, as saídas de operação mínima e máxima da unidade geradora i (em MW), e $rand()$ é um valor aleatório pertencente ao intervalo $[0,1]$. Esta estratégia gera soluções iniciais cujos limites mínimos e máximos de operação dos geradores não estão violados. Após o cruzamento também projeta na região definida pelos limites.

A fim de atender à restrição de balanço de potência utilizamos o fator de penalização q conforme o problema de otimização (3.37) incorporado na função objetivo do PDE-PCV,

obtendo-se assim uma função *fitness* representada em (5.4):

$$\text{minimizar} \quad \text{Fit} = F_e + q * \left(\sum_{i=1}^n P_i - P_L - P_D \right)^2, \quad (5.4)$$

em que F_e é a função custo de combustível original; q é uma constante de penalidade mantida fixa aplicada à restrição de igualdade do problema; P_i é a saída para unidade geradora i (em MW); P_L são as perdas de potência ativa na transmissão (em MW); P_D é a potência demandada de carga total (em MW). Nos testes abordados neste trabalho, são desconsideradas as perdas de transmissão, portanto, nestes casos $P_L = 0$.

O capítulo seguinte apresenta trabalhos encontrados na literatura que utilizaram métodos híbridos baseados no método ED.

6 MÉTODOS HÍBRIDOS BASEADOS NO ALGORITMO DE EVOLUÇÃO DIFERENCIAL PARA A RESOLUÇÃO DO PROBLEMA DE DESPACHO ECONÔMICO

Neste capítulo, são descritos trabalhos da literatura que utilizam métodos híbridos baseados no algoritmo de ED para resolução do PDE e suas variantes.

6.1 O trabalho de [Coelho e Mariani \(2006a\)](#)

Em [Coelho e Mariani \(2006a\)](#), o método híbrido proposto integra o método ED com a técnica de programação quadrática sequencial (do inglês *sequential quadratic programming* ou SQP) para a resolução do PDE-PCV para sistemas teste de 3, 13 e 40 unidades geradoras. Foi considerado o PDE-PCV com restrições de desigualdade referentes à capacidade de geração de cada unidade geradora e a restrição de balanço de potência. As perdas de potência ativa na transmissão não são consideradas.

O método ED é utilizado como minimizador global e a SQP para busca local, utilizando alguns dos melhores indivíduos da ED como soluções iniciais do problema. A SQP refere-se à minimização ou maximização de uma função objetivo quadrática que é linearmente restrita. No caso, a SQP obtém melhores soluções que os melhores indivíduos testados. Estes indivíduos, na próxima geração da ED, são substituídos pelas soluções obtidas pela SQP.

A implementação da SQP resume-se em três etapas principais, que são brevemente descritas a seguir:

- Calcular uma aproximação da matriz Hessiana da função Lagrangeana usando um método quase-Newton;
- Gerar o sub-problema de programação quadrática;
- Proceder em uma direção de descida usando um método de busca em linha.

Neste trabalho de [Coelho e Mariani \(2006a\)](#), foram testadas seis formas de hibridização utilizando os métodos ED e SQP:

- ED-SQP(1): a cada geração g é verificado se houve melhora do valor da função *fitness* do melhor indivíduo da população da ED. Em caso afirmativo, é realizada uma busca local usando SQP com estimativa inicial igual ao melhor indivíduo da população da ED. Caso a SQP obtenha uma solução melhor que a obtida pelo melhor indivíduo da ED, esta solução substituirá o melhor indivíduo da população da ED na próxima geração;
- ED-SQP(2): idem a ED-SQP(1). Entretanto, se a cada β gerações não houve melhoria do melhor indivíduo da população da ED, é gerada uma nova solução factível;
- ED-SQP(3): idem a ED-SQP(2). Entretanto, é aplicado um algoritmo cultural normativo ([REYNOLDS, 1994](#)) a cada ϕ gerações;
- ED-SQP(4): após o critério de parada de otimização da ED ter sido atingido é realizada uma busca local usando SQP com estimativa inicial igual ao melhor indivíduo da população da ED;
- ED-SQP(5): após o critério de parada de otimização da ED (usando conhecimento normativo a cada ϕ gerações) ter sido atingido é realizada uma busca local usando SQP com estimativa inicial igual ao melhor indivíduo da população da ED;
- ED-SQP(6): idem a ED-SQP(5). Entretanto, é realizada uma análise de diversidade dos valores da função *fitness* de toda a população da ED a cada geração.

Em relação ao tratamento das restrições do problema de despacho econômico são realizadas duas etapas. A primeira etapa consiste em gerar soluções dentro dos limites inferior e superior. Quando uma restrição de limitante inferior não é satisfeita, aplica-se a regra do “reparo” dada por:

$$P_{g+1,i} = P_{g,i} + w * rand_i[0, 1] * (P_{g,i}^{max} - P_{g,i}^{min}), \quad (6.1)$$

em que:

$P_{g,i}$ é um indivíduo da i -ésima unidade geradora na g -ésima geração;

$w \in [0, 1]$ é um parâmetro de projeto;

$rand(0, 1)$ é um valor aleatório gerado com distribuição uniforme entre 0 e 1;

$P_{g,i}^{min}$ e $P_{g,i}^{max}$ são os limitantes inferiores e superiores, respectivamente.

Da mesma forma, se a restrição de limite superior não for atendida, utiliza-se a seguinte regra do "reparo":

$$P_{g+1,i} = P_{g,i} - w * rand_i[0, 1] * (P_{g,i}^{max} - P_{g,i}^{min}), \quad (6.2)$$

onde $w \in [0, 1]$ é um parâmetro de projeto e é um valor gerado aleatoriamente com distribuição normal entre 0 e 1.

Na segunda etapa deve-se considerar restrição de igualdade (2.3). Se a igualdade não for atendida, a função (2.2) é modificada para:

$$\text{minimizar } F_e + q * \left| \sum_{i=1}^n P_i - P_L - P_D \right|, \quad (6.3)$$

onde q é uma constante de penalidade aplicada à restrição de igualdade do problema.

Segundo os autores, as abordagens propostas encontraram soluções satisfatórias de alta qualidade em tempo computacional aceitável. No entanto, verificou-se que em relação à SQP, a desvantagem foi o custo computacional maior da ED e da ED-SQP para obtenção da convergência.

6.2 O trabalho de Chiou (2007)

Em Chiou (2007), o autor traz uma metodologia híbrida denominada *Variable Scaling Hybrid Differential Evolution* (VSHDE) para a resolução do PDE-PCV para sistemas teste de 15 e 40 unidades geradoras considerando as restrições de desigualdade referentes à capacidade de geração de cada unidade geradora, a restrição de balanço de potência, zonas de operação proibidas e perdas de potência ativa nas linhas de transmissão. Neste método, um fator de escala F^g variável é ajustado através da regra de 1/5 das estratégias de evolução a fim de acelerar a pesquisa da solução. O fator de escala F^g variável é utilizado para superar a desvantagem do fator de escala fixo e aleatório e é dado

por:

$$F^{g+1} = \begin{cases} c_d F^g, & \text{se } p_s^g < \frac{1}{5} \\ c_j F^g, & \text{se } p_s^g > \frac{1}{5} \\ F^g, & \text{se } p_s^g = \frac{1}{5}, \end{cases} \quad j = 1, \dots, N_p \quad (6.4)$$

onde p_s^g é a frequência das mutações bem-sucedidas, F^g é o fator de escala utilizado na geração g definido inicialmente como 1,2; $c_d = 0,82$ e $c_j = 1/0,82$.

O processo computacional do método VSHDE para resolver o PDE-PCV proposto no trabalho de [Chiou \(2007\)](#) é descrito usando um fluxograma, conforme mostrado na Figura 6.1. A seguir, são descritos os passos do método VSHDE:

1) A população inicial é gerada aleatoriamente.

2) O processo de mutação na g -ésima geração começa selecionando aleatoriamente cinco indivíduos da população. O vetor mutante é dado por:

$$v_{g+1,i} = P_{g,r1} + F^g(P_{g,r2} - P_{g,r3}) + F^g(P_{g,r4} - P_{g,r5}) \quad (6.5)$$

onde $P_{g,r1}$, $P_{g,r2}$, $P_{g,r3}$, $P_{g,r4}$ e $P_{g,r5}$ são os indivíduos escolhidos aleatoriamente da população.

3) No *crossover*, o indivíduo cujo valor é perturbado e o indivíduo atual são escolhidos por uma distribuição binomial para progredir na operação de cruzamento e gerar a prole e cada gene do i -ésimo indivíduo é reproduzido a partir dos vetores mutantes.

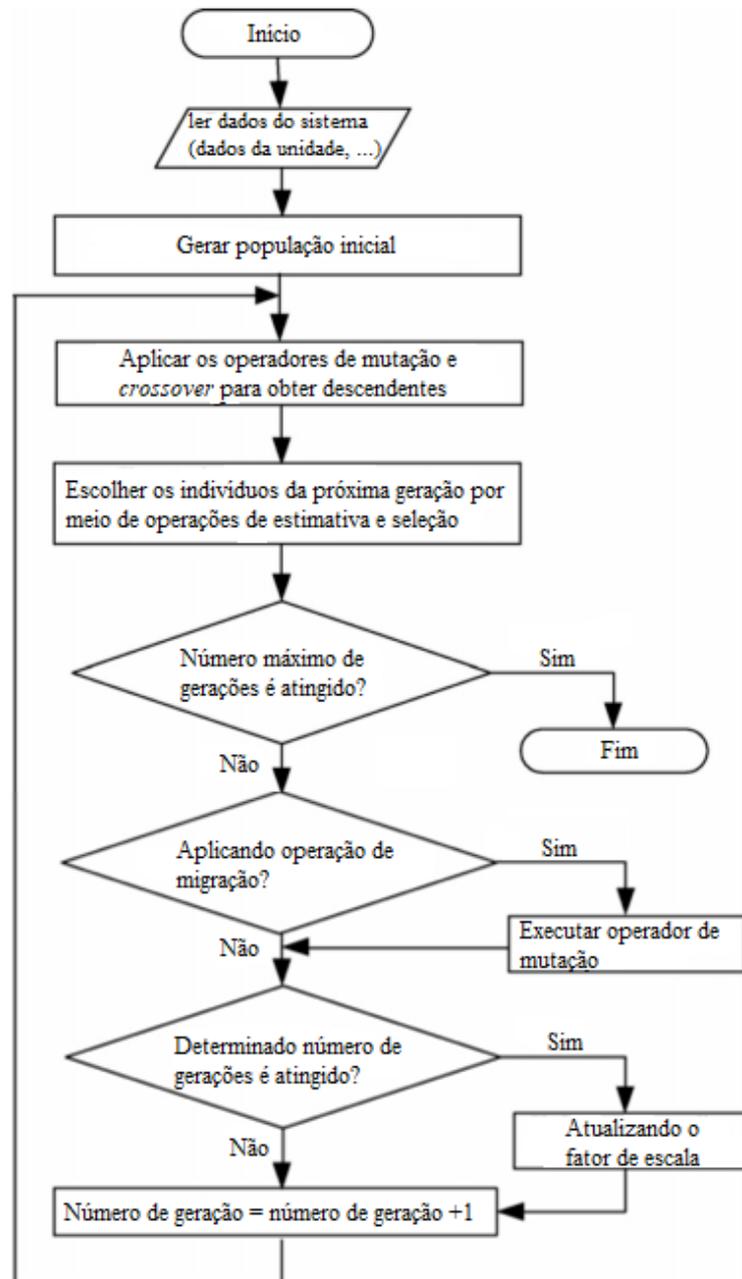
4) Na seleção dos melhores indivíduos, os pais são substituídos por seus filhos se as aptidões dos pais forem piores que as dos filhos. Entretanto, se as aptidões dos pais forem melhores que as dos filhos, eles são mantidos na próxima geração.

5) Para melhorar a investigação do espaço de busca, é introduzida uma fase de migração para regenerar uma nova população de indivíduos. A nova população é gerada com base no melhor indivíduo. O j -ésimo gene do i -ésimo indivíduo é o seguinte:

$$u_{g+1,i,j} = \begin{cases} P_{g+1,best,j} + \sigma_i(P_j^{min} - P_{g+1,best,j}), & \text{se } \delta_i < \frac{P_{g+1,best,j} - P_j^{min}}{P_j^{max} - P_j^{min}} \\ P_{g+1,best,j} + \sigma_i(P_j^{max} - P_{g+1,best,j}), & \text{caso contrário} \end{cases}, \quad (6.6)$$

onde $u_{g+1,i,j}$ é o novo indivíduo da população; $P_{g+1,best,j}$ é a j -ésima componente do

Figura 6.1 – Fluxograma do algoritmo de VSHDE.



Fonte: Adaptado de Chiou (2007).

melhor indivíduo da população corrente, P_j^{min} e P_j^{gmax} são os valores mínimo e máximo, respectivamente para a potência ativa gerada pelo gerador j ; σ_i e δ_i são números gerados com distribuição uniforme no intervalo $[0,1]$; $i = 1, \dots, N_p$, sendo N_p o número de indivíduos; $j = 1, \dots, n$.

6) O fator de escala deve ser atualizado por (6.4) em todas as iterações. Quando a operação de migração (CHIOU; WANG, 1998) realizada ou o fator de escala é muito

pequeno para encontrar a melhor solução, o fator de escala é redefinido em:

$$F^g = 1 - \frac{iter}{itermax}, \quad (6.7)$$

em que $iter$ e $itermax$ são, respectivamente, o número de iteração atual e a iteração máxima.

7) Repetem-se as etapas 2 a 6 até que a quantidade máxima de iterações ou uma solução adequada seja obtida.

Os resultados computacionais obtidos por [Chiou \(2007\)](#) mostraram que o desempenho do método VSHDE é melhor do que aqueles obtidos pelo HDE, AGs e SA. Observou-se através de exemplos que o método VSHDE é especialmente adequado para aplicação na prática de sistemas de grande escala, como o de 40 unidades geradoras.

6.3 O trabalho de Wang, Chiou e Liu (2007)

[Wang, Chiou e Liu \(2007\)](#) apresentam uma nova abordagem de otimização estocástica para determinar uma solução viável para o PDE-PCV considerando três sistemas de teste com 3, 13 e 40 unidades geradoras. São consideradas as restrições de desigualdade referentes à capacidade de geração de cada unidade geradora, a restrição de balanço de potência com a influência das perdas de potência na linha de transmissão e zonas de operação proibidas.

O algoritmo de otimização proposto é chamado de *self-tuning hybrid differential evolution* (*self-tuning* HDE) e é uma modificação ao método VSHDE visto anteriormente. O *self-tuning* HDE, assim como no trabalho de [Chiou \(2007\)](#), também utiliza o conceito de 1/5 da regra de sucesso das estratégias de evolução a fim de superar a desvantagem do fator F^g de escala fixo aleatório. O valor do fator F^g de escala usado pelo *self-tuning* HDE é definido como:

$$F^{g+1} = \begin{cases} c_d F^g, & \text{se } p_s^g < \frac{1}{5} \\ c_j F^g, & \text{se } p_s^g > \frac{1}{5} \quad j = 1, \dots, N_p \\ F^g, & \text{se } p_s^g = \frac{1}{5}, \end{cases} \quad (6.8)$$

onde p_s^g é a frequência das mutações bem-sucedidas, F^g é o fator de escala utilizado na geração g definido inicialmente como 1,2; $c_d = 0,82$ e $c_j = 1/0,82$.

O processo computacional do método *self-tuning* HDE para resolver o PDE-PCV proposto no trabalho de Wang, Chiou e Liu (2007) é descrito usando o mesmo fluxograma mostrado na Figura 6.1, pois a diferença entre os métodos está na fase de migração e na atualização do fator de escala. O método *self-tuning* HDE é brevemente descrito a seguir.

- 1) A população inicial é gerada aleatoriamente de maneira uniforme.
- 2) O processo de mutação na g -ésima geração começa selecionando aleatoriamente cinco indivíduos da população. O vetor mutante é dado por:

$$v_{g+1,i} = P_{g,r1} + F^g(P_{g,r2} - P_{g,r3}) + F^g(P_{g,r4} - P_{g,r5}) \quad (6.9)$$

onde $P_{g,r1}$, $P_{g,r2}$, $P_{g,r3}$, $P_{g,r4}$ e $P_{g,r5}$ são indivíduos escolhidos aleatoriamente da população.

