

Laboratório Nacional de Computação Científica
Programa de Pós Graduação em Modelagem Computacional

**Evolução Diferencial para Problemas de Otimização
Restrita**

Por
Eduardo Krempser da Silva

PETRÓPOLIS, RJ - BRASIL

MARÇO DE 2009

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

**EVOLUÇÃO DIFERENCIAL PARA PROBLEMAS DE
OTIMIZAÇÃO RESTRITA**

Eduardo Krempser da Silva

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO LABORATÓRIO
NACIONAL DE COMPUTAÇÃO CIENTÍFICA COMO PARTE DOS REQUI-
SITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM
CIÊNCIAS EM MODELAGEM COMPUTACIONAL

Aprovada por:

Prof. Helio José Corrêa Barbosa, D.Sc
(Presidente)

Prof. Laurent Emmanuel Dardenne, D.Sc.

Prof. Nelson Francisco Favilla Ebecken, D.Sc.

PETRÓPOLIS, RJ - BRASIL
MARÇO DE 2009

Silva, Eduardo Krempser da

S586e Evolução diferencial para problemas de otimização restrita / Eduardo Krempser da Silva. Petrópolis, RJ. : Laboratório Nacional de Computação Científica, 2009.

xv, 100 p. : il.; 29 cm

Orientador: Helio José Corrêa Barbosa

Dissertação (M.Sc.) – Laboratório Nacional de Computação Científica, 2009.

1. Otimização. 2. Evolução diferencial. 3. Metaheurística. I. Barbosa, Helio José Corrêa. II. LNCC/MCT. III. Título.

CDD 006.33

“Tudo tem seu tempo e até certas
manifestações mais vigorosas e originais
entram em voga ou saem de moda. Mas a
sabedoria tem uma vantagem: é eterna.”

(Baltasar Gracián)

Aos meus pais, irmãos e noiva.

Agradecimentos

Diversas pessoas contribuíram para a realização deste trabalho, e eu agradeço a todas. Entretanto faço um especial agradecimento:

Ao meu orientador, Helio, por ser referência para todo o meu caminho acadêmico;

Ao prof. Lemonge, pelo importante auxílio nos trabalhos realizados;

À minha noiva, Daniele, pelo amor e carinho, além de ser fundamental em todos os momentos;

Aos meus irmãos, Fernanda, Luciana, Leonardo e Rafael, os quais são grande fonte de motivação e orgulho;

Aos meus pais, Carmem e José Carlos, os quais devo tudo que sei e sou;

Aos amigos Paulo, Leonardo, Odázio e Roselaine, pela simples amizade;

Ao Laboratório Nacional de Computação Científica (LNCC), pelas oportunidades;

Aos amigos e funcionários do LNCC, pelo apoio constante;

Ao CNPq, pelo apoio financeiro.

Resumo da Dissertação apresentada ao LNCC/MCT como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

EVOLUÇÃO DIFERENCIAL PARA PROBLEMAS DE OTIMIZAÇÃO RESTRITA

Eduardo Krempser da Silva

Março , 2009

Orientador: Helio José Corrêa Barbosa, D.Sc

A otimização é uma grande área de conhecimento voltada para a necessidade de um melhor aproveitamento de recursos e atividades, tornando-se indispensável na resolução de grande parte dos problemas oriundos de estudos e formulações de problemas reais. Além disso, as restrições que devem ser respeitadas para cada situação introduzem nas metodologias de otimização um complicador adicional. A Evolução Diferencial, que em sua formulação original é aplicada somente a problemas de otimização irrestrita e em espaços contínuos, apresenta também bons resultados quando aplicada à otimização restrita com variáveis contínuas e discretas. Este trabalho apresenta os aperfeiçoamentos necessários à Evolução Diferencial para sua adequada aplicação sobre essa classe de problemas, além de propor uma nova combinação de técnicas para essa aplicação, bem como um mecanismo de seleção dinâmica da variante adequada da técnica. A proposta inicial é a combinação da Evolução Diferencial com uma técnica adaptativa de penalização (APM) e a segunda proposta visa a seleção dinâmica de variantes durante o processo de busca. Vários experimentos computacionais são executados confirmando a competitividade dos algoritmos propostos.

Abstract of Dissertation presented to LNCC/MCT as a partial fulfillment of the requirements for the degree of Master of Sciences (M.Sc.)

DIFFERENTIAL EVOLUTION FOR CONSTRAINED OPTIMIZATION PROBLEMS

Eduardo Krempser da Silva

March, 2009

Advisor: Helio José Corrêa Barbosa, D.Sc

Optimization is a large area of knowledge concerned with the need of a better use of resources and activities, becoming indispensable in the solution of several problems which arise from the study and formulation of real-world problems. Furthermore, the constraints that must be respected for each situation introduce in the methodologies of optimization an additional complication. Differential Evolution, which in its original formulation is applied only to unconstrained optimization problems in continuous space, also provides good results when applied to constrained optimization with discrete and continuous variables. This work presents the necessary improvements to Differential Evolution for its proper application to this class of problems, and proposes a new combination of techniques for this application, as well as a mechanism for dynamic selection of the appropriate variant of the technique. The initial proposal is a combination of Differential Evolution with a technique of adaptive penalty (APM) and the second proposal concerns the dynamic selection of variants during the search process. Several computational experiments are carried out confirming the competitiveness of the proposed algorithms.

Sumário

1	Introdução	1
2	Conceitos Básicos de Otimização	3
2.1	Otimização	3
2.1.1	Convexidade	5
2.1.2	Otimização Restrita	7
2.2	Metaheurísticas	11
2.2.1	Busca Local Estocástica	11
2.2.2	Simulated Annealing	14
2.2.3	Algoritmos Evolucionistas	16
3	Evolução Diferencial	19
3.1	Histórico	20
3.2	Descrição do Algoritmo	21
3.2.1	Um Exemplo Simples	25
3.3	Variantes	28
3.4	Aplicação em Problemas Mistos	30
3.5	Critérios de Seleção e Tratamento de Restrições	31
3.5.1	Critério de Seleção de Deb	32
3.5.2	Penalização	32
3.5.3	A Técnica APM (Adaptive Penalty Method)	35
3.5.4	A Técnica de Ordenação Estocástica (Stochastic Ranking)	37

3.5.5	Tratamento dos Limites das Variáveis	38
3.6	Configuração dos Parâmetros	39
3.7	Seleção da Variante	41
3.8	Implementações Propostas	42
3.8.1	DE+APM	42
3.8.2	DUVDE+APM	43
4	Experimentos Numéricos	46
4.1	Objetivos e Organização	46
4.2	Grupo de Problemas	48
4.3	A Viga Soldada	61
4.4	O Vaso de Pressão	65
4.5	Treliça de Dez Barras	68
4.5.1	Caso Contínuo	69
4.5.2	Caso Discreto	69
4.6	Treliça de Vinte e Cinco Barras	73
4.6.1	Caso Contínuo	73
4.6.2	Caso Discreto	75
4.7	Treliça de Sessenta Barras	78
4.8	Treliça de Setenta e Duas Barras	80
5	Conclusão	87
	Referências Bibliográficas	89
	Apêndice	
A	Otimização Estrutural	95
A.1	Exemplo de otimização estrutural	96
A.1.1	Determinação das restrições	97
A.1.2	Determinação dos deslocamentos e tensões	98

Lista de Figuras

Figura

2.1	Função estritamente convexa, convexa e não-convexa (Friedlander, 1994)	6
2.2	Descrição de um algoritmo evolucionista	18
3.1	Esquema de geração do vetor teste (“trial”) - Sendo N_p o número de vetores na população	24
3.2	Função $f(x, y) = x^2 + y^2$	26
3.3	População sobre a função $f(x, y) = x^2 + y^2$	26
3.4	Operação de mutação	28
4.1	Experimento 2 - Viga Soldada. A carga aplicada na extremidade da viga é $F = 6000$ lb	62
4.2	Experimento 2 - Viga Soldada. Decaimento médio da função objetivo.	64
4.3	Experimento 3 - Vaso de pressão	66
4.4	Experimento 3 - Vaso de Pressão. Decaimento médio da função objetivo.	67
4.5	Experimento 4 - Treliça de 10 barras	68
4.6	Experimento 4 - Treliça de 10 Barras, caso contínuo. Decaimento médio da função objetivo.	70
4.7	Experimento 4 - Treliça de 10 Barras, caso discreto. Decaimento médio da função objetivo.	72
4.8	Experimento 5 - Treliça de 25 barras	74

4.9	Experimento 5 - Treliça de 25 Barras, caso contínuo. Decaimento médio da função objetivo	76
4.10	Experimento 5 - Treliça de 25 Barras, caso discreto. Decaimento médio da função objetivo	77
4.11	Experimento 6 - Treliça anelar de 60 barras	79
4.12	Experimento 6 - Treliça de 60 Barras. Decaimento médio da função objetivo.	82
4.13	Experimento 7 - Treliça 72 barras	83
4.14	Experimento 7 - Treliça de 72 Barras. Decaimento médio da função objetivo	86
A.1	Treliça de três barras	96
A.2	Possível configuração da treliça de três barras após aplicação do carregamento	97

Lista de Tabelas

Tabela

3.1	Nomenclaturas utilizadas na DE.	24
4.1	Definição das funções teste	49
4.2	Definição dos limites das variáveis para as funções da G-Suite	50
4.3	Resumo das 11 funções da G-Suite	50
4.4	Resultado da primeira etapa do experimento 1, $G_1 - G_5$	52
4.5	Resultado da primeira etapa do experimento 1, $G_6 - G_{11}$	53
4.6	Resultado da segunda etapa do experimento 1, $G_1 - G_{11}$	55
4.7	Resultado da terceira etapa do experimento 1, $G_1 - G_{11}$	56
4.8	Resultado da quarta etapa do experimento 1, $G_1 - G_4$	58
4.9	Resultado da quarta etapa do experimento 1, $G_5 - G_8$	59
4.10	Resultado da quarta etapa do experimento 1, $G_9 - G_{11}$	60
4.11	Resultado do experimento 2, Viga Soldada	63
4.12	Resultado do experimento 3, Vaso de Pressão	67
4.13	Resultado do experimento 4, Treliça de 10 Barras - Caso Contínuo	69
4.14	Melhores soluções encontradas para o experimento 4 - Caso Contínuo, Treliça de 10 Barras	70
4.15	Resultado do experimento 4, Treliça de 10 Barras - Caso Discreto	71
4.16	Melhor solução encontrada para o experimento 4, Treliça de 10 Barras - Caso Discreto. Todas as propostas encontraram a mesma solução.	71
4.17	Grupos de barras para a treliça de 25 barras	73
4.18	Dados do carregamento para a treliça de 25 barras (kips)	74

4.19	Resultado do experimento 5, Treliça de 25 Barras - Caso Contínuo .	75
4.20	Melhores soluções encontradas para o experimento 5, Treliça de 25 Barras - Caso Contínuo	76
4.21	Resultado do experimento 5, Treliça de 25 Barras - Caso Discreto .	77
4.22	Melhor solução encontrada para o experimento 5, Treliça de 25 Barras - Caso Discreto. Todas as propostas encontraram a mesma solução.	77
4.23	Casos de carregamento para estrutura em anel de 60 barras (kips) .	78
4.24	Variáveis de projeto e correspondentes barras para estrutura anelar de 60 barras	79
4.25	Resultado do experimento 6, Treliça de 60 Barras	80
4.26	Melhores soluções encontradas para o experimento 6, Treliça de 60 Barras	81
4.27	Grupamentos para a treliça de 72 barras	82
4.28	Dados do carregamento para estrutura 72 barras (kips)	83
4.29	Resultado do experimento 7, Treliça de 72 Barras	84
4.30	Melhores soluções encontradas para o experimento 7, Treliça de 72 Barras	85