- 3) No *crossover*, o indivíduo cujo valor é perturbado e o indivíduo atual são escolhidos por uma distribuição binomial para progredir na operação de cruzamento e gerar a prole e cada gene do i -ésimo indivíduo é reproduzido a partir dos vetores mutantes.

- 4) Na seleção dos melhores indivíduos, os pais são substituídos por seus filhos se as aptidões dos pais forem piores que as dos filhos. Entretanto, se as aptidões dos pais forem melhores que as dos filhos, eles são mantidos na próxima geração.

- 5) Para melhorar a investigação do espaço de busca, é introduzida uma fase de migração para regenerar uma nova população de indivíduos. A nova população é gerada com base no melhor indivíduo. O j -ésimo gene do i -ésimo indivíduo é o seguinte:

$$u_{g+1,i,j} = \begin{cases} \frac{P_{g+1,best,j} + \sigma_i}{P_j^{min} - P_{g+1,best,j}}, & \text{se } \delta_i < \frac{P_{g+1,best,j} - P_j^{min}}{P_j^{max} - P_j^{min}} \\ \frac{P_{g+1,best,j} + \sigma_i}{P_j^{max} - P_{g+1,best,j}}, & \text{caso contrário} \end{cases}, \quad (6.10)$$

onde $u_{g+1,i,j}$ é o novo indivíduo da população; $P_{g+1,best,j}$ é o melhor indivíduo da população corrente, P_j^{min} e P_j^{max} são os valores mínimos e máximos, respectivamente; σ_i e δ_i são números gerados com distribuição uniforme no intervalo $[0,1]$; $i = 1, \dots, N_p$, sendo N_p o número de indivíduos; $j = 1, \dots, n$.

- 6) Quando o melhor indivíduo da geração atual não apresenta mais melhorias pelas operações de mutação e *crossover* é realizada a operação acelerada definida como:

$$P_{g+1,best} = \begin{cases} P_{g+1,best}, & \text{se } J(P_{g+1,best}) < J(P_{g,best}) \\ P_{g+1,best} - \alpha \nabla J, & \text{caso contrário} \end{cases} \quad (6.11)$$

onde $P_{g+1,best}$ é o melhor indivíduo da população corrente. O gradiente da função objetivo, ∇J , pode ser calculado aproximadamente por diferença finita.

7) O fator de escala deve ser atualizado em todas as iterações. Quando a operação de migração realizada ou o fator de escala é muito pequeno para encontrar a melhor solução, o fator de escala é redefinido em:

$$F^g = 1 - \frac{iter}{itermax}, \quad (6.12)$$

em que $iter$ e $itermax$ são, respectivamente, o número de iteração atual e a iteração máxima.

8) Repetem-se as etapas 2 a 7 até que a quantidade máxima de iteração ou uma solução adequada seja obtida.

Os resultados computacionais mostraram que o algoritmo *self-tuning* HDE proposto por Wang, Chiou e Liu (2007) em relação aos outros três algoritmos analisados, apresentou soluções com menor custo e obteve um melhor desempenho computacional.

6.4 O trabalho de He, Wang e Zhizhong (2008)

Em He, Wang e Zhizhong (2008), o algoritmo de ED não é utilizado como otimizador principal como nos métodos descritos anteriormente. Os autores combinam o AG com a técnica de ED e SQP para melhorar o desempenho do algoritmo. Essa abordagem híbrida é denominada pelos autores como *hybrid genetic algorithm* (HGA). O AG é o otimizador principal, enquanto o ED e o SQP são usados para ajustar a solução da execução do AG. Neste trabalho, a eficácia do algoritmo HGA é analisado para dois casos de teste de PDE-PCV, com 13 e 40 unidades geradoras considerando as restrições de desigualdade referentes à capacidade de geração de cada unidade geradora e a restrição de balanço de potência ativa com a influência das perdas de transmissão.

Para melhorar o desempenho do método de SQP, a função de custo do PDE-PCV é aproximada usando uma função suave e diferenciável, com base no princípio da entropia máxima. São utilizados dois sistemas de teste cuja função objetivo leva em consideração os efeitos dos PCV. É utilizado o método Simplex, que encontra ótimos melhorando o ponto inferior, procurando do ponto inferior em direção ao pivô de $n + 1$ pontos iniciais do Simplex para o problema n -dimensional. O algoritmo Simplex é semelhante ao operador de *crossover* do AG ao utilizar informações de multipontos. Assim, o operador Simplex é atualizado para melhorar a velocidade de convergência e conduzir a população à melhores soluções.

No método proposto em He, Wang e Zhizhong (2008), é empregado um operador de mutação não uniforme. Este operador é descrito abaixo.

Se $P^t = (P_1^t, P_2^t, \dots, P_n^t)$ é um vetor cromossomo de saída de energia e P_i^t é a i -ésima unidade de saída que é escolhida para ser mutada, a nova saída P_i^{mut} será expressa, após a mutação, por:

$$P_i^{mut} = \begin{cases} P_i^t + \Delta(g, P_i^{max} - P_i^g), & \text{se } r = 0, \\ P_i^g - \Delta(t, P_i^g - P_i^{min}), & \text{se } r = 1, \end{cases} \quad (6.13)$$

onde a função $\Delta(g, y)$ retorna um valor no intervalo $[0, y]$ de modo que a probabilidade do valor retornado ser próximo de 0 aumenta com o número de geração g :

$$\Delta(g, y) = y \times (1 - \xi^{(1-g/g_{max})^b}), \quad (6.14)$$

onde ξ é um número de ponto flutuante aleatório no intervalo $[0, 1]$; g é a geração atual; g_{max} é o número máximo de gerações e b é um parâmetro que determina o grau de dependência do número de gerações. Desta forma, o operador faz uma busca uniforme no início da evolução e em estágios posteriores restringe a pesquisa em torno da área local do parâmetro semelhante a um operador de escala.

A função *fitness* para o problema em questão é dada da seguinte forma:

$$Fit(P_i) = \begin{cases} F_e, & \text{se } P_i \text{ é factível} \\ F_e \pm F_e \times g_{max}, & \text{caso contrário} \end{cases} \quad (6.15)$$

onde F_e é o valor da função objetivo, \pm é usado para manter a penalidade, g_{max} é dado como:

$$g_{max} = \max \{0, g_i(P), |h_j(P)|\}, \quad (6.16)$$

com $i \in \{1, 2, \dots, m_1\}$ e $j \in \{1, 2, \dots, m_2\}$. As funções $g_i(P)$ são as restrições de desigualdade, $h_i(P)$ são as restrições de igualdade, m_1 e m_2 são o número de restrições de desigualdade e de restrições de igualdade, respectivamente.

As 13 etapas da abordagem híbrida proposta por He, Wang e Zhizhong (2008) são descritas abaixo.

- 1ª etapa - Obtenha os dados para o sistema.
- 2ª etapa - Inicialize os parâmetros do algoritmo e o contador.
- 3ª etapa - Gerar população inicial.
- 4ª etapa - Avalie a função objetivo e atualize a contagem g .
- 5ª etapa - Identifique o ajuste $Fit_{best}(g)$ (melhor valor de função *fitness* na geração g) da geração atual g .
- 6ª etapa - Se $Fit_{best}(g) < Fit_{best}(g - 1)$ substitua $Fit_{best}(g - 1)$ por $Fit_{best}(g)$, caso contrário, vá para a etapa 7.
- 7ª etapa - Gerar descendência de seleção usando a operação de seleção.
- 8ª etapa - Gerar descendência cruzada usando a operação cruzada simplex.
- 9ª etapa - Gerar descendentes de mutação usando a operação de mutação não uniforme.
- 10ª etapa - Gerar descendência ED usando a operação ED.
- 11ª etapa - Gerar descendência SQP usando a operação SQP.
- 12ª etapa - Gerar descendência inteira com descendência de seleção, descendência cruzada, descendência mutante, descendência ED e descendência SQP.
- 13ª etapa - Enquanto (critério de parada não atendido).

O critério de parada estabelecido se dá quando o número especificado de gerações é atingido.

Os resultados foram comparados com os resultados obtidos utilizando alguns métodos encontrados na literatura. Os resultados numéricos mostram que o método HGA obteve soluções com menor custo e menor tempo computacional em comparação aos métodos analisados.

6.5 O trabalho de Yuan et al. (2009)

Em Yuan et al. (2009), foi desenvolvido um método híbrido para resolver o Problema de Despacho Econômico Dinâmico (PDED) com os efeitos de PCV, integrando o algoritmo de *improved differential evolution* (IDE) ao algoritmo R de Shor (KAPPEL; KUNTSEVICH, 2000). O PDED é utilizado para determinar o despacho das unidades geradoras, a fim de atender à demanda de carga prevista ao longo de um horizonte de tempo. O método híbrido proposto é aplicado para resolver o PDED de um sistema teste com 10 unidades geradoras considerando as restrições de desigualdade referentes à capacidade de geração de

cada unidade geradora, a restrição de balanço de potência sem a influência das perdas, e a restrição dos limites de rampa. Neste trabalho, aplica-se o método IDE como minimizador global e a busca local do algoritmo R de Shor é usada como um ajuste fino. Uma técnica de comparação de seleção baseada em viabilidade e uma regra heurística são criadas para lidar com restrições efetivamente no IDE, que não requer fatores de penalidade ou parâmetros extras e pode guiar a população rapidamente para a região viável.

No algoritmo de IDE, o processo de inicialização se dá criando um conjunto de indivíduos. Essa população inicial é gerada aleatoriamente dentro da faixa viável de tal forma que deve satisfazer a restrição dos limites mínimo e máximo como descritos na equação (2.6). Um componente de um indivíduo é inicializado como mostrado em (6.17).

$$P_i = P_i^{min} + (P_i^{max} - P_i^{min}) U(0, 1), \quad (6.17)$$

onde $U(0, 1)$ é um número aleatório distribuído uniformemente entre 0 e 1.

É muito importante criar uma população de indivíduos que satisfaça as restrições de igualdade do sistema. Para satisfazer as restrições (2.3), uma potência de geração dependente $P_{l,t}$ da unidade l no período t é selecionada aleatoriamente. Essa potência de geração dependente P_{lt} é calculada a partir de (6.18) como

$$P_{l,t} = P_D^t - \sum_{i \neq l}^N P_{i,t} \quad i \neq l, \quad t = 1, \dots, T \quad (6.18)$$

sendo P_D^t a potência total demandada no período t e T é o número de intervalos no horizonte programado.

Durante o processo de mutação, se algum indivíduo violar as restrições dos limites máximos e mínimos, ocorre a seguinte modificação:

$$P_{g+1,i,t} = \begin{cases} P_i^{min}, & \text{se } P_{g+1,i,t} < P_i^{min} \\ P_i^{max}, & \text{se } P_{g+1,i,t} > P_i^{max} \end{cases} \quad (6.19)$$

Para atender as restrições de igualdade (2.3) sem interferir no processo dinâmico característico da ED, são propostas as seguintes etapas heurísticas para todos os NP indivíduos em cada intervalo no horizonte programado.

1ª etapa: Seja a geração atual g .

2ª etapa: Escolha um elemento de potência de geração l do indivíduo aleatoriamente como a potência de geração dependente.

3ª etapa: A potência de geração dependente é calculada usando (6.18). Se a potência de geração calculada não violar as restrições (2.6), vá para a etapa 7. Caso contrário, vá para a próxima etapa.

4ª etapa: Altere o valor do elemento l usando (6.19).

5ª etapa: Um novo elemento aleatório é escolhido e $l = l + 1$. A potência de geração dependente é modificada usando (6.18) .

6ª etapa: Se $l \leq N$ (número de unidades geradoras), volte para a etapa 3. Caso contrário, vá para a próxima etapa.

7ª etapa: O processo de modificação é encerrado.

O novo conjunto de NP indivíduos assim obtido pelo processo de modificação irá satisfazer as restrições de igualdade. Ao gerar o elemento aleatório, é garantido que ele não seja selecionado repetidamente. O ciclo do ED continua com operações de crossover, avaliação e seleção. Assim, os indivíduos da próxima geração podem ser obtidos.

Os passos do método IDE híbrido proposto e do algoritmo R de Shor pode ser resumido da seguinte forma:

Passo 1: Defina os parâmetros do algoritmo.

Passo 2: Inicializar o indivíduo no espaço da população.

Passo 3: Avalie a população inicial.

Passo 4: Para cada indivíduo na população aplique o operador de mutação e crossover para gerar descendentes.

Passo 5: Avalie os indivíduos gerados.

Passo 6: Selecione o indivíduo com a prole de acordo com o método de tratamento de restrições.

Passo 7: Obtenha a próxima população com os indivíduos aceitos.

Passo 8: Até que a condição de parada do IDE seja alcançada use a melhor solução do processo IDE como ponto de partida e resolva o problema PDED usando o algoritmo R de Shor.

Passo 9: Obtenha o resultado final e interrompa o programa.

Em Yuan et al. (2009), uma regra heurística é proposta para lidar com as restrições de igualdade. Em contraste com o método da função de penalidade, este mecanismo de tratamento de restrições não requer fatores de penalidade ou quaisquer parâmetros extras e pode guiar a população para a região viável rapidamente. Além disso, as estratégias aprimoradas podem ser simplesmente incorporadas à ED. Assim, um novo método DE

híbrido melhorado é formado e aplicado para resolver o PDED. Experimentos numéricos mostraram que o método proposto pode obter soluções de melhor qualidade, fornecendo um novo método eficaz, simples e de fácil implementação.

6.6 O trabalho de Chiang e Wang (2010)

Chiang e Wang (2010) propõem o método *hybrid differential evolution with multiplier updating* (HDE-MU) para resolver problemas de *cogeneration economic dispatch* (CED). A atualização do multiplicador (MU) é introduzida para evitar a deformação da função Lagrangiana aumentada, que é adotada para gerenciar as restrições do sistema de problemas do CED.

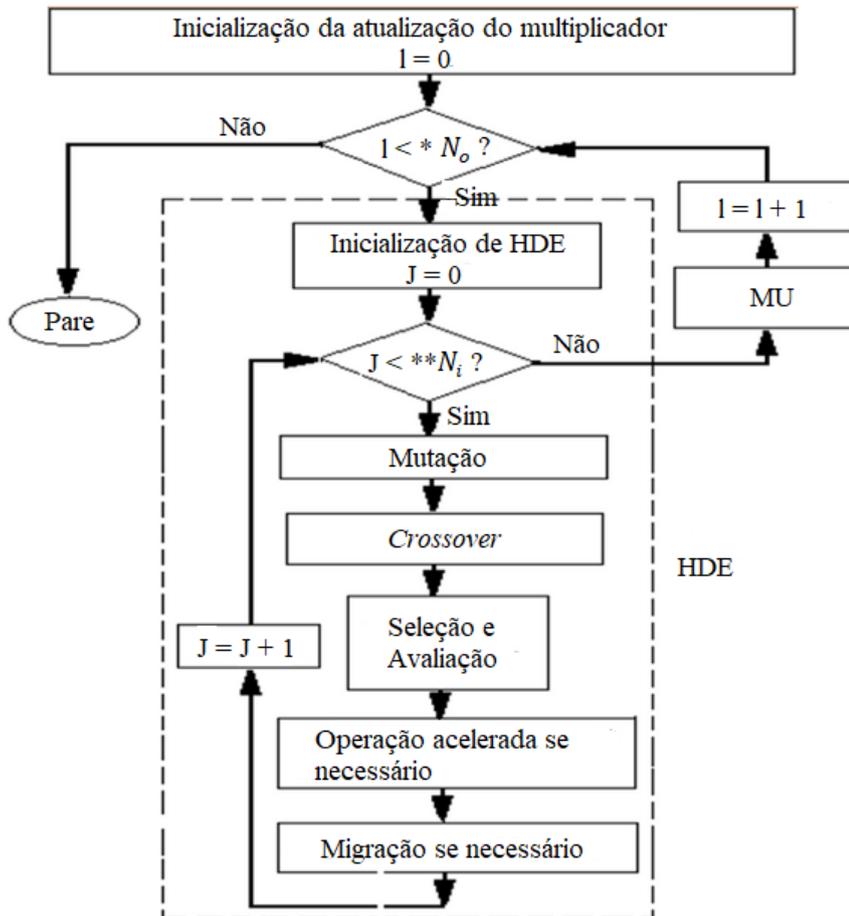
No HDE, a competição um para um terá uma velocidade de convergência mais rápida para dar uma maior probabilidade em direção a uma solução global com muito menos tempo de computação, além de poder usar uma pequena população de indivíduos no processo evolutivo. Nesse processo evolutivo, há duas questões críticas: direção evolutiva e diversidade populacional. Pode acontecer de ocorrer a redução da carga computacional e aumentar a probabilidade de encontrar rapidamente uma solução local. Assim, se a diversidade da população aumenta, o genótipo dos filhos difere mais dos pais. Desse modo, uma população altamente diversa pode aumentar a probabilidade de explorar a região de solução global e prevenir uma convergência prematura para uma solução local. Esses dois fatores importantes são equilibrados pelo emprego da operação acelerada e da migração em HDE que pode determinar uma direção eficaz na busca pela solução, ao mesmo tempo que mantém uma diversidade apropriada para uma pequena população.