Lista de Siglas e Abreviaturas

- APM: *Adaptive Penalty Method*
- DE (ou ED): Evolução Diferencial - *Differential Evolution*
- DE+APM: *Differential Evolution with Adaptive Penalty Method*
- DUVDE+APM: *Dynamic Use of Variants of Differential Evolution with Adaptive Penalty Method*
- EC (ou CE): Computação Evolucionista - *Evolutionary Computing*
- EE: Estratégia de Evolução
- GA (ou AG): Algoritmo Genético - *Genetic Algorithm*
- GEN: Número de Gerações
- NA: Número de avaliações da função objetivo
- PE: Programação Evolucionista
- PII: *Probabilistic Iterative Improvement*
- POP: Tamanho da População
- SA: *Simulated Annealing*
- SLS: Busca Local Estocástica - *Stochastic Local Search*

Capítulo 1

Introdução

Problemas práticos de otimização, onde se deseja maximizar ou minimizar alguma medida de desempenho de um dado sistema, ou mesmo outros tipos de problemas que podem ser convenientemente formulados via otimização, são muito comuns nas mais variadas áreas de trabalho. Em geral, a presença de restrições sobre as variáveis de projeto/decisão torna tais problemas mais complexos ainda. Além disso, técnicas de otimização capazes de lidar com restrições que são funções implícitas das variáveis de projeto/decisão, que não requeiram informações adicionais, como gradientes e hessianas, e que se acoplem diretamente a simuladores existentes são altamente desejáveis.

A Evolução Diferencial (DE, de “Differential Evolution”), proposta por Rainer M. Storn e Kenneth V. Price (Storn e Price, 1995) originalmente para problemas irrestritos e com variáveis contínuas, é um método simples, de fácil compreensão, e capaz de gerar bons resultados quando aplicado a diferentes tipos de problemas.

Neste trabalho são apresentadas propostas de aperfeiçoamento da Evolução Diferencial, bem como as propostas que tornam possível a aplicação da DE em problemas de otimização restrita e com variáveis discretas. Além disso, descreve-se detalhadamente a DE e apresenta-se também conceitos básicos necessários ao entendimento do método e da classe de problemas considerados.

O trabalho não se resume ao estudo da DE, mas também apresenta propostas

de aperfeiçoamento do método. A primeira proposta, denominada DE+APM, corresponde à inclusão de um mecanismo de penalização adaptativa junto à evolução diferencial para a melhora dos resultados em problemas com restrições. A segunda proposta consiste no uso de um mecanismo dinâmico de seleção de variantes da DE durante o processo, sendo apresentadas duas alternativas: DUVDE+APM e DUVDE2+APM.

O capítulo seguinte apresenta conceitos básicos de otimização irrestrita e restrita, busca local, os algoritmos evolucionistas e o recozimento simulado. No terceiro capítulo a evolução diferencial é detalhada e diferentes variantes são apresentadas, juntamente com as propostas elaboradas neste trabalho. Para a validação das propostas, experimentos numéricos são apresentados no quarto capítulo, focando-se principalmente em problemas de otimização estrutural. Finalmente, algumas conclusões são apresentadas.

Capítulo 2

Conceitos Básicos de Otimização

Este capítulo apresenta alguns dos conceitos básicos (Friedlander, 1994) importantes para a contextualização e entendimento do trabalho desenvolvido.

2.1 Otimização

Pode-se definir otimização como o processo de minimização ou maximização de uma dada função, denominada usualmente função objetivo. A função objetivo, f , é uma regra que associa a cada elemento x pertencente a um dado conjunto S o valor real $f(x)$. Embora x possa pertencer a um espaço de dimensão infinita¹ considera-se aqui o caso em que $x \in \mathfrak{R}^n$. Assim, cada componente x_i de x é chamada variável de projeto ou decisão.

Por simplicidade, apenas problemas de minimização serão considerados. Isso pode ser feito sem perda de generalidade já que

$$\arg \max_{x \in S} f(x) = \arg \min_{x \in S} [-f(x)] \quad \forall x \in S$$

Como descrito anteriormente, podemos definir matematicamente otimização como:

$$\begin{aligned} &\text{Minimizar } f(x) \\ &\text{sujeita a } x \in S \end{aligned} \tag{2.1}$$

onde $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$ e $S \subset \mathfrak{R}^n$. S é chamado *conjunto viável ou factível* e (2.1) é a

¹ Consulte um texto sobre Cálculo das Variações

forma mais comum dos problemas de programação matemática.

Em seguida, deve-se definir dois importantes conceitos de otimização: mínimo local e mínimo global.

Definição 2.1.1 Um ponto $x^* \in S$ é um *minimizador local* de f em S se e somente se existe $\varepsilon > 0$ tal que $f(x) \geq f(x^*)$ para todo $x \in S$ tal que $\|x - x^*\| < \varepsilon$.

Se $f(x) > f(x^*)$ para todo $x \in S$ tal que $x \neq x^*$ e $\|x - x^*\| < \varepsilon$, diz-se que se trata de um *minimizador local estrito em S* .

Definição 2.1.2 Um ponto $x^* \in S$ é um *minimizador global* de f em S se e somente se $f(x) \geq f(x^*)$ para todo $x \in S$.

Se $f(x) > f(x^*)$ para todo $x \in S$ tal que $x \neq x^*$, diz-se que se trata de um *minimizador global estrito em S* .

A existência de ótimos locais, e não globais, interfere fortemente sobre as metodologias de resolução de problemas de otimização, dado que em muitos casos a técnica empregada pode convergir à ótimos locais, fato este não desejado quando da busca pelo ótimo global.

A existência do ótimo global em um problema de otimização é garantida pelo teorema de Bolzano-Weierstrass.

Teorema 2.1.1 Uma função real contínua f , definida em um conjunto fechado e limitado $S \subset \mathbb{R}^n$, admite um minimizador global em S .

Quando o conjunto S coincide com o próprio \mathbb{R}^n , diz-se que a otimização é irrestrita.

A programação matemática baseia-se fortemente nas definições de otimalidade a serem apresentadas a seguir.

Na otimização irrestrita a condição de otimalidade necessária de primeira ordem é dada pela

Proposição 2.1.1 Seja $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^1$. Se x^* é um minimizador local de f em \mathbb{R}^n , então $\nabla f(x^*) = 0$.

As condições necessárias de otimalidade de segunda ordem são dadas pela

Proposição 2.1.2 Seja $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $f \in C^2$. Se x^* é um minimizador local de f em \mathfrak{R}^n , então:

(i) $\nabla f(x^*) = 0$.

(ii) $\nabla^2 f(x^*)$ é semidefinida positiva.

Por fim, a proposição 2.1.3 apresenta as condições suficientes de segunda ordem.

Proposição 2.1.3 Seja $f : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $f \in C^2$. Se $x^* \in \mathfrak{R}^n$, $\nabla f(x^*) = 0$ e $\nabla^2 f(x^*)$ é definida positiva, então x^* é um minimizador local estrito de f em \mathfrak{R}^n .

2.1.1 Convexidade

Nota-se que todas as proposições enunciadas anteriormente garantem somente a otimalidade local. É possível garantir minimizadores globais apenas para funções objetivo que possuam características especiais. Destaca-se aqui uma característica que pode garantir a otimalidade global: a convexidade.

Definição 2.1.3 Um subconjunto $S \subset \mathfrak{R}^n$ é convexo se e somente se para todo $x, y \in S$, $\lambda \in [0, 1]$ se verifica que $\lambda x + (1 - \lambda)y \in S$.

Definição 2.1.4 Uma função f definida em um convexo S é convexa se e somente se para todo $x, y \in S$, $\lambda \in [0, 1]$ se verifica que:

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

Se para todo $\lambda \in (0, 1)$ e $x \neq y$ vale que:

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y)$$

diz-se que f é estritamente convexa.

A figura 2.1 ilustra esses conceitos em uma dimensão.

As proposições 2.1.4 e 2.1.5 a seguir permitem identificar a convexidade de uma função diferenciável.

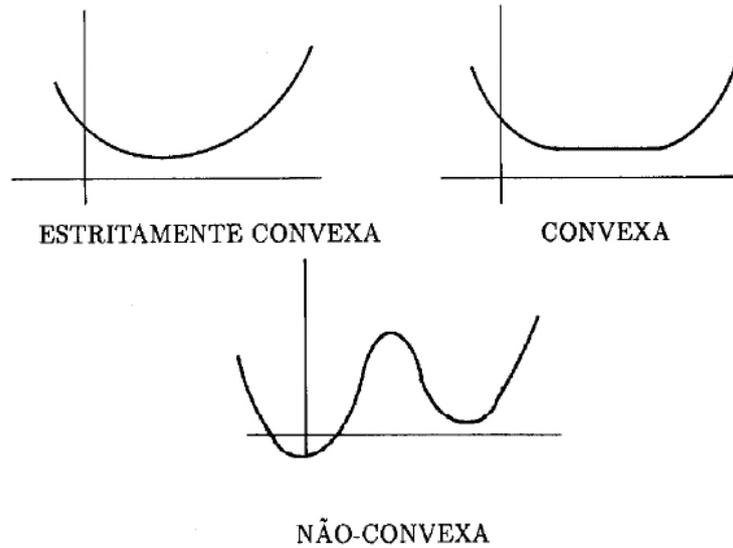


Figura 2.1: Função estritamente convexa, convexa e não-convexa (Friedlander, 1994)

Proposição 2.1.4 Seja $f \in C^1$. Então, f é convexa em S convexo se e somente se para todo $x, y \in S$ se verifica:

$$f(y) \geq f(x) + \nabla^t f(x)(y - x)$$

Proposição 2.1.5 Seja $f \in C^2$. Seja $S \subset \mathbb{R}^n$ convexo tal que S° não é vazio. Então, f é convexa se e somente se $\nabla^2 f(x) \geq 0$ para todo $x \in S$.

De posse dos conceitos apresentados, as proposições 2.1.6 e 2.1.7 a seguir possibilitam a identificação de minimizadores globais de uma função convexa.

Proposição 2.1.6 Seja f uma função convexa definida em S convexo. Então:

- (i) O conjunto $\Gamma \subset S$ onde f toma seu valor mínimo é convexo;
- (ii) Qualquer minimizador local de f é um minimizador global de f .

Proposição 2.1.7 Seja $f \in C^1$ convexa definida em S convexo. Se existe $x^* \in S$ tal que para todo $y \in S$ se verifica que:

$$\nabla^t f(x^*)(y - x^*) \geq 0,$$

então x^* é um minimizador global de f em S .

Entretanto, na prática, muitos problemas não apresentam convexidade e muitos ótimos locais podem estar presentes, não se podendo afirmar que o ótimo local encontrado é também global.

2.1.2 Otimização Restrita

Para os casos nos quais as variáveis de projeto estão sujeitas a determinadas condições (restrições), ou seja, a região S na qual busca-se o ótimo para a função é subconjunto de \mathbb{R}^n , $S \subset \mathbb{R}^n$, tem-se então a otimização restrita.

Uma maneira clássica de descrever o conjunto viável S é através de equações e inequações. O problema de otimização restrita fica então:

$$\begin{aligned} & \text{Minimizar } f(x) \\ & \text{sujeita a } h(x) = 0, g(x) \leq 0, \end{aligned} \tag{2.2}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ com $m < n$ e $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$. Ou seja, $S = \{x \in \mathbb{R}^n | h(x) = 0 \text{ e } g(x) \leq 0\}$. O problema tem m restrições de igualdade e p restrições de desigualdade.

Problemas de otimização restrita mostram-se evidentes e de fundamental importância, principalmente na modelagem de problemas reais em engenharia, como por exemplo na minimização do peso de estruturas (Lemonge, 1999), na minimização de perdas em corte de materiais (Silva e Barbosa, 2008a), e em inúmeros outros exemplos. Isso ocorre porque dificilmente uma situação ou fenômeno está livre de limitações, sejam elas de qualquer natureza.