A Figura 6.2 mostra o fluxograma do algoritmo proposto pelos autores. A função Lagrangiana aumentada denominada por L_a é usada para obter um valor mínimo com os parâmetros de penalidade e multiplicadores fornecidos, que são então atualizados para produzir um limite superior de L_a . Quando ambas as iterações internas e externas tornam-se suficientemente grandes, o método converge para um ponto de sela do problema dual. As vantagens do HDE-MU proposto são que o HDE procura com eficiência a solução ideal no processo de despacho econômico e o MU enfrenta as restrições do sistema de maneira eficaz.

A complicação do problema de CED reside nas restrições impostas pelas dependências mútuas de demanda múltipla e capacidade de energia térmica. O HDE auxilia o método proposto na busca eficiente e na exploração refinada. O MU ajuda o método proposto a evitar deformar a função Lagrangiana e resultar em dificuldade de busca da solução. O algoritmo proposto integra o HDE e o MU que consegue uma ampla gama de parâmetros de penalidade e uma pequena população. Os resultados da simulação demonstram que o algoritmo proposto apresenta mais vantagens na solução dos problemas

de CED do que os métodos anteriores. As contribuições deste artigo são que o MU lida efetivamente com as restrições da região viável, o HDE busca com eficiência as soluções ótimas no processo de ED, fornecendo um algoritmo válido e eficaz para o problema CED.

Figura 6.2 – Fluxograma do algoritmo de HDE-MU.



Fonte: Adaptado de Chiang e Wang (2010).

Os dois exemplos de cogeração empregados demonstram que o algoritmo proposto possui os benefícios da objetividade, facilidade de implementação, melhor eficácia que os métodos anteriores, melhor eficácia e eficiência do que a ED clássica e ajuste automático da penalidade atribuída aleatoriamente a um valor apropriado.

6.7 O trabalho de Duvvuru e Swarup (2011)

Duvvuru e Swarup (2011) propõem um algoritmo híbrido que combina o método de ponto interior (MPI) com o método ED para resolver o PDE-PCV. O método proposto foi utilizado para resolver três diferentes casos de teste envolvendo 13 e 40 unidades geradoras considerando apenas as restrições de desigualdade referentes à capacidade de geração de cada unidade geradora e a restrição de balanço de potência. As perdas não são consideradas.

O primeiro estágio do algoritmo emprega o MPI primal-dual para minimizar a função custo sem considerar o efeito do ponto de válvula, enquanto que no segundo estágio esse efeito é considerado.

O MPI pode resolver um problema de programação linear em grande escala, buscando a solução no interior da região viável, ao contrário do método simplex, que se restringe à fronteira. O MPI foi originalmente proposto para resolver problemas de programação linear, porém, mais tarde foi implementado para lidar com problemas de programação não linear com eficiência.

O algoritmo híbrido é resumido pelas seguintes etapas.

1ª etapa: Obtenha os dados dos geradores.

2ª etapa: Resolva o PDE sem considerar os efeitos de PCV utilizando o MPI (estágio 1). A solução obtida é chamada P_{i_ipm} .

3ª etapa: Gere uma população inicial em torno de P_{i_ipm} utilizando a estratégia da ED.

4ª etapa: Resolva o PDE com os efeitos de PCV utilizando a ED.

O MPI é uma técnica matemática sofisticada conhecida por sua rápida convergência e tratamento de problemas de grande escala com restrições de igualdade e desigualdade. Portanto, é usado para minimizar a função de custo suave sem considerar o efeito do ponto da válvula. O MPI empregado é capaz de alcançar a solução do estágio 1 em menos de dez iterações. Uma nova estratégia é introduzida para gerar a população inicial para o estágio 2 usando evolução diferencial. O espaço de busca é reduzido a uma região menor, e o ED é capaz de alcançar a convergência em tempos mais curtos.

Os resultados obtidos são comparados com alguns resultados encontrados na literatura. O algoritmo proposto por Duvvuru e Swarup (2011) supera os resultados dos algoritmos comparados, obtendo valores de função objetivo menores e também tempo computacional menor.

6.8 O trabalho de Zou et al. (2016)

Em Zou et al. (2016), é apresentado mais um algoritmo de ED aprimorado, denominado também pelos autores como IDE. A finalidade é resolver o PDE com e sem os efeitos de PCV para casos de teste com 3, 6, 38 e 40 unidades geradoras considerando apenas as restrições de desigualdade referentes à capacidade de geração de cada unidade geradora e a restrição de balanço de potência. As perdas não são consideradas. A diferença do IDE para o algoritmo de ED clássico, está em três aspectos. O primeiro é que dois operadores de mutação são usados para gerar vetores mutantes. O segundo está na operação de

crossover, onde uma taxa de cruzamento dinâmica é usada para atualizar os vetores testes. O terceiro aspecto está em uma randomização (sequência aleatória) populacional adotada para superar a convergência prematura. Também é introduzido um processo de reparo modificado para lidar com as violações das restrições.

No algoritmo IDE, cada indivíduo P_i da população inicial é gerada de forma aleatória como segue:

$$P_i = P_i^{min} + (P_i^{max} - P_i^{min}) U(0, 1), \quad (6.20)$$

onde $U(0, 1)$ é um número aleatório distribuído uniformemente entre 0 e 1, e P_i^{min} e P_i^{max} são os limites mínimos e máximos, respectivamente.

Na operação de mutação, o IDE adota dois operadores de mutação para produzir vetores mutantes de acordo com:

$$v_{g,i} = \begin{cases} P_{g,r1} + rand_1(P_{g,r2} - P_{g,r3}), & \text{se } rand[0, 1] < p_g \\ P_{g,r1} + rand_2(P_{g,r2} - P_{g,r3}) + rand_3(P_{g,r4} - P_{g,r5}), & \text{caso contrário} \end{cases} \quad (6.21)$$

onde $rand_1$, $rand_2$, $rand_3$ e $rand[0, 1]$ são quatro valores aleatórios uniformemente distribuídos no intervalo $[0,1]$. Como são utilizados números aleatórios para gerar fatores de mutação para IDE em cada geração, um fator de mutação pode ter qualquer valor na faixa $[0,1]$ por meio do processo evolutivo, o que é útil para pesquisa global e local. $P_{g,r1}$, $P_{g,r2}$, $P_{g,r3}$, $P_{g,r4}$ e $P_{g,r5}$ são indivíduos aleatórios da população e p_g é ajustado de acordo com:

$$p_g = P_i^{min} + (P_i^{max} - P_i^{min}) \frac{g}{g^{max}}, \quad (6.22)$$

onde g é a geração atual e g^{max} é a geração máxima.

Uma taxa de crossover dinâmica CR_g é usada para obter vetores testes e é fornecida como:

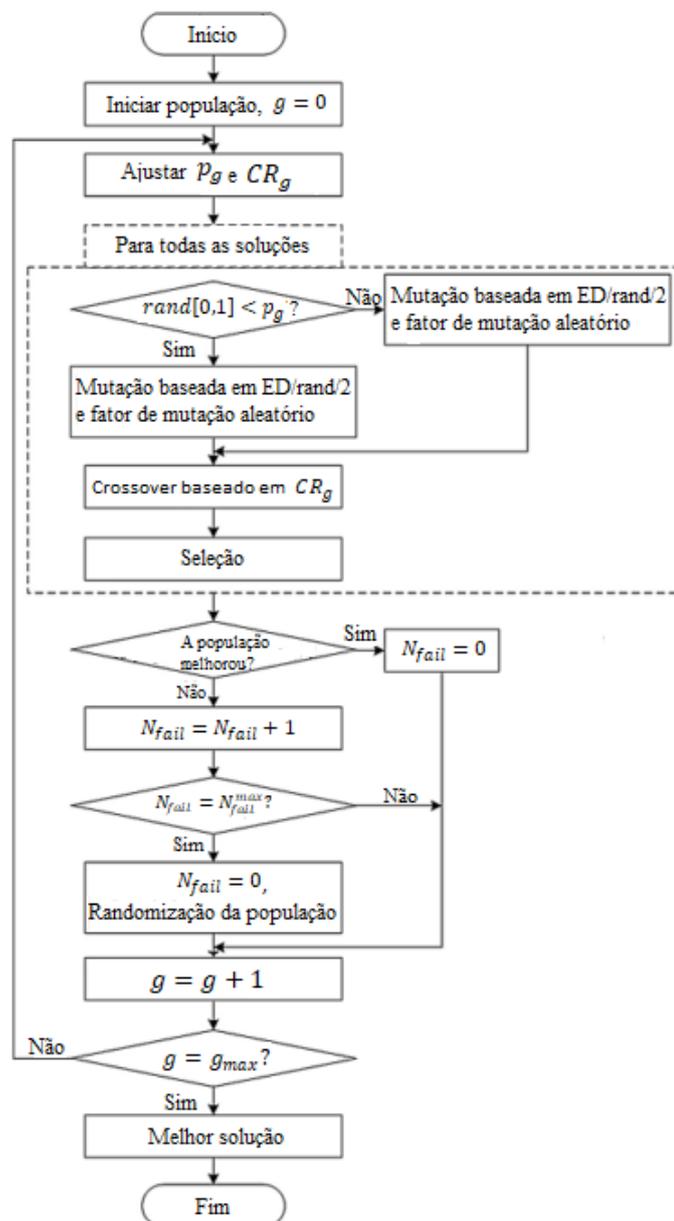
$$CR_g = CR^{max} - (CR^{max} - CR^{min}) \frac{g}{g^{max}}, \quad (6.23)$$

onde CR^{max} e CR^{min} são as taxas de crossover máxima e mínima, respectivamente. No estágio inicial do processo evolutivo, um valor CR_g alto permite que o IDE atualize vetores testes usando mais informações de vetores mutantes, o que é útil para a pesquisa global.

Para não ficar preso em mínimos locais, o IDE utiliza a randomização populacional, que é dada da seguinte forma: dado o número máximo permitido de falhas consecutivas

N_{fail}^{max} e o número atual de falhas consecutivas $N_{fail} = 0$. Por outro lado, $N_{fail} = N_{fail} + 1$. Se N_{fail} alcança N_{fail}^{max} , N_{fail} será definido como 0 imediatamente. Ao mesmo tempo, todas as soluções da população são classificadas em ordem crescente de acordo com seus valores de função objetivo. A primeiras $N_{fail}/2$ soluções são reservadas para a próxima geração e são utilizadas principalmente para manter a qualidade da população. As restantes $N_{fail}/2$ soluções são excluídas da população. Além disso, $N_{fail}/2$ novas soluções são introduzidas na população, e são geradas usando o mesmo método aleatório da inicialização da população. Essas soluções geradas aleatoriamente podem aumentar a diversidade da população.

Figura 6.3 – Fluxograma do algoritmo de IDE.



Fonte: Adaptado de Zou et al. (2016).

O IDE combina $ED/rand/2$ e $ED/rand/1$ para melhorar suas habilidades de pesquisa global e local. É mais provável que o primeiro seja escolhido para conduzir o

operador de mutação na evolução inicial. É mais provável que o último seja escolhido na evolução posterior, o que aumenta a capacidade de refinamento do IDE. Além disso, cada solução individual pode ser usada para gerar vetores mutantes no IDE. Isso é muito benéfico para manter a diversidade da população. A melhor solução conhecida em uma geração não é necessária para a geração de novas soluções candidatas, portanto, pode efetivamente evitar a convergência prematura causada pela forte atração da melhor solução conhecida.

A Figura 6.3 mostra o fluxograma do algoritmo proposto por Zou et al. (2016). O processo de reparo modificado mostrou-se mais eficiente na redução das violações das restrições em relação a ED clássica. Com base neste processo eficaz, o IDE é capaz de obter rapidamente soluções viáveis no estágio evolutivo inicial. São utilizadas quatro variantes da ED para lidar com oito casos de teste do PDE-PCV. De acordo com os resultados experimentais, o IDE pode produzir resultados que são pelo menos iguais e, em alguns casos, melhores do que os das outras três abordagens da DE. Em comparação com as abordagens da literatura, o IDE é mais eficiente em encontrar as melhores soluções viáveis na maioria dos casos.

Através dos métodos vistos neste capítulo, é possível observar a qualidade do método ED para a resolução do PDE-PCV. No próximo capítulo, serão apresentados os testes numéricos realizados neste trabalho.

7 TESTES NUMÉRICOS

Neste capítulo, são apresentados os resultados obtidos pelo método ED-BFGS aplicado ao PDE-PCV com restrições de balanço de potência e limites de operação das unidades geradoras. Foram realizadas 4 etapas de testes envolvendo os sistemas com 3, 13, 19 e 40 unidades geradoras. Os dados dos sistemas para os casos com 3 e 40 unidades geradoras foram extraídos de [Sinha, Chakrabarti e Chattopadhyay \(2003\)](#). Para os sistemas com 13 e 19 unidades geradoras, os dados foram extraídos de [Samed \(2004\)](#) e [Balamurugan, Muralisachithnndam e Krishnan \(2014\)](#), respectivamente. Os resultados são comparados aos obtidos pelo método ED clássico, também implementado neste trabalho. São realizadas ainda, comparações com os resultados de alguns trabalhos encontrados na literatura que resolvem as mesmas instâncias, como [Coelho e Mariani \(2006a\)](#), [Silva \(2014\)](#), [Zou et al. \(2016\)](#), [Secui \(2015\)](#), [Zhao et al. \(2018\)](#), [Le et al. \(2015\)](#), [He, Rao e Huang \(2016\)](#), [Elsayed et al. \(2017\)](#) e [Li, Fu e Lu \(2021\)](#).

Nos testes realizados, o modo de cruzamento adotado no método ED e ED-BFGS é da forma binomial e a estratégia de mutação escolhida em ambos foi $ED/rand/1$, como descritos no Capítulo 4. Cada execução dos algoritmos foi realizada com uma população inicial gerada de forma aleatória e indicada em cada caso. Para todos os casos, o valor referente às perdas de transmissão foi desconsiderado, ou seja, $P_L = 0$. Os algoritmos dos métodos ED e ED-BFGS para resolução do PDE-PCV, foram implementados no *software* Matlab® versão 2010b. Todos os testes foram realizados em um computador com processador Intel® Core™ i3-7100U de 2,40 GHz com 4 GB de memória RAM.

7.1 Caso 1: 3 unidades geradoras

Na Tabela A.1, que se encontra no Apêndice A, são apresentados os coeficientes utilizados nos testes para um sistema com 3 unidades geradoras, extraídos de [Sinha, Chakrabarti e Chattopadhyay \(2003\)](#). Para este caso, foi considerada a potência demandada igual a 850 MW. O critério de parada adotado para os algoritmos de ED e ED-BFGS foi de 300 gerações e uma população com 30 indivíduos. O tempo de execução dos algoritmos de ED e ED-BFGS foram, respectivamente, 0,53 (s) e 6,50 (s). Os parâmetros adotados foram $F = 0,9$ e $CR = 0,7$ para a ED, e $F = 0,8$ e $CR = 0,7$ para a ED-BFGS, determinados por testes numéricos. Foram realizadas 100 simulações com os algoritmos de ED e ED-BFGS e são apresentados os melhores resultados obtidos destas simulações.