Se qualquer uma das funções f , g , h não for linear, tem-se um problema de programação não-linear.

Na otimização restrita é importante o conceito de direção factível, ou seja, dado um ponto, uma direção factível é definida por permitir movimentos que levem a novos pontos dentro da região que atende às restrições impostas, mais formalmente, dado $x \in S \subset \mathbb{R}^n$, diz-se que $d \in \mathbb{R}^n$ é uma direção factível em x , se $\exists \bar{\alpha} > 0$ tal que $\tilde{x} = (x + \alpha d) \in S \quad \forall \alpha \in (0, \bar{\alpha}]$.

A primeira observação para problemas de otimização com restrições de igualdade não-lineares é que não há direções factíveis a partir de um ponto $x \in S$, ao contrário dos casos em que apenas restrições lineares estão presentes. Assim, para permanecer em S , a partir de um ponto $x \in S$, os movimentos devem ser curvilíneos, sendo a dificuldade da determinação de um novo ponto $\tilde{x} \in S$ dependente da função h .

Se para um dado ponto $\tilde{x} \in \mathfrak{R}^n$ tem-se $g_i(\tilde{x}) = 0$, diz-se então que a restrição de desigualdade g_i está ativa em \tilde{x} .

No caso em que $S = \{x \in \mathfrak{R}^n | h(x) = 0 \text{ e } g(x) \leq 0\}$ e com $x \in S$ e $K(x) = \{i_1, i_2, \dots, i_{s(x)}\}$ tais que $j \in K(x)$ se e somente se $g_j(x) = 0$, tem-se que $0 \leq s(x) \leq p$. Como as m restrições correspondentes a $h(x)$ estão sempre ativas, se $r(x)$ é o número total de restrições ativas em x , tem-se que $m \leq r(x) = m + s(x) \leq m + p$.

Definição 2.1.5 Diz-se que $x \in S$ é um ponto regular se e somente se o conjunto de vetores $\{\nabla h_1(x), \dots, \nabla h_m(x), \nabla g_{i_1}(x), \dots, \nabla g_{i_{s(x)}}(x)\}$ é linearmente independente.

Com as caracterizações realizadas pode-se definir as condições de otimalidade de KARUSH-KUHN-TUCKER.

Teorema 2.1.2 (KARUSH-KUHN-TUCKER) Considerando o problema 2.2, seja x^* um ponto factível e regular. Seja $K(x^*) = \{i_1(x^*), \dots, i_{s(x^*)}(x^*)\}$ o conjunto de índices correspondentes às restrições de desigualdade que estão ativas em x^* . Sejam $J_h \in \mathfrak{R}^{m \times n}$ e $W_K \in \mathfrak{R}^{s(x^*) \times n}$, dados por

$$J_h(x^*) = \begin{bmatrix} \nabla^t h_1(x^*) \\ \vdots \\ \nabla^t h_m(x^*) \end{bmatrix} \quad W_K = \begin{bmatrix} \nabla^t g_{i_1}(x^*) \\ \vdots \\ \nabla^t g_{i_{s(x^*)}}(x^*) \end{bmatrix}$$

Se x^* é um minimizador local de 2.2, então existem $\lambda^* \in \mathfrak{R}^m$ e $\mu^* \in \mathfrak{R}^{s(x^*)}$ tais que

$$\nabla f(x^*) + J_h^t(x^*)\lambda^* + W_K^t \mu^* = 0$$

e

$$\mu_k^* \geq 0 \text{ para todo } k \text{ tal que } 1 \leq k \leq s(x^*)$$

Teorema 2.1.3 Suponha $f, h, g \in C^2$. Seja x^* (regular) um minimizador local de 2.2. Então, existem $\lambda^* \in \mathfrak{R}^m$, $\mu^* \in \mathfrak{R}^{s(x^*)}$ tais que as condições de primeira ordem sejam verificadas e, além disso, a matriz $\nabla_x^2 L(x^*, \lambda^*, \mu^*)$, definida por

$$\nabla_x^2 L(x^*, \lambda^*, \mu^*) \equiv \nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 h_i(x^*) + \sum_{j=1}^{s(x^*)} \mu_j^* \nabla^2 g_{ij}(x^*),$$

verifica

$$y^t \nabla_x^2 L(x^*, \lambda^*, \mu^*) y \geq 0 \text{ para todo } y \in \tilde{T} \equiv \{y \in \mathfrak{R}^n \mid By = 0\},$$

onde

$$B = \begin{bmatrix} J_h(x^*) \\ W_K \end{bmatrix}$$

e W_K definido como no teorema 2.1.2.

Teorema 2.1.4 Sejam $f, h, g \in C^2$. Seja x^* factível em 2.2 tal que existam $\lambda^* \in \mathfrak{R}^m$ e $\mu^* \in \mathfrak{R}^{s(x^*)}$ com $\mu_k^* \geq 0$ para todo $k \in \{1, \dots, s(x^*)\}$ e

$$\nabla f(x^*) + \sum_{j=1}^m \lambda_j^* \nabla h_j(x^*) + \sum_{j=1}^{s(x^*)} \mu_j^* \nabla g_{ij}(x^*) = 0$$

Supondo que $\nabla_x^2 L(x^*, \lambda^*, \mu^*)$, definida como no teorema 2.1.3, verifica

$$y^t \nabla_x^2 L(x^*, \lambda^*, \mu^*) y > 0$$

para todo $y \in T' \equiv \{y \mid J_h(x^*)y = 0 \text{ e } \nabla g_j(x^*)y = 0 \forall j \in K\}$, onde $K = \{j \in K(x^*) \mid \mu_j^* > 0\}$ então, x^* é um minimizador local estrito de 2.2.

As demonstrações dos teoremas apresentados nesta seção podem ser encon-

tradas em (Luenberger, 1986).

Na prática, restrições adicionais podem estar presentes. Um caso comum é quando cada variável está limitada por um valor mínimo e um valor máximo, ou seja, $x_i^L \leq x_i \leq x_i^U$. Pode ser ainda que as variáveis de projeto estejam restritas a tomar valores num conjunto finito de reais ou inteiros

$$x_i \in C_i = \{v_1, v_2, \dots, v_q\} \quad (2.3)$$

Isto ocorre quando existe um conjunto de componentes, disponíveis comercialmente, por exemplo, de onde será escolhido um determinado número de elementos.

Quando uma ou mais variáveis de projeto estiverem submetidas a uma restrição do tipo 2.3 diz-se que o problema é de otimização mista (variáveis contínuas e discretas) e os teoremas enunciados não podem ser aplicados diretamente.

Deve-se ressaltar que, em muitos casos, métodos determinísticos são as melhores escolhas para a resolução de certas classes de problemas de otimização restrita; por exemplo, o método simplex aplicado à problemas de otimização linear (Dantzig, 1963). Entretanto, existem casos nos quais tais métodos não são capazes de encontrar soluções ou necessitam de exacerbado esforço computacional. Nesses casos outras metodologias são empregadas, tais como a busca local estocástica e meta-heurísticas como a evolução diferencial, apresentada em detalhes neste trabalho.

Apenas para exemplificar métodos determinísticos empregados na solução de problemas de otimização, podemos citar: método de gradientes conjugados (Hestenes e Stiefek, 1952), métodos quase-Newton (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970), bem como os métodos denominados de “ordem zero”, os quais fazem uso apenas do valor da função objetivo, não se utilizando portanto de gradiente, como é o caso do simplex de Nelder-Mead (Nelder e Mead, 1965) e o Hooke & Jeeves (Hooke e Jeeves, 1961).

Observa-se que tais métodos utilizam-se dos conceitos e definições anteriormente apresentados e alguns fazem uso do gradiente e da hessiana. Apesar disso,

sabe-se que em determinados casos não é possível nem ao menos o cálculo do gradiente da função. Isso porque f pode não ser contínua ou não possuir derivadas contínuas, ou ainda cujo cálculo seja muito complexo, o mesmo podendo ocorrer para as restrições. Além disso, as variáveis de decisão podem ser discretas. Nestes casos o uso de (meta-) heurísticas pode se mostrar vantajoso.

2.2 Metaheurísticas

Uma metaheurística é um método heurístico² para a solução de uma classe muito geral de problemas para os quais, normalmente, não se conhece algoritmos determinísticos eficientes. A palavra metaheurística vem do grego, e combina o prefixo “meta” (que significa “além” ou “mais”, aqui no sentido de “alto nível”) e heurística (que significa “encontrar” ou “descobrir”). Assim, metaheurísticas são heurísticas de mais alto nível, que podem lançar mão de outras heurísticas visando a formulação de algoritmos eficientes com características de busca local e de busca global.

Essa seção tem a finalidade de apresentar conceitos importantes envolvidos no estudo de metaheurísticas, bem como apresentar algumas das mais conhecidas técnicas assim classificadas.

2.2.1 Busca Local Estocástica

Uma opção aos métodos determinísticos existentes para solução de problemas de otimização está no denominado paradigma de busca, o qual visa encontrar soluções aproximadas dos problemas. Os algoritmos de busca visam percorrer diversas regiões do espaço no qual a função a ser otimizada é definida avaliando as possíveis soluções.

Como definido em (Hoos e Stützle, 2005) a busca pode ser classificada como sistemática ou local. Quando sistemática (também conhecida como completa)

² De acordo com (Ferreira, 1999), heurística é uma metodologia ou algoritmo para resolver problemas por métodos que, embora não rigorosos, refletem o conhecimento humano e permitem obter uma solução satisfatória.

examina-se todas as soluções possíveis e retorna-se o ótimo garantidamente global ou a não-existência de uma solução global ótima.

Por outro lado, a busca local não garante que a solução encontrada seja ótimo global e nem a não-existência de solução, entretanto, possui custo computacional bastante reduzido quando comparado ao caso anterior.

A busca pode ser definida ainda como construtiva ou perturbativa. No primeiro caso a busca é iniciada sem nenhuma solução e a solução final encontrada é construída a cada passo da busca. No caso da busca perturbativa a busca é iniciada com uma solução completa e a mesma é modificada a cada iteração, visando com isso a melhora da solução a cada iteração.

A busca local, como também já descrito, inicia a busca de um determinado ponto no espaço de busca e os movimentos realizados são definidos somente por informação local, sendo assim, a busca é normalmente incompleta e não garante a solução ótima ao fim do processo.

Com as definições anteriores pode-se dizer que o desempenho de buscas sistemáticas é superior ao das buscas locais quando avalia-se somente o resultado obtido, desconsiderando o custo computacional envolvido.

Dessa forma a busca local possui vantagens principalmente quando aplicada a problemas em que se requer soluções em um curto limite de tempo, ou quando outras técnicas mostram-se inapropriadas à aplicação.

Algoritmos de busca local estocástica fazem escolhas aleatórias na geração ou seleção de candidatos à solução. Esses algoritmos são largamente utilizados na aproximação de soluções de problemas NP-Difíceis, principalmente em otimização combinatorial.

Nesse momento torna-se necessário a definição de novos conceitos, tais como instância de um problema, a qual pode ser definida como uma exemplificação do mesmo: um problema definido de forma genérica quando munido de dados específicos, tais como sua dimensão e demais dados de entrada, é considerado uma instância

O conceito de solução candidata é referente à geração de uma solução tentativa pelo algoritmo utilizado, que pode ser modificada ou descartada. Assim, a solução gerada e avaliada no momento é uma solução candidata. Em algoritmos populacionais, como os evolucionistas, pode-se usar o termo indivíduo.

Outra definição importante é a de vizinhança de uma solução candidata, que corresponde ao conjunto de soluções candidatas que podem ser alcançadas a partir de um passo de busca. Por exemplo, considerando uma codificação binária onde uma solução candidata corresponda à cadeia $\{000\}$, e considerando que o passo de busca consista na alteração de um dos seus *bits*, o conjunto que compõe a vizinhança de $\{000\}$ é dado por $\{100, 010, 001\}$.