Considerando que os dados e a formulação para o PDE-PCV utilizados neste trabalho são os mesmos utilizados nos trabalhos de [Coelho e Mariani \(2006a\)](#), [Silva \(2014\)](#)

e Zou et al. (2016), é possível realizar uma comparação entre os mesmos. Os métodos utilizados em cada trabalho são, respectivamente, ED-SQP(1), PDPCBLM e SADE. Na Tabela 7.1 é apresentado o custo mínimo obtido por cada um destes métodos, enquanto que a Tabela 7.2 mostra a melhor solução de cada um, sendo a saída de potência P_i (em MW) para cada uma das unidades geradoras de energia termoelétrica e também a soma das potências ($\sum_{i=1}^n P_i(MW)$).

Tabela 7.1 – Função custo para o PDE-PCV com 3 unidades geradoras.

Método	Custo mínimo (\$)
ED-SQP(1) (COELHO; MARIANI, 2006a)	8234,0740
PDPCBLM (SILVA, 2014)	8234,200
SADE (ZOU et al., 2016)	8234,0200
ED	8234,0800
ED-BFGS	8233,9998

Tabela 7.2 – Soluções para o PDE-PCV com 3 unidades geradoras.

Método	$P_1(MW)$	$P_2(MW)$	$P_3(MW)$	$\sum_{i=1}^n P_i(MW)$
ED-SQP(1)	300,26418	149,73583	400,00000	850,00000
PDPCBLM	300,06712	149,94901	399,98387	850,00000
SADE	299,7595	150,4285	399,8120	850,0000
ED	300,26418	149,73583	400,0000	850,0000
ED-BFGS	300,2573	149,7333	399,9989	850,0000

Figura 7.1 – a) Convergência da melhor solução do método ED para o caso 1. b) Convergência da melhor solução do método ED-BFGS para o caso 1.

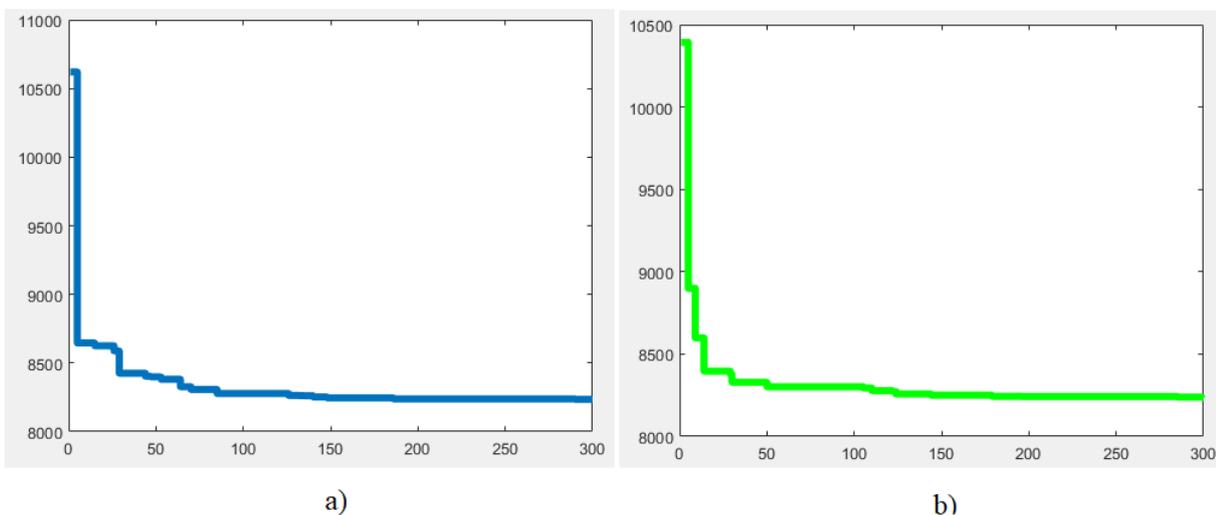


Tabela 7.3 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 3 unidades geradoras.

	F = 0,5	F = 0,6	F = 0,7
CR = 0,1	8291,9782	8360,8942	8407,8813
CR = 0,2	8313,1039	8389,0592	8391,8746
CR = 0,3	8299,0027	8326,4939	8297,7006
CR = 0,4	8269,1437	8250,9217	8279,9319
CR = 0,5	8287,4960	8247,1459	8266,9887
CR = 0,6	8246,6345	8236,9607	8243,6976
CR = 0,7	8243,6215	8243,5348	8243,1047
CR = 0,8	8237,5933	8236,8751	8237,9352
CR = 0,9	8234,6973	8234,8783	8238,5005
CR = 1,0	8585,5269	8585,5269	8585,5269
	F = 0,8	F = 0,9	F = 1,0
CR = 0,1	8389,7971	8404,1721	8392,9778
CR = 0,2	8393,7473	8322,8339	8281,0406
CR = 0,3	8262,1409	8303,3349	8336,0569
CR = 0,4	8303,2013	8311,1904	8330,6082
CR = 0,5	8277,4059	8241,7417	8314,5018
CR = 0,6	8238,7451	8242,4411	8241,9207
CR = 0,7	8233,9998	8234,0450	8234,7658
CR = 0,8	8235,9209	8235,9456	8234,5284
CR = 0,9	8234,3605	8234,3284	8234,2080
CR = 1,0	8354,0467	8354,0265	8354,0158

Como podemos observar na Tabela 7.1, o método ED-BFGS mostra-se eficiente para este caso, conseguindo uma solução muito próxima aos demais métodos analisados. Mesmo sendo uma diferença numérica muito pequena, isso mostra que o método ED-BFGS conseguiu uma solução superior em relação ao método ED clássico e aos outros métodos neste primeiro caso. A Tabela 7.2 mostra que todos os valores de geração estão dentro dos limites, e também que a demanda do sistema foi atendida.

Na Figura 7.1 são apresentados os gráficos de convergência para a melhor solução obtida pelos métodos ED e ED-BFGS. Nos gráficos de convergência, o eixo x representa o número de gerações executadas e o eixo y representa o valor obtido na avaliação da função *fitness*. Observando esses gráficos, vemos que as funções para ambos os métodos convergem rapidamente, já que se trata de um problema de menor dimensão. Também é observado nos gráficos, que a busca local realizada pelo método BFGS interferiu positivamente no processo de minimização, fazendo com que a convergência para a solução fosse mais rápida. Na Figura 7.1 b), vemos que o método ED-BFGS obtém sua melhor solução em aproximadamente 150 gerações, enquanto que o método ED obtém sua melhor solução em aproximadamente 200 gerações, como mostra a Figura 7.1 a).

A Tabela 7.3 apresenta as melhores soluções obtidas pelo método ED-BFGS com

diferentes parâmetros para o fator de mutação F e a probabilidade de cruzamento CR . O fator de mutação F foi variado de 0,5 a 1,0 e a probabilidade de cruzamento CR foi variada de 0,1 a 1,0. Foram realizadas 100 simulações e são apresentados os melhores resultados obtidos destas simulações. Na Tabela 7.3 a melhor solução obtida pelo método ED-BFGS está destacada em negrito.

7.2 Caso 2: 13 unidades geradoras

Na Tabela A.2, que se encontra no Apêndice A, são apresentados os coeficientes utilizados nos testes com 13 unidades geradoras, extraídos de Samed (2004). Para este caso, foi considerada a potência demandada igual a 2520 MW. O critério de parada adotado para os métodos ED e ED-BFGS foi de 300 gerações com uma população de 100 indivíduos. O tempo de execução dos algoritmos de ED e ED-BFGS foram, respectivamente, 2,88 (s) e 22,66 (s). Os parâmetros adotados foram $F = 1,0$ e $CR = 1,0$ para a ED, e $F = 0,9$ e $CR = 0,9$ para a ED-BFGS, determinados por testes numéricos. Foram realizadas 100 simulações com os algoritmos de ED e ED-BFGS e são apresentados os melhores resultados obtidos destas simulações.

Considerando que os dados e a formulação para o PDE-PCV utilizados neste trabalho são os mesmos utilizados nos trabalhos de Silva (2014), Secui (2015) e Zhao et al. (2018), é possível realizar uma comparação entre os mesmos. Os métodos utilizados em cada trabalho são, respectivamente, PDPCBLM, MABC/P/Log e MCSA. Na Tabela 7.4 é apresentado o custo mínimo obtido por cada um destes métodos, enquanto que na Tabela 7.5 é apresentada a melhor solução obtida por cada um, sendo a saída de potência P_i (em MW) para cada uma das unidades geradoras de energia termoeletrica e a soma das potências ($\sum_{i=1}^n P_i(MW)$).

Tabela 7.4 – Função custo para o PDE-PCV com 13 unidades geradoras.

Método	Custo mínimo (\$)
PDPCBLM (SILVA, 2014)	24144,4900
MABC/P/Log (SECUI, 2015)	24169,9177
MCSA (ZHAO et al., 2018)	24169,9177
ED	24174,2081
ED-BFGS	24169,9178

Como podemos observar na Tabela 7.4, o método ED-BFGS mostra-se eficiente para este caso, conseguindo uma solução melhor que o método ED e próxima dos métodos heurísticos MABC/P/Log e MCSA. Em relação ao método exato proposto em Silva (2014), o método ED-BFGS não conseguiu uma solução superior, mas mostrou-se competitivo com uma solução com valor de função objetivo próxima. Comparando o resultado obtido pelo método ED-BFGS com os resultados de Secui (2015) e Zhao et al. (2018), vemos que

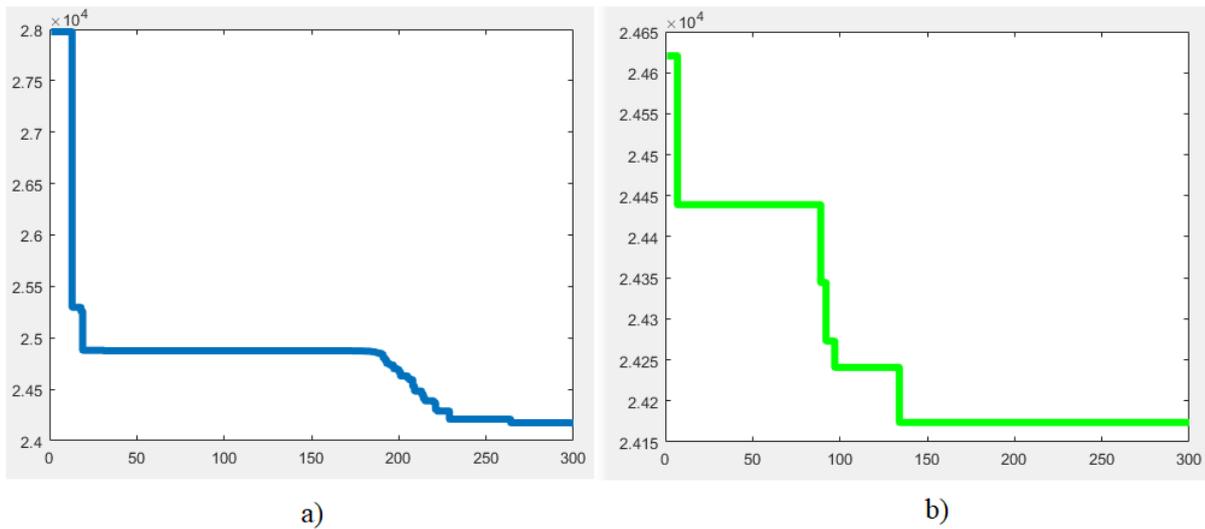
Tabela 7.5 – Soluções para o PDE-PCV com 13 unidades geradoras.

Potência (MW)	PDPCBLM	MABC/P/Log	MCSA	ED	ED-BFGS
P ₁	628,08088	628,3185	628,3185	626,7482	628,3185
P ₂	298,79312	299,1993	299,1993	298,7111	299,1993
P ₃	298,79312	299,1993	299,1993	298,7111	299,1993
P ₄	159,53795	159,7331	159,7331	159,5445	159,7331
P ₅	159,53795	159,7331	159,7331	159,5445	159,7331
P ₆	159,53795	159,7331	159,7331	159,5445	159,7331
P ₇	159,53795	159,7331	159,7331	159,5445	159,7331
P ₈	159,53795	159,7331	159,7331	159,5445	159,7331
P ₉	159,53795	159,7331	159,7331	159,5445	159,7331
P ₁₀	76,92247	77,39999	77,3999	77,42328	77,39991
P ₁₁	76,92247	77,39999	77,3999	77,42328	77,39991
P ₁₂	91,63009	87,6845	92,3999	91,86111	92,39989
P ₁₃	91,63009	92,3999	87,6845	91,86111	87,68458
$\sum_{i=1}^n P_i$	2519,99994	2519,9999	2520,000	2520,0061	2520,0000

Tabela 7.6 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 13 unidades geradoras.

	F = 0,5	F = 0,6	F = 0,7
CR = 0,1	24542,6203	24604,0179	24561,2321
CR = 0,2	24446,3481	24527,0195	24446,3558
CR = 0,3	24458,9524	24462,9378	24460,2924
CR = 0,4	24439,2013	24344,0444	24487,9266
CR = 0,5	24525,8109	24519,2345	24417,3561
CR = 0,6	24446,9768	24560,2054	24511,4753
CR = 0,7	24550,9710	24574,2888	24541,5594
CR = 0,8	24209,2209	24299,0196	24242,3505
CR = 0,9	24173,9196	24171,5336	24179,5722
CR = 1,0	24293,7958	24273,8555	24170,1141
	F = 0,8	F = 0,9	F = 1,0
CR = 0,1	24612,8564	24714,0967	24727,5822
CR = 0,2	24490,1158	24580,8306	24595,6049
CR = 0,3	24433,6221	24516,6064	24560,9975
CR = 0,4	24457,8096	24431,6479	24449,6058
CR = 0,5	24547,2888	24481,9987	24612,6831
CR = 0,6	24545,5919	24605,5028	24612,5760
CR = 0,7	24555,2393	24613,6420	24555,0066
CR = 0,8	24237,0649	24224,2748	24179,4123
CR = 0,9	24170,3701	24169,9178	24216,2073
CR = 1,0	24170,8823	24170,6553	24273,8555

Figura 7.2 – a) Convergência da melhor solução do método ED para o caso 2. b) Convergência da melhor solução do método ED-BFGS para o caso 2.



a diferença é quase zero. A Tabela 7.5 mostra que todos os valores de geração estão dentro dos limites solicitados, e também que a demanda do sistema foi atendida.

Na Figura 7.2 são apresentados os gráficos de convergência para a melhor solução obtida pelos métodos ED e ED-BFGS. Nos gráficos de convergência, o eixo x representa o número de gerações executadas e o eixo y representa o valor obtido na avaliação da função *fitness*. Observando esses gráficos, vemos que as funções para ambos os métodos convergem rapidamente, já que se trata de um problema de pequena dimensão. Também é observado nos gráficos, que a busca local realizada pelo método BFGS interferiu positivamente no processo de minimização, fazendo com que a convergência para a solução fosse mais rápida. Na Figura 7.2 b), vemos que o método ED-BFGS obtém sua melhor solução em aproximadamente 150 gerações, precocemente quando comparado com o método ED, ilustrado na Figura 7.2 a).

A Tabela 7.6 apresenta as melhores soluções obtidas pelo método ED-BFGS com diferentes parâmetros para o fator de mutação F e a probabilidade de cruzamento CR . O fator de mutação F foi variado de 0,5 a 1,0 e a probabilidade de cruzamento CR foi variada de 0,1 a 1,0. Foram realizadas 100 simulações e são apresentados os melhores resultados obtidos destas simulações. Na Tabela 7.6 a melhor solução obtida pelo método ED-BFGS está destacada em negrito.

7.3 Caso 3: 19 unidades geradoras

Na Tabela A.3, que se encontra no Apêndice A, são apresentados os coeficientes utilizados nos testes para um sistema com 19 unidades geradoras, extraídos de Balamurugan, Muralisachithnndam e Krishnan (2014). Para este caso, foi considerada a potência

Tabela 7.7 – Função custo para o PDE-PCV com 19 unidades geradoras.