Outro ponto importante para uma busca estocástica e bem como para os Algoritmos Evolucionistas, é a função de avaliação. Isso porque é necessário definir quando um candidato à solução é melhor que um outro candidato. O que pode ser feito pela comparação do valor da função de avaliação aplicada sobre cada candidato. Tal função depende do problema tratado e está intimamente ligada à função objetivo do problema e às eventuais restrições presentes.

Em Algoritmos Evolucionistas a função de avaliação é conhecida como função de aptidão.

Por fim, outros dois conceitos devem ser introduzidos em torno da construção de algoritmos de busca: a “Intensificação” e a “Diversificação”. Tais atividades são importantes durante todo o processo de busca e são conflitantes, dado que o aumento da intensificação implica diretamente na redução da diversificação e vice-versa.

A estratégia de diversificação refere-se à exploração do espaço de busca. Ou seja, a diversificação visa avaliar diferentes regiões do espaço de busca quanto ao seu potencial de conter boas soluções.

A intensificação é a etapa da busca focada numa dada região do espaço de soluções. Essa etapa é de grande importância para o melhoramento da solução. Entretanto pode causar a convergência do processo a um ótimo local, não neces-

sariamente global.

As estratégias de busca devem ser combinadas de forma a obter-se o melhor desempenho, encontrando assim o melhor compromisso entre intensificação e diversificação, ambas estratégias fundamentais ao processo.

Ressalta-se ainda que existe uma grande variedade de métodos baseados em buscas locais estocásticas, tais como “Variable Depth Search”, “Randomized Iterative Improvement” e “Tabu Search”.

2.2.2 Simulated Annealing

“Simulated Annealing” (SA) ou “Recozimento Simulado”, é uma família de algoritmos proposta originalmente em (Kirkpatrick et al., 1983). Pode ser considerada a generalização de um método denominado “Probabilistic Iterative Improvements” (PII), no qual soluções consideradas inferiores podem ser aceitas com uma determinada probabilidade. Entretanto, o SA simula o resfriamento de sólidos, descrito em detalhes em diversos trabalhos, tais como (Kirkpatrick et al., 1983; Cerný, 1985; Harland e Salamon, 1988).

O SA difere do PII por possuir uma probabilidade variável de aceite de soluções inferiores durante o processo de busca. Dessa forma, a probabilidade é dada pela equação 2.4, onde t é a temperatura, reduzida ao longo do processo, e δE é denominado de energia livre, sendo calculada pela diferença entre os valores da função objetivo aplicada à solução atual e à nova solução calculada.

$$P(\delta E) = \exp\left(-\frac{\delta E}{t_k}\right) \quad (2.4)$$

O algoritmo 1 apresenta um pseudo-código do SA básico.

Algoritmo 1 Algoritmo básico de SA

procedure SA(t_0, M_0) $i_0 \in S$ ▷ Solução inicial $k = 0$ ▷ Contador da temperatura inicial $t_k = t_0, t_k \geq 0$ ▷ Escolha da temperatura inicial $M_k = M_0, M_k > 0$ ▷ Escolha do número de iterações para cada t_k **while**stopCriterion = false **do****for** $m = 0 : M_k$ **do**Select $i \in N(i_0)$ ▷ Seleciona um vizinho da solução atual $\delta E = f(i) - f(i_0)$ **if** ($\delta E \leq 0$) **then** $i_0 = i$ **else** $u \in [0, 1)$ **if** ($u < \exp(-\frac{\delta E}{t_k})$) **then** ▷ Verifica se aceita a nova solução $i_0 = i$ **end if****end if****end for** $k = k + 1$ $t_k = \alpha(t_{k-1})$ ▷ Calcula o decaimento da temperatura**end while****end procedure**

Como citado anteriormente, o SA pode ser considerado uma família de algoritmos, isso porque diversas variações foram propostas, podendo-se citar (Price, 1994; Tsallis e Stariolo, 1996; Lamberti, 2008). Além disso, uma importante proposta baseada no SA foi o “Genetic Annealing”, devido a Kenneth Price. Tal proposta utiliza um população de soluções candidatas (Price, 1994), sendo essa implementação um dos precursores da Evolução Diferencial.

2.2.3 Algoritmos Evolucionistas

Os “Algoritmos Evolucionistas”, (“Evolutionary Algorithms” - EA), são largamente aplicados em problemas de otimização, sendo baseados em população, ou seja, manipulam diversos candidatos à solução simultaneamente no processo de busca.

Os EAs são inspirados no modelo de origem das espécies de Darwin(Bäck, 1996; Mitchell, 1996), no qual indivíduos mais adaptados possuem maior probabilidade de gerarem descendentes e dessa forma manterem na população suas características.

Uma das metodologias mais conhecidas e mais aplicadas são os Algoritmos Genéticos desenvolvidos por John Holland nos anos 60, aprimorados em conjunto com seus alunos nos anos 70 e aplicados para otimização por K. De Jong(Jong, 1975).

A diferenciação entre os Algoritmos Evolucionistas existentes dá-se principalmente pelo mecanismo de seleção proposto, a codificação utilizada e na aplicação de seus operadores.

Alguns dos pontos mais importantes dos Algoritmos Evolucionistas são dados pela codificação e pelos operadores de recombinação e mutação. A codificação é a representação de uma solução candidata, ou seja, como ela é armazenada e manipulada, sendo que em muitos casos codificações binárias são aplicadas.

Os principais operadores dos Algoritmos Evolucionistas são os operadores de mutação e recombinação. A mutação é um operador unário responsável pela intensificação da busca enquanto a recombinação produz soluções novas manipulando pelo menos um par de soluções existentes.