Método	Custo mínimo (\$)
PDPCBLM (SILVA, 2014)	17551,0460
ED	17005,1048
ED-BFGS	16982,5601

Tabela 7.8 – Soluções para o PDE-PCV com 19 unidades geradoras.

Potência (MW)	PDPCBLM	ED	ED-BFGS
P ₁	200,35634	121,8329	104,3720
P ₂	308,49096	370,7512	371,2835
P ₃	226,22324	247,0016	249,8589
P ₄	25,00000	24,30101	24,92514
P ₅	63,75000	59,23283	63,71548
P ₆	186,77848	271,199	269,7616
P ₇	63,75000	55,4475	62,93011
P ₈	299,37001	148,6519	130,1541
P ₉	200,00000	201,4534	201,1047
P ₁₀	40,00000	37,40973	39,59142
P ₁₁	150,00000	147,6397	149,8667
P ₁₂	75,00000	71,56939	74,86636
P ₁₃	63,75000	60,6148	63,7417
P ₁₄	95,00000	93,52962	94,86275
P ₁₅	120,53097	208,3584	216,8842
P ₁₆	80,00000	79,60325	79,6400
P ₁₇	80,00000	76,99884	79,97481
P ₁₈	230,00000	229,3547	229,8043
P ₁₉	400,00000	402,9884	400,3321
$\sum_{i=1}^n P_i$	2908,00009	2907,9382	2908,0005

demandada igual a 2908 MW. O critério de parada adotado para os métodos ED e ED-BFGS foi de 500 gerações com uma população de 100 indivíduos. O tempo de execução dos algoritmos de ED e ED-BFGS foram, respectivamente, 4,55 (s) e 52,30 (s). Os parâmetros adotados foram $F = 0,8$ e $CR = 0,9$ para a ED, e $F = 1,0$ e $CR = 0,9$ para o ED-BFGS e foram determinados por testes numéricos. Foram realizadas 100 simulações com os algoritmos de ED e ED-BFGS e são apresentados os melhores resultados obtidos destas simulações.

Na revisão bibliográfica realizada neste trabalho, verificou-se que apenas Silva (2014) resolveu o PDE-PCV com este caso. Gonçalves (2015) e Balamurugan, Muralisachithndam e Krishnan (2014) também utilizam o caso com 19 geradores, entretanto, resolvem o PDE-PCV combinado com o despacho ambiental. Então, considerando que os dados e a formulação para o PDE-PCV utilizados neste trabalho são os mesmos utilizados no trabalho de Silva (2014), é possível realizar uma comparação entre os mesmos. Na Tabela 7.7 é apresentado o custo mínimo obtido por cada método, enquanto que na Tabela 7.8 é

Tabela 7.9 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 19 unidades geradoras.

	F = 0,5	F = 0,6	F = 0,7
CR = 0,1	17870,2054	17865,5108	18239,9775
CR = 0,2	17598,3910	17592,2184	17697,6758
CR = 0,3	17344,5246	17374,3476	17546,2291
CR = 0,4	17279,4697	17267,9742	17372,542
CR = 0,5	17262,3080	17374,0788	17333,947
CR = 0,6	17207,9697	17264,5693	17398,2374
CR = 0,7	17309,9575	17334,0814	17501,0238
CR = 0,8	17242,3382	17329,9242	17388,5515
CR = 0,9	17061,9850	17132,3365	17152,6343
CR = 1,0	17805,4394	17841,6474	17840,9652
	F = 0,8	F = 0,9	F=1,0
CR = 0,1	18263,4868	17939,9797	18270,1078
CR = 0,2	17845,9468	18151,4856	17681,5749
CR = 0,3	17454,6366	17576,0828	17875,9976
CR = 0,4	17361,9426	17499,1356	17541,0470
CR = 0,5	17446,0130	17489,3121	17464,2945
CR = 0,6	17419,0258	17441,2742	17616,2595
CR = 0,7	17726,8290	17498,9896	17535,8713
CR = 0,8	17463,9469	17483,9129	17288,2055
CR = 0,9	17842,4298	17091,5945	16982,5601
CR = 1,0	17152,2566	17817,6803	18631,9830

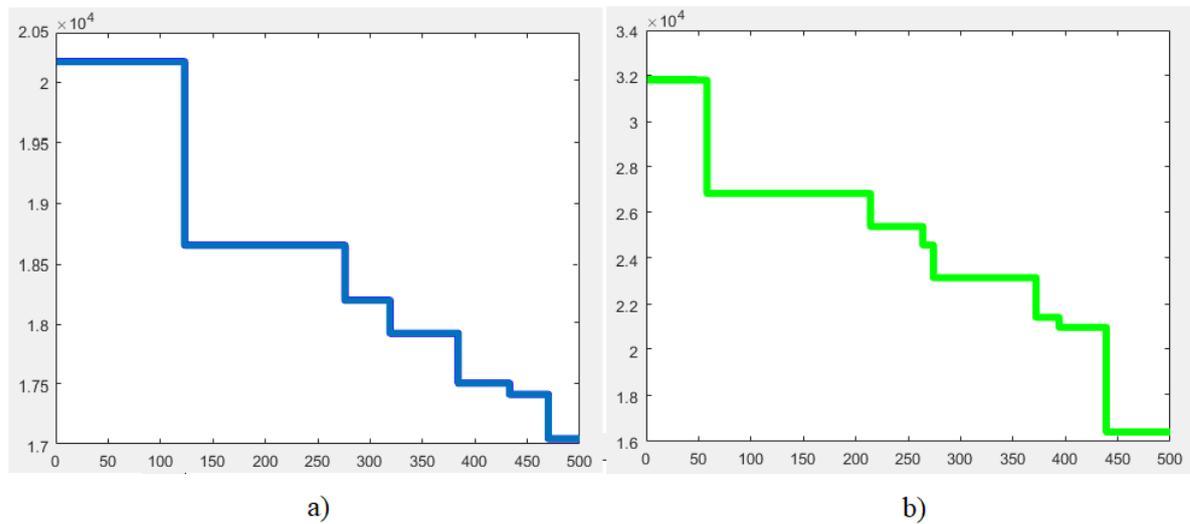
apresentada a melhor solução obtida por cada um, sendo a saída de potência P_i (em MW) para cada uma das unidades geradoras de energia termoelétrica e a soma das potências ($\sum_{i=1}^n P_i(MW)$).

Como podemos observar na Tabela 7.7, o método ED-BFGS mostra-se eficiente para este caso, conseguindo uma solução melhor que o método ED clássico e o método proposto em Silva (2014). A Tabela 7.8 mostra que todos os valores de geração estão dentro dos limites solicitados, e também que a demanda do sistema foi atendida.

Na Figura 7.3 são apresentados os gráficos de convergência para a melhor solução obtida pelos métodos ED e ED-BFGS. Nos gráficos de convergência, o eixo x representa o número de gerações executadas e o eixo y representa o valor obtido na avaliação da função *fitness*. É observado nos gráficos, que a busca local realizada pelo método BFGS interferiu positivamente no processo de minimização, fazendo com que a convergência para a melhor solução fosse mais rápida. Na Figura 7.3 b), vemos que o método ED-BFGS obtém sua melhor solução antes de 450 gerações, enquanto que o método ED obtém sua melhor solução após 450 gerações, como ilustra a Figura 7.3 a).

A Tabela 7.9 apresenta as melhores soluções obtidas pelo método ED-BFGS com diferentes parâmetros para o fator de mutação F e a probabilidade de cruzamento CR .

Figura 7.3 – a) Convergência da melhor solução do método ED para o caso 3. b) Convergência da melhor solução do método ED-BFGS para o caso 3.



O fator de mutação F foi variado de 0,5 a 1,0 e a probabilidade de cruzamento CR foi variada de 0,1 a 1,0. Foram realizadas 100 simulações e são apresentados os melhores resultados obtidos destas simulações. Na Tabela 7.9 a melhor solução obtida pelo método ED-BFGS está destacada em negrito.

7.4 Caso 4: 40 unidades geradoras

Na Tabela A.4, que se encontra no Apêndice A, são apresentados os coeficientes utilizados nos testes para um sistema com 40 unidades geradoras, extraídos de [Sinha, Chakrabarti e Chattopadhyay \(2003\)](#). Para este caso, foi considerada a potência demandada igual a 10500 MW. O critério de parada adotado para os métodos ED e ED-BFGS foi de 6000 gerações e uma população com 200 indivíduos. O tempo de execução dos algoritmos de ED e ED-BFGS foram, respectivamente, 8,78 (s) e 90,32 (s). Os parâmetros adotados foram $F = 0,8$ e $CR = 0,9$ para a ED, e $F = 6$ e $CR = 1,0$ para a ED-BFGS, determinados por testes numéricos. Foram realizadas 100 simulações com os algoritmos de ED e ED-BFGS e são apresentados os melhores resultados obtidos destas simulações.

Considerando que os dados e a formulação do PDE-PCV utilizados neste trabalho são os mesmos utilizados nos trabalhos de [Le et al. \(2015\)](#), [He, Rao e Huang \(2016\)](#), [Elsayed et al. \(2017\)](#), [Zhao et al. \(2018\)](#) e [Li, Fu e Lu \(2021\)](#), é possível realizar uma comparação entre os mesmos. Os métodos utilizados em cada trabalho são, respectivamente, DEGSA, CTLBO, IRDPSO, MCSA e NCHM. Na Tabela 7.10 é apresentado o custo mínimo obtido por cada um destes métodos, enquanto que a Tabela 7.12 mostra a melhor solução obtida pelos métodos ED, ED-BFGS e por [Zhao et al. \(2018\)](#). As soluções de [Le et al. \(2015\)](#), [He, Rao e Huang \(2016\)](#), [Elsayed et al. \(2017\)](#) e [Li, Fu e Lu \(2021\)](#) não são informadas pelos autores. Na Tabela 7.12, a solução é dada pela saída de potência P_i (em MW) para

cada uma das unidades geradoras de energia termoelétrica e é apresentada a soma das potências ($\sum_{i=1}^n P_i(MW)$).

Tabela 7.10 – Função custo para o PDE-PCV com 40 unidades geradoras.

Método	Custo mínimo (\$)
DEGSA (LE et al., 2015)	121412,55
CTLBO (HE; RAO; HUANG, 2016)	121553,83
IRDPSO (ELSAYED et al., 2017)	121506,040
MCSA (ZHAO et al., 2018)	121412,5355
NCHM (LI; FU; LU, 2021)	121412,5355
ED	121508,0200
ED-BFGS	121490,9216

Tabela 7.11 – Função custo obtida pelo método ED-BFGS variando os parâmetros F e CR para o PDE-PCV com 40 unidades geradoras.

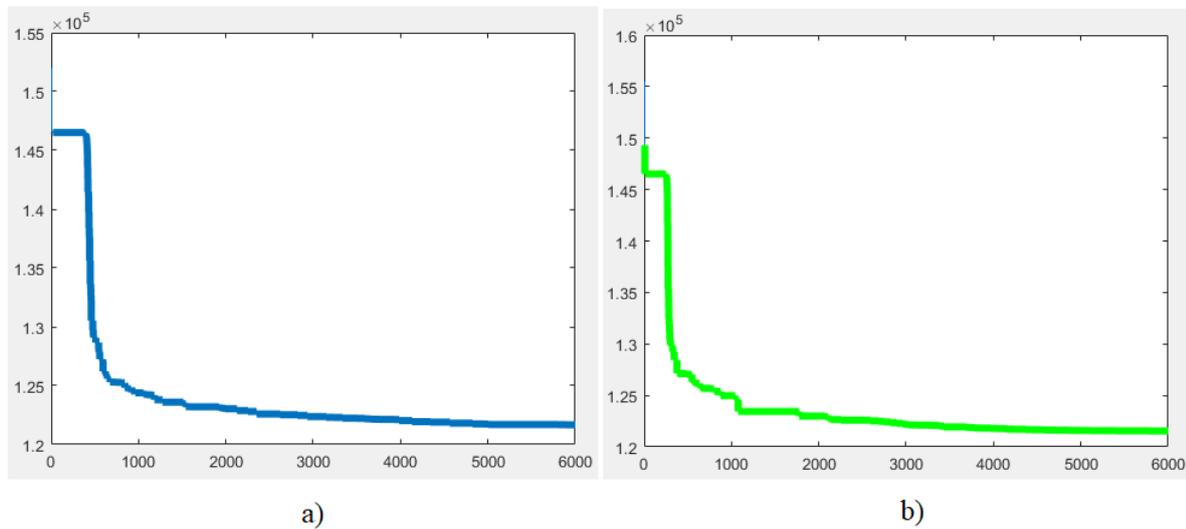
	F = 0,5	F = 0,6	F = 0,7
CR = 0,1	124785,7418	124665,5328	124516,2464
CR = 0,2	124228,8821	124549,1150	124886,8412
CR = 0,3	124527,6951	124428,5087	124867,7717
CR = 0,4	124857,8081	124876,1376	125312,9702
CR = 0,5	125249,7810	125114,4278	125900,6737
CR = 0,6	125666,4013	126144,3509	126138,1986
CR = 0,7	126103,2444	126200,4817	126619,4721
CR = 0,8	126295,0937	126661,7871	127294,9597
CR = 0,9	124933,9005	126620,2478	125738,3375
CR = 1,0	121494,3670	121490,9216	121504,6204
	F = 0,8	F = 0,9	F = 1,0
CR = 0,1	124235,2358	124925,2747	124739,9035
CR = 0,2	124995,7832	124939,2340	124893,6594
CR = 0,3	125090,8869	125328,0784	125527,4692
CR = 0,4	125695,4815	125938,1144	125535,4508
CR = 0,5	126262,3363	126459,4064	126484,5917
CR = 0,6	126549,7322	126783,6834	126346,6248
CR = 0,7	126699,7047	127698,4127	127945,8546
CR = 0,8	127576,1730	127725,8745	128055,4699
CR = 0,9	126895,5443	127039,5297	125026,6531
CR = 1,0	121512,9203	121499,5288	124460,5907

Como podemos observar na Tabela 7.10, o método ED-BFGS mostra-se competitivo para este caso, conseguindo uma solução melhor que o método ED clássico e os métodos apresentados em He, Rao e Huang (2016) e Elsayed et al. (2017). Em relação aos métodos propostos em Le et al. (2015), Li, Fu e Lu (2021) e Zhao et al. (2018), o método ED-BFGS não obteve uma solução superior, mas mostrou-se competitivo com solução próxima a desses métodos. A Tabela 7.12 mostra que todos os valores de geração estão dentro dos limites solicitados, e também que o erro no somatório das potências é quase zero.

Tabela 7.12 – Soluções para o PDE-PCV com 40 unidades geradoras.

Potência (MW)	MCSA	ED	ED-BFGS
P ₁	110,7998	111,5431	110,7998
P ₂	110,7998	111,5431	110,7998
P ₃	97,3999	97,40576	97,39975
P ₄	179,7331	179,7328	179,7236
P ₅	87,7999	96,99827	87,79993
P ₆	140,0000	139,9967	140,0000
P ₇	259,5997	259,6076	259,6003
P ₈	284,5997	284,6067	280,5996
P ₉	284,5997	284,6067	284,5996
P ₁₀	130,0000	204,7986	130,0002
P ₁₁	94,0000	168,8003	168,7977
P ₁₂	94,0000	168,8003	168,7977
P ₁₃	214,7598	304,5197	304,5196
P ₁₄	394,2794	304,5197	304,5196
P ₁₅	394,2794	304,5197	304,5196
P ₁₆	394,27947	304,5197	304,5196
P ₁₇	489,2794	489,2796	489,2163
P ₁₈	489,2794	489,2796	489,2163
P ₁₉	511,2794	511,2791	511,2794
P ₂₀	511,2794	511,2791	511,2794
P ₂₁	523,2794	523,2796	523,2822
P ₂₂	523,2794	523,2796	523,2822
P ₂₃	523,2794	523,2796	523,2822
P ₂₄	523,2794	523,2796	523,2822
P ₂₅	523,2794	523,2796	523,2822
P ₂₆	523,2794	523,2796	523,2822
P ₂₇	10,0000	10,00028	10,00004
P ₂₈	10,0000	10,00028	10,00004
P ₂₉	10,0000	10,00028	10,00004
P ₃₀	87,7999	96,99827	87,79993
P ₃₁	190,0000	189,9999	189,9999
P ₃₂	190,0000	189,9999	189,9999
P ₃₃	190,0000	189,9999	189,9999
P ₃₄	164,7998	164,8031	196,4129
P ₃₅	194,3978	164,8031	196,4129
P ₃₆	200,0000	164,8031	196,4129
P ₃₇	110,0000	109,9995	109,9999
P ₃₈	110,0000	109,9995	109,9999
P ₃₉	110,0000	109,9995	109,9999
P ₄₀	511,2794	511,2791	511,2794
$\sum_{i=1}^n P_i$	10500,00057	10499,9994	10499,9991

Figura 7.4 – a) Convergência da melhor solução do método ED para o caso 4. b) Convergência da melhor solução do método ED-BFGS para o caso 4.



Na Figura 7.4 são apresentados os gráficos de convergência para a melhor solução obtida pelos métodos ED e ED-BFGS. Nos gráficos de convergência, o eixo x representa o número de gerações executadas e o eixo y representa o valor obtido na avaliação da função *fitness*. Na Figura 7.4 b), vemos que o método ED-BFGS encontra sua melhor solução em aproximadamente 4000 gerações, enquanto que o método ED encontra sua melhor solução em aproximadamente 5000 gerações como mostra a Figura 7.4 a). Observando esses gráficos, vemos que as soluções para ambos os algoritmos convergem mais lentamente comparados aos casos anteriores por conta do aumento do número das variáveis. Os problemas de maior dimensionalidade requerem um número maior de gerações até a convergência. Logo, confirmamos a necessidade de um número maior de gerações para a convergência da solução para problemas que possuem maior dimensionalidade e grande espaço de busca.

A Tabela 7.11 apresenta as melhores soluções obtidas pelo método ED-BFGS com diferentes parâmetros para o fator de mutação F e a probabilidade de cruzamento CR . O fator de mutação F foi variado de 0,5 a 1,0 e a probabilidade de cruzamento CR foi variada de 0,1 a 1,0. Foram realizadas 100 simulações com cada combinação de parâmetros e na Tabela 7.11 são apresentados os melhores resultados destas simulações. A melhor solução obtida pelo método ED-BFGS está destacada em negrito.

Considerando os testes realizados com o método ED-BFGS apresentados neste capítulo e as comparações realizadas, conclui-se que o método proposto mostrou-se competitivo obtendo resultados satisfatórios quando comparados com outros resultados da literatura.

No próximo capítulo, serão apresentadas as conclusões deste trabalho.

8 CONCLUSÕES

O algoritmo de ED é uma ferramenta eficiente para resolver problemas complexos de otimização. Tem sido aplicado para resolver vários problemas de diversas áreas. Ele também é usado com eficácia na solução de problemas complexos na área de sistemas de energia, como é o caso do PDE-PCV. Muitos são os trabalhos encontrados na literatura que utilizam o método ED e suas variações para a resolução do PDE-PCV, mostrando sua eficiência. No presente trabalho, em geral a ED proporcionou convergência rápida. No entanto, às vezes esbarra no problema de perder a diversidade da população, o que por sua vez leva a uma convergência prematura. As principais vantagens da ED são sua estrutura simples, velocidade, propriedade de busca global, não exige esforço computacional adicional para a representação binária, como é o caso dos AGs e não requer uma população com grande número de indivíduos.

O método BFGS pode explorar o espaço de busca rapidamente com uma direção de gradiente, mas esta técnica de busca de gradiente local é sensível aos pontos de partida iniciais e sempre fica presa em uma solução ótima local. Outra vantagem do método BFGS está em não precisar recalcular a matriz Hessiana a cada iteração. Assim, a ED e o BFGS apresentam vantagens que se complementam. Para superar as desvantagens e exercer essas vantagens, este trabalho propôs uma abordagem híbrida combinando o método ED com o método BFGS para resolver o PDE-PCV.

Como o PDE-PCV é um problema não convexo e não diferenciável, justifica-se o uso de uma heurística para sua resolução. Basicamente, o algoritmo proposto consiste em duas partes. A primeira parte emprega a ED, a qual propicia uma pesquisa global. A segunda parte emprega o método BFGS para explorar a região de busca localmente e ajustar as soluções da execução da ED, a fim de obter uma solução melhor. Na ED clássica, a cada geração é obtida uma nova população, a qual substitui a antiga, que é descartada. A etapa de seleção gulosa é responsável por selecionar os indivíduos mais aptos e inserí-los na nova população. No método ED-BFGS, a cada geração, a melhor solução da população é utilizada como solução inicial do método BFGS, e esta nova solução é incorporada na população.

Após a implementação do algoritmo do método ED associado ao método BFGS, o mesmo foi aplicado a quatro casos de PDE-PCV contendo 3, 13, 19 e 40 unidades geradoras. Os resultados foram comparados com os resultados obtidos pelo método ED clássico e com outros resultados encontrados na literatura que resolvem os mesmos casos de testes, como [Coelho e Mariani \(2006a\)](#), [Silva \(2014\)](#), [Zou et al. \(2016\)](#), [Secui \(2015\)](#), [Zhao et al. \(2018\)](#) [Le et al. \(2015\)](#), [He, Rao e Huang \(2016\)](#), [Elsayed et al. \(2017\)](#) e [Li, Fu](#)

e Lu (2021).

A principal dificuldade do método proposto foi o elevado tempo computacional para a obtenção da solução nos casos de maior dimensionalidade, como os de 19 e 40 unidades geradoras. Isso acontece pelo fato de a cada geração do método ED, ser realizada uma nova otimização sobre o seu melhor indivíduo através do método BFGS. Entretanto, os resultados numéricos revelam a viabilidade da abordagem híbrida proposta para resolver o PDE-PCV, pois obteve resultados melhores que a ED clássica em todos os casos de testes e outros trabalhos analisados, como os de Coelho e Mariani (2006a), Silva (2014) e Zou et al. (2016) no caso com 3 unidades geradoras; Silva (2014) para o caso com 19 unidades geradoras; e He, Rao e Huang (2016), Elsayed et al. (2017) para o caso com 40 unidades geradoras. Em relação aos resultados que o método ED-BFGS não superou, o mesmo mostrou-se competitivo, obtendo soluções próximas e de boa qualidade.

Concluimos que o método ED-BFGS mostrou-se mais eficiente que o método ED clássico na resolução de todos os casos analisados do PDE-PCV com restrições de balanço de potência e limites de operação das unidades geradoras, e mostrou-se competitivo em relação aos métodos encontrados na literatura que foram abordados neste trabalho.

Em continuidade a este trabalho, destacam-se como propostas futuras: aplicar o método ED-BFGS a outras variantes do PDE, considerando perdas nas linhas de transmissão, zonas de operação proibidas e limites de rampa.

REFERÊNCIAS

ABIDO, M. A. A novel multiobjective evolutionary algorithm for environmental/economic power dispatch. p. 71–81, 2003. Citado na página 21.

ANDRADE, G. L. *Algoritmo evolução diferencial modificado aplicado ao problema do despacho econômico de carga*. Dissertação (Mestrado) — Universidade Federal do Pampa, 2019. Citado na página 22.

ANEEL. *Resolução Normativa n. 414*. [S.l.], 2010. Disponível em: <<http://www2.aneel.gov.br/cedoc/bren2010414.pdf>>. Acesso em: 17 Ago. 2017. Citado na página 21.

BALAMURUGAN, K.; MURALISACHITHNNDAM, R.; KRISHNAN, S. R. Differential evolution based solution for combined economic and emission power dispatch with valve loading effect. *International Journal on Electrical Engineering and Informatics*, v. 6, n. 1, p. 74–92, 2014. Citado 4 vezes nas páginas 81, 86, 87 e 101.

BATISTA, L. S.; GUIMARÃES, F. G.; RAMIREZ, J. A. Differential mutation operator for the archive population of multi-objective evolutionary algorithms. In: *2009 IEEE Congress on Evolutionary Computation*. Trondheim, Noruega: [s.n.], 2009. p. 1108–1115. Citado na página 46.

BAZARAA, M. S.; SHERALI, H. D.; SHETTY, C. M. *Nonlinear programming: theory and algorithms*. [S.l.]: New York: John Wiley & Sons, 1979. Citado 6 vezes nas páginas 19, 32, 34, 35, 40 e 111.

CHIANG, C.-L.; WANG, C.-A. Hybrid differential evolution for cogeneration economic dispatch problem. *International Conference on Machine Learning and Cybernetics*, IEEE, v. 3, p. 1560–1565, 2010. Citado 3 vezes nas páginas 19, 75 e 76.

CHIOU, J.; WANG, F. A hybrid method of differential evolution with application to optimal control problems of a bioprocess system. In: *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*. Anchorage, AK, USA: [s.n.], 1998. p. 627–632. Citado na página 67.

CHIOU, J.-P. Variable scaling hybrid differential evolution for large-scale economic dispatch problems. *Electric Power Systems Research*, Elsevier, v. 77, n. 4, p. 212–218, 2007. Citado 5 vezes nas páginas 19, 65, 66, 67 e 68.

COELHO, L. d. S.; MARIANI, V. C. Evolução diferencial híbrida com programação quadrática aplicada ao problema de despacho econômico de energia elétrica. *Sba: Controle & automação Sociedade Brasileira de Automatica*, SciELO Brasil, v. 17, n. 4, p. 409–423, 2006. Citado 9 vezes nas páginas 19, 22, 23, 63, 64, 81, 82, 93 e 94.

COELHO, L. d. S.; MARIANI, V. C. Otimização de despacho econômico com ponto de válvula usando estratégia evolutiva e método quase-newton. *Learning Nonlinear Models—Revista da Sociedade Brasileira de Redes Neurais (SBRN)*, v. 4, n. 1, p. 1–12, 2006. Citado na página 58.

COLEMAN, T.; BRANCH, M. A.; GRACE, A. *Optimization toolbox. For Use with MATLAB. User's Guide for MATLAB 5, Version 2*. Release II, 1999. Citado 2 vezes nas páginas 23 e 60.

DE-OLIVEIRA, K. W. R. C. B.; NASCIMENTO-JR, N. T.; SAAVEDRA, O. R. Uma abordagem via estratégias evolutivas para o despacho econômico considerando restrições de geração. *IEEE Latin America Transactions*, IEEE, v. 6, n. 1, p. 42–50, 2008. Citado 4 vezes nas páginas 26, 28, 29 e 48.

DORIGO, M.; DI-CARO, G.; GAMBARDILLA, L. M. *Ant algorithms for discrete optimization*. [S.l.]: Artificial life, 1999. v. 5. 137–172 p. Citado na página 45.

DUVVURU, N.; SWARUP, K. A hybrid interior point assisted differential evolution algorithm for economic dispatch. *IEEE Transactions on Power Systems*, IEEE, v. 26, n. 2, p. 541–549, 2011. Citado 4 vezes nas páginas 20, 22, 76 e 77.

EBERHART, R.; KENNEDY, J. Particle swarm optimization. In: *Proceedings of ICNN'95 - International Conference on Neural Networks*. Perth, Australia: [s.n.], 1995. p. 1942–1948. Citado na página 22.

EIBEN, A. E.; SMITH, J. E. *Introduction to evolutionary computing*. [S.l.]: Springer, 2003. v. 53. Citado na página 45.

ELSAYED, W. T. et al. Improved random drift particle swarm optimization with self-adaptive mechanism for solving the power economic dispatch problem. *IEEE Transactions on Industrial Informatics*, v. 13, n. 3, p. 1017–1026, 2017. Citado 6 vezes nas páginas 23, 81, 89, 90, 93 e 94.

FLEMMING, D.; GONÇALVES, M. B. *Cálculo A: Funções, limite, derivação e integração*. [S.l.]: Pearson Education, 2012. Citado na página 104.

FRIEDLANDER, A. *Elementos de programação não-linear*. [S.l.]: Editora da UNICAMP, 1994. Citado 2 vezes nas páginas 38 e 106.

GAING, Z. Particle swarm optimization to solving the economic dispatch considering the generator constraints. *IEEE Transactions on Power Systems*, IEEE, v. 18, n. 3, p. 1187–1195, 2003. Citado 4 vezes nas páginas 25, 28, 29 e 30.

GASPAR-CUNHA, A.; TAKAHASHI, R.; ANTUNES, C. H. *Manual de computação evolutiva e metaheurística*. [S.l.]: Imprensa da Universidade de Coimbra/Coimbra University Press, 2012. Citado 8 vezes nas páginas 43, 44, 45, 46, 47, 51, 52 e 53.

GOMEZ-EXPOSITO, A.; CONEJO, A.; CANIZARES, C. *Differential Evolution: Fundamentals and Applications in Electrical Engineering*. [S.l.]: LTC, 2011. Citado na página 25.

GONÇALVES, E. *Métodos híbridos de pontos interiores/exteriores e de aproximantes de funções em problemas multiobjetivo de despacho econômico e ambiental*. Dissertação (Mestrado) — Universidade Estadual Paulista "Júlio de Mesquita Filho", 2015. Citado 2 vezes nas páginas 21 e 87.

- GONG, W.; CAI, Z. A multiobjective differential evolution algorithm for constrained optimization. In: *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*. Hong Kong, China: [s.n.], 2008. p. 181–188. Citado na página 46.
- HE, D.; WANG, F.; ZHIZHONG, M. A hybrid genetic algorithm approach based on differential evolution for economic dispatch with valve-point effect. *International Journal of Electrical Power & Energy Systems*, Elsevier, v. 30, n. 1, p. 31–38, 2008. Citado 3 vezes nas páginas 19, 70 e 71.
- HE, X.; RAO, Y.; HUANG, J. A novel algorithm for economic load dispatch of power systems. *Neurocomputing*, v. 171, n. 0, p. 1454–1461, 2016. Citado 6 vezes nas páginas 23, 81, 89, 90, 93 e 94.
- HOLLAND, J. *Algoritmos Genéticos*. [S.l.]: Investigación y Ciencia, 1992. v. 45. Citado na página 23.
- ILONEN, J.; KAMARAINEN, J.-K.; LAMPINEN, J. Differential evolution training algorithm for feed-forward neural networks. *Neural Processing Letters*, v. 17, n. 1, p. 93–105, 2003. Citado na página 46.
- JERONYMO, D. C. *Metaheurísticas aplicadas ao problema de despacho econômico de energia elétrica*. Dissertação (Mestrado) — Universidade Federal do Paraná, 2011. Citado na página 22.
- KAPPEL, F.; KUNTSEVICH, A. V. An implementation of shor’s r-algorithm. *Computational Optimization and Applications*, v. 15, n. 2, p. 193–205, 2000. Citado na página 72.
- KIM, H.-K. et al. Differential evolution strategy for constrained global optimization and application to practical engineering problems. *IEEE Transactions on Magnetics*, v. 43, n. 4, p. 1565–1568, 2007. Citado na página 46.
- LE, L. D. et al. Hybrid differential evolution and gravitational search algorithm for nonconvex economic dispatch. *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems*, v. 2, n. 0, p. 89–103, 2015. Citado 4 vezes nas páginas 81, 89, 90 e 93.
- LI, X.; FU, L.; LU, Z. A novel constraints handling mechanism based on virtual generator unit for economic dispatch problems with valve point effects. *International Journal of Electrical Power & Energy Systems*, v. 129, n. 0, p. 106825, 2021. Citado 4 vezes nas páginas 81, 89, 90 e 94.
- LIN, W.; CHENG, F.; TSAY, M. An improved tabu search for economic dispatch with multiple minima. *IEEE Transactions on Power Systems*, IEE, v. 17, n. 1, p. 108–112, 2002. Citado na página 22.
- LUENBERGER, D. G.; YE, Y. *Linear and Nonlinear Programming*. [S.l.]: Springer, 2008. Citado na página 111.
- MICHALSKI, K. A. Electromagnetic imaging of elliptical-cylindrical conductors and tunnels using a differential evolution algorithm. *Neural Processing Letters*, v. 28, n. 3, p. 164–169, 2001. Citado na página 46.

- NOCEDAL, J.; WRIGHT, S. *Numerical optimization*. [S.l.]: Springer, 2006. Citado 2 vezes nas páginas 19 e 34.
- QIN, Q. et al. Solving non-convex/non-smooth economic load dispatch problems via an enhanced particle swarm optimization. *Applied Soft Computing*, v. 59, n. 0, p. 229–242, 2017. Citado na página 61.
- RECHENBERG, I. *Evolutionsstrategien*. [S.l.]: Simulationsmethoden in der Medizin und Biologie, 1978. 83–114 p. Citado na página 45.
- REDDY, M. J.; KUMAR, D. N. Multiobjective differential evolution with application to reservoir system optimization. *Journal of Computing in Civil Engineering*, v. 21, n. 2, p. 136–146, 2007. Citado na página 46.
- REYNOLDS, R. G. An introduction to cultural algorithms. *World Scientific*, v. 24, n. 0, p. 131–139, 1994. Citado na página 64.
- RUZEK, B.; KVASNICKA, M. Differential evolution algorithm in the earthquake hypocenter location. *Pure and Applied Geophysics*, v. 158, n. 4, p. 667–693, 2001. Citado na página 46.
- SAMED, M. M. A. *Um algoritmo genético Híbrido co-evolutivo para resolver problemas de despacho*. Tese (Doutorado) — Universidade Estadual de Maringá, 2004. Citado 3 vezes nas páginas 81, 84 e 100.
- SECUI, D. C. A new modified artificial bee colony algorithm for the economic dispatch problem. *Energy Conversion and Management*, v. 89, n. 0, p. 43–62, 2015. Citado 3 vezes nas páginas 81, 84 e 93.
- SERAPIÃO, A. B. d. S. Fundamentos de otimização por inteligência de enxames: uma visão geral. *Sba: Controle & Automação Sociedade Brasileira de Automática*, SciELO Brasil, v. 20, n. 3, p. 271–304, 2009. Citado 3 vezes nas páginas 21, 22 e 27.
- SILVA, D. N. *Método primal-dual previsor-corretor de pontos interiores e exteriores com estratégias de correção de inércia e suavização hiperbólica aplicado ao problema de despacho econômico com ponto de carregamento de válvula e representação da transmissão*. Dissertação (Mestrado) — Universidade Estadual Paulista "Júlio de Mesquita Filho", 2014. Citado 9 vezes nas páginas 23, 81, 82, 84, 87, 88, 93, 94 e 111.
- SINHA, N.; CHAKRABARTI, R.; CHATTOPADHYAY, R. Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on Evolutionary Computation*, IEEE, v. 7, n. 1, p. 83–94, 2003. Citado 3 vezes nas páginas 81, 89 e 100.
- SINHA, N.; CHATTOPADHYAY, P. K. Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on evolutionary computation*, v. 7, n. 1, p. 83–94, 2003. Citado na página 102.
- STEINBERG, M.; SMITH, T. *Differential Evolution: Fundamentals and Applications in Electrical Engineering*. [S.l.]: J. Wiley & sons, Incorporated, 2011. Citado na página 25.
- STEINBRUCH, A.; WINTERLE, P. *Álgebra Linear*. [S.l.]: Pearson Universidades, 1987. Citado na página 104.

- STORN, R. On the usage of differential evolution for function optimization. In: *Proceedings of North American Fuzzy Information Processing*. Berkeley, USA: [s.n.], 1996. p. 519–523. Citado 3 vezes nas páginas 48, 49 e 50.
- STORN, R. Designing digital filters with differential evolution. In: *New ideas in optimization*. London, UK: [s.n.], 1999. p. 109–126. Citado na página 46.
- STORN, R.; PRICE, K. *Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces*. Berkeley, Brasil, 1995. Citado 7 vezes nas páginas 45, 48, 49, 50, 53, 54 e 55.
- STORN, R.; PRICE, K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, IET Digital Library, v. 11, n. 4, p. 341–359, 1997. Citado 9 vezes nas páginas 22, 23, 47, 48, 49, 53, 54, 55 e 58.
- WALTERS, D. C.; SHEBLE, G. B. Genetic algorithm solution of economic dispatch with valve point loading. *IEEE Transactions on Power Systems*, IEEE, v. 8, n. 3, p. 1325–1332, 1993. Citado 2 vezes nas páginas 22 e 26.
- WANG, S.; CHIOU, J.; LIU, C. Non-smooth/non-convex economic dispatch by a novel hybrid differential evolution algorithm. *IET Generation, Transmission and Distribution*, IET, v. 1, n. 5, p. 793–803, 2007. Citado 4 vezes nas páginas 19, 68, 69 e 70.
- WONG, K.; FUNG, C. Simulated annealing based economic dispatch algorithm. *IEEE proceedings C (generation, transmission and distribution)*, IET Digital Library,, v. 140, n. 6, p. 509–515, 1993. Citado na página 22.
- WOOD, A. J.; WOLLENBERG, B. F.; SHEBLE, G. B. *Power generation, operation and control*. [S.l.]: John Wiley & Sons, 2013. Citado 2 vezes nas páginas 21 e 28.
- YUAN, X. et al. A hybrid differential evolution method for dynamic economic dispatch with valve-point effects. *Expert Systems with Applications*, Elsevier, v. 36, n. 2, p. 4042–4048, 2009. Citado 4 vezes nas páginas 19, 72, 73 e 74.
- ZHANG, Q. et al. An adaptive differential evolutionary algorithm incorporating multiple mutation strategies for the economic load dispatch problem. *Applied Soft Computing*, v. 78, n. 0, p. 641–669, 2019. Citado na página 22.
- ZHAO, J. et al. Modified cuckoo search algorithm to solve economic power dispatch optimization problems. *IEEE/CAA Journal of Automatica Sinica*, v. 5, n. 4, p. 794–806, 2018. Citado 6 vezes nas páginas 23, 81, 84, 89, 90 e 93.
- ZOU, D. et al. An improved differential evolution algorithm for the economic load dispatch problems with or without valve-point effects. *Applied Energy*, v. 181, n. 0, p. 375–390, 2016. Citado 10 vezes nas páginas 20, 22, 23, 77, 79, 80, 81, 82, 93 e 94.

A DADOS DOS PROBLEMAS DE DESPACHO ECONÔMICO

Este apêndice apresenta os dados dos sistemas elétricos utilizados para o PDE-PCV com restrições de balanço de potência e limites de operação das unidades geradoras. Os casos de testes envolvem sistema com 3, 13, 19 e 40 unidades geradoras. As variáveis a_i , b_i e c_i caracterizam os valores dos coeficientes da parte quadrática da função custo, e_i e f_i são os valores que representam os pontos de carregamento de válvula; P_i^{min} e P_i^{max} são os limites operacionais das unidades geradoras do sistema; P_D é a potência total demandada. A unidade calculada no despacho econômico é o custo por hora (\$/h).

A.1 Caso de teste 1: 3 unidades geradoras

Tabela A.1 – Dados para o caso 1.

Potência	$P_i^{min}(MW)$	$P_i^{max}(MW)$	a	b	c	e	f
P ₁	100	600	0,001562	7,92	561	300	0,0315
P ₂	50	200	0,004820	7,97	78	150	0,063
P ₃	100	400	0,001940	7,85	310	200	0,042

Demanda: $P_d=850$ MW

Fonte: [Sinha, Chakrabarti e Chattopadhyay \(2003\)](#).

A.2 Caso de teste 2: 13 unidades geradoras

Tabela A.2 – Dados para o caso 2.

Potência	$P_i^{min}(MW)$	$P_i^{max}(MW)$	a	b	c	e	f
P ₁	0	680	0,00028	8,1	550	300	0,035
P ₂	0	360	0,00056	8,1	309	200	0,042
P ₃	0	360	0,00056	8,1	307	200	0,042
P ₄	60	180	0,00324	7,74	240	150	0,063
P ₅	60	180	0,00324	7,74	240	150	0,063
P ₆	60	180	0,00324	7,74	240	150	0,063
P ₇	60	180	0,00324	7,74	240	150	0,063
P ₈	60	180	0,00324	7,74	240	150	0,063
P ₉	60	180	0,00324	7,74	240	150	0,063
P ₁₀	40	120	0,00284	8,6	126	100	0,084
P ₁₁	40	120	0,00284	8,6	126	100	0,084
P ₁₂	55	120	0,00284	8,6	126	100	0,084
P ₁₃	55	120	0,00284	8,6	126	100	0,084

Demanda: $P_d=2520$ MW

Fonte: [Samed \(2004\)](#).

A.3 Caso de teste 3: 19 unidades geradoras

Tabela A.3 – Dados para o caso 3.

Potência	$P_i^{min}(MW)$	$P_i^{max}(MW)$	a	b	c	e	f
P ₁	100	300	0,0097	6,8	119	90	0,72
P ₂	120	438	0,0055	4	90	79	0,05
P ₃	100	250	0,0055	4	45	0	0
P ₄	8	25	0,0025	0,85	0	0	0
P ₅	50	63,75	0	5,28	0,891	0	0
P ₆	150	300	0,008	3,5	110	0	0
P ₇	50	63,75	0	5,439	21	0	0
P ₈	100	500	0,0075	6	88	50	0,5
P ₉	200	600	0,0085	6	55	0	0
P ₁₀	15	40	0,009	5,2	90	0	0
P ₁₁	50	150	0,0045	1,6	65	0	0
P ₁₂	25	75	0,0025	0,85	78	58	0,02
P ₁₃	50	63,75	0	2,55	49	0	0
P ₁₄	0	95	0,0045	1,6	85	0	0
P ₁₅	20	220	0,0065	4,7	80	92	0,75
P ₁₆	15	80	0,0045	1,4	90	0	0
P ₁₇	15	80	0,0025	0,85	10	0	0
P ₁₈	50	230	0,0045	1,6	25	0	0
P ₁₉	400	500	0,008	5,5	90	0	0

Demanda: $P_d=2908$ MWFonte: [Balamurugan, Muralisachithnndam e Krishnan \(2014\)](#).

A.4 Caso de teste 4: 40 unidades geradoras

Tabela A.4 – Dados para o caso 4.

Potência	$P_i^{min}(MW)$	$P_i^{max}(MW)$	a	b	c	e	f
P ₁	36	114	0,00690	6,73	94,705	100	0,084
P ₂	36	114	0,00690	6,73	94,705	100	0,084
P ₃	60	120	0,02028	7,07	309,54	100	0,084
P ₄	80	190	0,00942	8,18	369,03	150	0,063
P ₅	47	97	0,0114	5,35	148,89	120	0,077
P ₆	68	140	0,01142	8,05	222,33	100	0,084
P ₇	110	300	0,00357	8,03	287,71	200	0,042
P ₈	135	300	0,00492	6,99	391,98	200	0,042
P ₉	135	300	0,00573	6,60	455,76	200	0,042
P ₁₀	130	300	0,00605	12,9	722,82	200	0,042
P ₁₁	94	375	0,00515	12,9	635,20	200	0,042
P ₁₂	94	375	0,00569	12,8	654,69	200	0,042
P ₁₃	125	500	0,00421	12,5	913,40	300	0,035
P ₁₄	125	500	0,00752	8,84	1760,4	300	0,035
P ₁₅	125	500	0,00708	9,15	1728,3	300	0,035
P ₁₆	125	500	0,00708	9,15	1728,3	300	0,035
P ₁₇	220	500	0,00313	7,97	647,85	300	0,035
P ₁₈	220	500	0,00313	7,95	649,69	300	0,035
P ₁₉	242	550	0,00313	7,97	647,83	300	0,035
P ₂₀	242	550	0,00313	7,97	647,81	300	0,035
P ₂₁	254	550	0,00298	6,63	785,96	300	0,035
P ₂₂	254	550	0,00298	6,63	785,96	300	0,035
P ₂₃	254	550	0,00284	6,66	794,53	300	0,035
P ₂₄	254	550	0,00284	6,66	794,53	300	0,035
P ₂₅	254	550	0,00277	7,10	801,32	300	0,035
P ₂₆	254	550	0,00277	7,10	801,32	300	0,035
P ₂₇	10	150	0,52124	3,33	1055,1	120	0,077
P ₂₈	10	150	0,52124	3,33	1055,1	120	0,077
P ₂₉	10	150	0,52124	3,33	1055,1	120	0,077
P ₃₀	47	90	0,01140	5,35	148,89	120	0,77
P ₃₁	60	190	0,00160	6,43	222,92	150	0,063
P ₃₂	60	190	0,00160	6,43	222,92	150	0,063
P ₃₃	60	190	0,00160	6,43	222,92	150	0,063
P ₃₄	90	200	0,0001	8,95	107,87	200	0,042
P ₃₅	90	200	0,0001	8,62	116,58	200	0,042
P ₃₆	90	200	0,001	8,95	107,87	200	0,042
P ₃₇	25	110	0,0161	5,88	307,45	80	0,098
P ₃₈	25	110	0,0161	5,88	307,45	80	0,098
P ₃₉	25	110	0,0161	5,88	307,45	80	0,098
P ₄₀	242	550	0,00313	7,97	647,83	300	0,035

Demanda: $P_d=10500$ MWFonte: [Sinha e Chattopadhyay \(2003\)](#).

B REVISÃO TEÓRICA DE OTIMIZAÇÃO

B.1 Conceitos básicos

Este apêndice apresenta alguns conceitos básicos necessários para a compreensão da abordagem de otimização proposta neste trabalho.

B.1.1 Matriz Hessiana

Seja $f(x)$ continuamente diferenciável de segunda ordem, o que denominaremos por $f \in C^2$. A matriz Hessiana é uma matriz quadrada formada pelas derivadas parciais de segunda ordem de $f(x)$ em relação às coordenadas x_i , avaliadas em um ponto particular x , e é dada por:

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial^2 x_n} \end{pmatrix} \quad (\text{B.1})$$

B.1.2 Matriz definida positiva ou semidefinida positiva

Uma matriz simétrica A de ordem n é definida positiva se, para todo vetor $x \in \mathbb{R}^n$, com $x \neq 0$, temos:

$$x^T A x > 0, \quad (\text{B.2})$$

ou semidefinida positiva se:

$$x^T A x \geq 0. \quad (\text{B.3})$$

Outra maneira de verificarmos se a matriz A é definida positiva, semidefinida positiva ou nenhum dos casos, é verificando se seus menores principais são positivos ou não negativos.

B.1.3 Matriz definida negativa ou semidefinida negativa

Uma matriz simétrica A de ordem n é definida negativa se, para todo vetor $x \in \mathbb{R}^n$, com $x \neq 0$, temos:

$$x^T Ax < 0, \quad (\text{B.4})$$

ou semidefinida negativa se:

$$x^T Ax \leq 0. \quad (\text{B.5})$$

B.1.4 Matriz singular ou não singular

Uma matriz quadrada $A = [a_{ij}]$ cujo determinante é nulo é uma matriz singular. Outra característica de uma matriz singular, é que esta não possui inversa. Por outro lado, se o determinante desta matriz for diferente de zero, dizemos que ela é uma matriz não singular ou regular, admitindo inversa sempre ([STEINBRUCH; WINTERLE, 1987](#)).

B.1.5 Fórmula de Taylor

A fórmula de Taylor consiste num método de aproximação de uma função por um polinômio, com um erro possível de ser estimado.

Definição: Seja $f : I \rightarrow \mathbb{R}$ uma função que admite derivadas até ordem n num ponto c do intervalo I . O polinômio de Taylor de ordem n de f no ponto c , que denotamos por $P_n(x)$, é dado por ([FLEMMING; GONÇALVES, 2012](#)):

$$P_n(x) = f(c) + f'(c)(x - c) + \frac{f''(c)}{2!}(x - c)^2 + \cdots + \frac{f^{(n)}(c)}{n!}(x - c)^n. \quad (\text{B.6})$$

Notamos que no ponto $x = c$, $P_n(c) = f(c)$.

B.1.6 Funções convexas

Funções convexas definidas sobre conjuntos convexos possuem propriedades importante para a teoria da otimização, sendo que uma delas é garantia de que todo mínimo local também é mínimo global.

Uma função $f : A \rightarrow \mathbb{R}$ é dita convexa se $f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$, $\forall x, y \in A$, $\forall \alpha \in [0, 1]$ e $A \subseteq \mathbb{R}^n$ um conjunto convexo. Se esta condição for satisfeita, então o mínimo local de f também é mínimo global.

Nesta seção será discutido sobre a otimização não linear e serão apresentados os conceitos de otimização irrestrita e restrita, fundamentais para resolução do problema de despacho econômico com ponto de carregamento de válvula (PDE-PCV) em destaque neste trabalho.

B.2 Otimização não linear

Analistas, engenheiros, gerentes e planejadores trabalham em seu dia a dia com problemas que precisam resolver. Esses problemas podem envolver alocação de recursos escassos, previsão e planejamento industrial, automação e otimização de processos, entre outros, e sempre requerem uma tomada de decisão. Para a tomada de decisão, são definidas variáveis de decisão, é formulado um objetivo a ser alcançado levando em conta as limitações ou restrições, caso elas existam, para depois escolher o curso de ação ótimo, isto é, aquele que conduz a melhor solução. Se o objetivo puder ser escrito como função das variáveis de decisão e as limitações representadas por um conjunto de equações e inequações que envolvam as variáveis de decisão temos um modelo matemático para o problema. Na linha de resolução de problemas deste tipo está a otimização, que tem por objetivo encontrar a melhor solução, entre todas as soluções em potencial, de um determinado problema.

Os problemas de otimização em que a função objetivo e/ou uma das restrições são não lineares são denominados de problemas de otimização não linear restritos. Diferentemente da otimização linear, a otimização não linear pode trabalhar com um conjunto de restrições não convexas, e não tem-se a garantia de que o ponto ótimo esteja em um vértice ou aresta da região factível, ou de que um ponto ótimo local também é um ponto ótimo global, quando da ausência da convexidade. Sendo assim, há uma vasta área de pesquisa em relação à métodos eficientes para solução destes problemas.

Os problemas de programação não linear podem ser irrestritos ou restritos. No caso irrestrito, tem-se apenas a função objetivo e não há nenhuma restrição. Já no restrito, como o próprio nome diz, tem-se a função objetivo e as restrições, as quais definem o conjunto de soluções viáveis (factíveis). Nem sempre podemos contar com a resolução gráfica como sendo um recurso, pois a maioria dos modelos possui um número elevado de variáveis (nem nos lineares). Em programação não linear não existe um método geral de resolução de seus problemas, de modo que os métodos são aplicados conforme as características das funções envolvidas, tais como continuidade, unimodalidade, diferenciabilidade de primeira e/ou segunda ordem, entre outras.

B.2.1 Otimização irrestrita

Para a resolução de um problema de programação não linear pode-se utilizar métodos iterativos, os quais geram uma seqüência de pontos que representaremos por

x_0, x_1, \dots, x_k que idealmente convergirão para um ponto desejável x^* , o qual é a solução do problema.

Na otimização irrestrita, visa-se minimizar uma função de variáveis reais a um valor real sem que haja restrições aos valores das variáveis. A forma genérica dos problemas de programação não linear ou otimização é dada por (FRIEDLANDER, 1994):

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } x \in \aleph \end{aligned} \tag{B.7}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ e \aleph é chamado conjunto factível.

Como iremos trabalhar com problemas de minimização representados por B.7, a solução poderá ser um ponto de mínimo local ou um ponto de mínimo global. A definição desses pontos é descrita a seguir de acordo com Friedlander (1994).

- **Mínimo local:** Um ponto $x^* \in \aleph$ é um mínimo local de f em \aleph se, e somente se, existe $\epsilon > 0$ tal que $f(x) \geq f(x^*)$ para todo $x \in \aleph$ tal que $\|x - x^*\| < \epsilon$. Se $f(x) > f(x^*)$ para todo $x \in \aleph$ tal que $x \neq x^*$ e $\|x - x^*\| < \epsilon$, diremos que se trata de um minimizador local estrito em \aleph .
- **Mínimo global:** Um ponto $x^* \in \aleph$ é um mínimo global de f em \aleph se, e somente se, $f(x) \geq f(x^*)$ para todo $x \in \aleph$. Se $f(x) > f(x^*)$ para todo $x \in \aleph$ tal que $x \neq x^*$, diremos que se trata de um minimizador global estrito em \aleph .

Os pontos de máximo local e máximo global são definidos de maneira análoga.

B.2.2 Condições de otimalidade para otimização irrestrita

As condições de otimalidade são relações entre as derivadas da função objetivo e as derivadas das funções que definem as restrições. São elas que dizem se as derivadas envolvidas permitirão a redução do valor da função objetivo. Qualquer condição satisfeita por uma solução é uma condição de otimalidade. Encontrar um ponto que cumpra apenas uma das condições não garante que este seja a solução do problema, pois há muitos pontos viáveis que não são solução. Por isso, para um ponto ser a solução de um problema ele deve satisfazer todas as condições de otimalidade descritas a seguir (FRIEDLANDER, 1994).

- **Condições necessárias de primeira ordem:** Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^1$. Se x^* é um ponto de mínimo local de f em \mathbb{R}^n , então $\nabla f(x^*) = 0$.

- **Condições necessárias de segunda ordem:** Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^2$. Se x^* é um ponto de mínimo local de f em \mathbb{R}^n , então
 - 1) $\nabla f(x^*) = 0$;
 - 2) $\nabla^2 f(x^*)$ é semidefinida positiva.

Como dito anteriormente, um ponto satisfazer apenas as condições necessárias, não garante que o mesmo seja um ponto de mínimo local para o problema em questão. Já as condições suficientes de segunda ordem apresentados a seguir garantem que um ponto que as satisfaz seja um ponto de mínimo local.

- **Condições suficientes de segunda ordem:** Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^2$. Se $x^* \in \mathbb{R}^n$, $\nabla f(x^*) = 0$ e $\nabla^2 f(x^*)$ é definida positiva, então x^* é um ponto de mínimo local estrito de f em \mathbb{R}^n .

B.2.3 Otimização restrita

Em otimização restrita, queremos minimizar uma função de variáveis reais a um valor real com restrições aos valores das variáveis.

B.2.3.1 Restrição de igualdade

Um problema de otimização no qual há apenas restrições de igualdade pode ser escrito, genericamente como:

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } g(x) = 0 \end{aligned} \tag{B.8}$$

em que:

$g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, representa o conjunto das restrições de igualdade do problema;
 $x \in \Omega \subseteq \mathbb{R}^n$, sendo Ω um conjunto aberto.

Relacionada a função g no ponto x , temos a matriz Jacobiana definida como:

$$Jg(x) = \begin{pmatrix} \frac{\partial g_1}{\partial x_1}(x) & \dots & \frac{\partial g_1}{\partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial g_m}{\partial x_1}(x) & \dots & \frac{\partial g_m}{\partial x_n}(x) \end{pmatrix} = \begin{pmatrix} \nabla g_1(x)^t \\ \vdots \\ \nabla g_m(x)^t \end{pmatrix}. \tag{B.9}$$

O Teorema dos Multiplicadores de Lagrange apresentado a seguir é o equivalente às condições necessárias de primeira ordem, considerando restrições de igualdade.

Teorema B.1. (Teorema dos Multiplicadores de Lagrange): Seja x^* um mínimo local regular para o problema (B.8). Então existem únicos $\lambda_1^*, \dots, \lambda_m^*$ números reais, denominados multiplicadores de Lagrange, tais que $\nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) = 0$.

O Teorema B.2 é equivalente às condições necessárias de segunda ordem.

Teorema B.2. Suponha que $f, g \in C^2$, x^* é mínimo regular para o problema (B.8), que λ^* é o vetor de multiplicadores de Lagrange, cuja existência é garantida pelo Teorema B.1. Então $v^t (\nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(x^*)) v \geq 0$, para todo $v \in N(Jg(x^*))$.

Em que:

$N(A)$ é o núcleo da matriz A , e $Jg(x^*)$ é a matriz jacobiana de g calculada em x^* .

As condições suficientes de otimalidade também podem ser introduzidas para o caso com restrições de igualdade, como é apresentado no Teorema B.3

Teorema B.3. Sejam $f, g \in C^2$, $x^* \in \aleph$ e $\lambda^* \in \mathbb{R}^m$ tais que $\nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) = 0$.

Em que:

$\aleph = \{x \in \Omega : g(x) = 0\}$ é o conjunto factível do problema.

Considere ainda que a matriz $\nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(x^*)$ é definida positiva em $N(Jg(x^*))$. Então, x^* é um ponto de mínimo local estrito para o problema (B.8).

B.2.3.2 Restrição de desigualdade

Ao problema (B.8) também podem ser inseridas restrições de desigualdade e sua formulação é dada por:

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } h(x) \leq 0 \end{aligned} \tag{B.10}$$

em que:

$h : \mathbb{R}^n \rightarrow \mathbb{R}^p$, representa o conjunto das restrições de desigualdade do problema;
 $x \in \Omega \subseteq \mathbb{R}^n$, sendo Ω um conjunto aberto.

Para o caso do problema com restrições de desigualdade, as condições necessárias de otimalidade estão associadas ao teorema de Fritz-John, que é apresentado a seguir:

Teorema B.4. (Condições Necessárias de Fritz-John): Considere o problema (B.10). Seja x^* um ponto factível tal que f e h_j são diferenciáveis em x^* , para $j \in A(x^*)$

(conjunto das restrições ativas), e h_j é contínua em x^* , para $j \in A(x^*)$. Se x^* é um mínimo local, então existem escalares v_0^* e v_j^* para cada $j \in A(x^*)$, tais que

$$v_0^* \nabla f(x^*) + \sum_{j \in A(x^*)} v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.11})$$

$$v_0^*, v_j^* \geq 0, \text{ para } j \in A(x^*) \quad (\text{B.12})$$

$$(v_0^*, v_{A(x^*)}^*) \neq (0, 0) \quad (\text{B.13})$$

em que $v_{A(x^*)}^*$ é o vetor cujas componentes são v_j^* , com $j \in A(x^*)$. Se, adicionalmente, h_j for diferenciável em x^* , para todo $j \notin A(x^*)$, as condições supracitadas podem ser escritas da seguinte forma:

$$v_0^* \nabla f(x^*) + \sum_{j=1}^p v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.14})$$

$$v_j^* h_j(x^*) = 0, \text{ para } j = 1, \dots, p \quad (\text{B.15})$$

$$v_0^*, v_j^* \geq 0, \text{ para } j = 1, \dots, p \quad (\text{B.16})$$

$$(v_0^*, v^*) \neq (0, 0) \quad (\text{B.17})$$

em que v^* é o vetor cujas componentes são v_j^* , para $j = 1, \dots, p$.

De acordo com as condições necessárias de Fritz-John, pode ser enunciado o teorema com as condições necessárias de Karush-Kuhn-Tucker (KKT), de imensa importância na área de otimização não linear:

Teorema B.5. (Condições necessárias de Karush-Kuhn-Tucker): Considere o problema (B.10). Seja x^* um ponto factível tal que f e h_j são diferenciáveis em x^* , para $j \in A(x^*)$, e h_j é contínua em x^* , para $j \notin A(x^*)$. Suponha, ainda, que o conjunto $\{\nabla h_j(x^*)\}_{j \in A(x^*)}$ é linearmente independente (isto é, x^* é um ponto regular). Se x^* é um mínimo local, então existe escalares v_j^* , para cada $j \in A(x^*)$, tais que

$$\nabla f(x^*) + \sum_{j \in A(x^*)} v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.18})$$

$$v_j^* \geq 0, \text{ para } j \in A(x^*). \quad (\text{B.19})$$

Se, adicionalmente, h_j for diferenciável em x^* , para todo $j \notin A(x^*)$, as condições supracitadas podem ser escritas da seguinte forma:

$$\nabla f(x^*) + \sum_{j=1} v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.20})$$

$$v_j^* h_j(x^*) = 0, \text{ para } j = 1, \dots, p \quad (\text{B.21})$$

$$v_j^* \geq 0, \text{ para } j = 1, \dots, p \quad (\text{B.22})$$

B.2.3.3 Restrições de igualdade e desigualdade

Os problemas de otimização podem ser formulados considerando tanto as restrições de igualdade quanto restrições de desigualdade, tendo assim, um problema de otimização no formato geral. Esses problemas são definidos da seguinte forma:

$$\begin{aligned} & \text{minimizar } f(x) \\ & \text{sujeito a: } g(x) = 0 \\ & \quad \quad \quad h(x) \leq 0 \end{aligned} \quad (\text{B.23})$$

em que:

$$f : \mathbb{R}^n \rightarrow \mathbb{R};$$

$$g : \mathbb{R}^n \rightarrow \mathbb{R}^m \text{ e representa o conjunto das restrições de igualdade;}$$

$$h : \mathbb{R}^n \rightarrow \mathbb{R}^p \text{ e representa o conjunto das restrições de desigualdade;}$$

$x \in \Omega \subseteq \mathbb{R}^n$, sendo Ω um conjunto aberto.

As condições necessárias de otimalidade para o problema descrito acima são dadas pelo Teorema B.6.

Teorema B.6. (Condições necessárias de Karush-Kuhn-Tucker): *Seja x^* um ponto factível tal que f , g e h_j são diferenciáveis em x^* , para $j \in A(x^*)$, e h_j é contínua em x^* , para $j \notin A(x^*)$. Suponha, ainda, que o conjunto $\{\nabla g_1(x^*), \dots, \nabla g_m(x^*)\} \cup \{\nabla h_j(x^*)\}_{j \in A(x^*)}$ é linearmente independente (isto é, x^* é um ponto regular). Se x^* é um mínimo local, então existem escalares unicamente determinados λ_i^* , para $i = 1, \dots, m$ e v_j^* para cada $j \in A(x^*)$, tais que*

$$\nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + \sum_{j \in A(x^*)} v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.24})$$

$$v_j^* \geq 0, \text{ para } j \in A(x^*). \quad (\text{B.25})$$

Se, adicionalmente, h_j for diferenciável em x^* , para todo $j \notin A(x^*)$, as condições supracitadas podem ser escritas da seguinte forma:

$$\nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^p v_j^* \nabla h_j(x^*) = 0 \quad (\text{B.26})$$

$$v_j^* h_j(x^*) = 0, \text{ para } j = 1, \dots, p \quad (\text{B.27})$$

$$v_j^* \geq 0, \text{ para } j = 1, \dots, p \quad (\text{B.28})$$

Pot fim, se as funções f , g e h forem de classe C^2 , e $V(x^*)$ for a matriz cujas linhas são os gradientes do conjunto $\{\nabla g_1(x^*), \dots, \nabla g_m(x^*)\} \cup \{\nabla h_j(x^*)\}_{j \in A(x^*)}$, então

$$v^t \left(\nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(x^*) + \sum_{j=1}^p v_j^* \nabla^2 h_j(x^*) \right) v \geq 0, \text{ para todo } v \in N(V(x^*)) \quad (\text{B.29})$$

isto é, a matriz $\nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(x^*) + \sum_{j=1}^p v_j^* \nabla^2 h_j(x^*)$ é semidefinida positiva em $N(V(x^*))$.

Como nos casos de igualdade, λ^* e v^* são denominados multiplicadores de Lagrange.

Teorema B.7. (Condições suficientes de Karush-Kuhn-Tucker): Suponha que as funções f , g e h são de classe C^2 . Suponha que para um dado ponto x^* , existam valores $\lambda^* = (\lambda_1^*, \dots, \lambda_m^*)^t$ e $v^* = (v_1^*, \dots, v_p^*)^t$ satisfazendo as equações (B.26), (B.27) e (B.28). Defina $A^+(x^*) = \{j \in A(x^*) : v_j^* > 0\}$, $A^0(x^*) = \{j \in A(x^*) : v_j^* = 0\}$ e o cone

$$S(x^*) = \left\{ d \in \mathbb{R}^n : d \neq 0, \nabla g_i(x^*)^t d = 0, \text{ para } i = 1, \dots, m, \right. \\ \left. \nabla h_j(x^*)^t d = 0, \text{ para } j \in A^+(x^*), \nabla h_j(x^*)^t d \leq 0, \text{ para } j \in A^0(x^*). \right.$$

Se $v^t \left(\nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(x^*) + \sum_{j=1}^p v_j^* \nabla^2 h_j(x^*) \right) v > 0$, para todo $v \in S(x^*)$, então x^* é um ponto de mínimo local estrito do problema (B.23).

Todas as demonstrações dos teoremas apresentados neste capítulo podem ser encontradas no trabalho de Silva (2014), que apoiou-se nos trabalhos de Luenberger e Ye (2008) e Bazaraa, Sherali e Shetty (1979).