Em algoritmos como os genéticos, a recombinação foi inspirada no processo de reprodução sexuada. Um dos mais simples mecanismos de recombinação utilizados é a recombinação de um ponto, no qual dois indivíduos são selecionados (denominados como pais) e um determinado ponto de suas codificações é escolhido, em seguida um novo individuo é gerado pela junção dos valores presentes na codi-

ficação de um dos pais selecionados, anteriores ao ponto, e os valores presentes na codificação do outro pai, posteriores ao ponto selecionado.

A mutação realiza normalmente modificações pontuais em uma solução manipulada. Na codificação binária, uma freqüente mutação realizada é a seleção de um dos valores presentes no indivíduo e em seguida a sua troca.

Ressalta-se ainda o critério de seleção existente nos Algoritmos Evolucionistas, pois é necessário definir como os indivíduos são selecionados para participarem da geração de novos indivíduos.

Existem diversos mecanismos de seleção e entre os mais utilizados destacam-se os usualmente aplicados em Algoritmos Genéticos, denominados “seleção por roleta”, “seleção por torneio” e “seleção por ranking”. No caso de uma seleção por roleta os indivíduos recebem probabilidades de serem selecionados correspondentes às suas aptidões relativas. Gera-se posteriormente um valor pseudo-aleatório, “gira-se a roleta”, e então o indivíduo correspondente a esse valor é selecionado para a aplicação do operador de recombinação. No caso da “seleção por torneio” dois ou mais indivíduos são tomados aleatoriamente na população, e então comparados; e aquele que possuir melhor aptidão é o selecionado. Para a “seleção por ranking” todos os indivíduos da população são ordenados de acordo com as aptidões, dessa forma a probabilidade de seleção depende somente da posição no “ranking”, e não mais do valor da aptidão.

A construção básica de uma algoritmo evolucionista pode ser descrita pelo diagrama da figura 2.2.

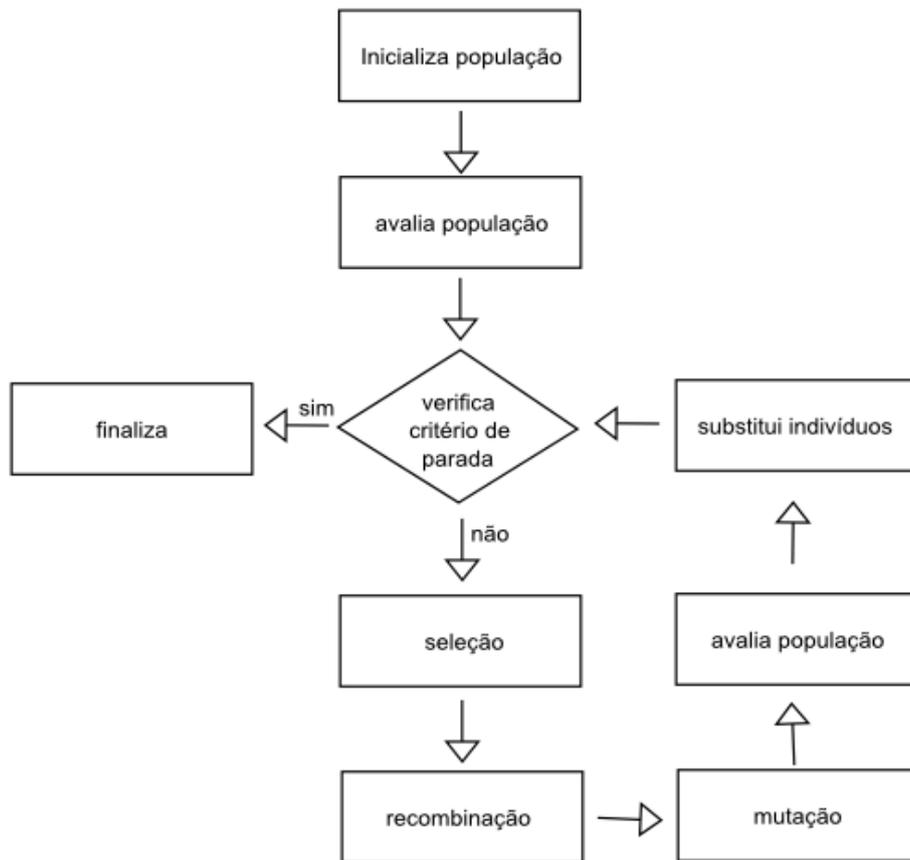


Figura 2.2: Descrição de um algoritmo evolucionista

Capítulo 3

Evolução Diferencial

Evolução Diferencial, do inglês “Differential Evolution” (DE), é uma metodologia estocástica populacional simples e eficiente para otimização global.

Sua formulação inicial foi proposta por Rainer M. Storn e Kenneth V. Price (Storn e Price, 1995) e, como outras técnicas evolucionistas, em sua primeira proposta, não possuía mecanismos de manipulação de restrições, bem como era aplicável somente aos problemas descritos por variáveis contínuas.

Pode-se descrever a DE de forma intuitiva, e conseqüentemente informal, como uma manipulação de vetores que representam as soluções candidatas. O processo evolutivo baseia-se nos movimentos gerados no espaço de busca dados pelo acréscimo de um determinado passo sobre um vetor base. Tal passo é definido pela ponderação da diferença entre outros vetores pertencentes à população. A operação básica é descrita por

$$v_{\text{novo}} = v_{\text{base}} + F(v_1 - v_2) \quad (3.1)$$

onde F determina a ponderação da diferença, v_{novo} é o novo vetor gerado, v_{base} um vetor selecionado na população, assim como os vetores v_1 e v_2 . No algoritmo básico da DE esses três últimos vetores são selecionados de forma aleatória na população.

O movimento descrito anteriormente define a heurística básica de evolução da técnica, sendo também a origem de seu nome.

A simplicidade da evolução diferencial e os importantes resultados obtidos

por sua aplicação nas mais diversas áreas alavancam seu aperfeiçoamento, principalmente sobre a otimização, ampliando seu domínio de aplicação.

As subseções seguintes apresentam um histórico da DE, bem como alguns dos avanços e propostas recentes visando a otimização restrita.

Por fim, este capítulo apresenta também duas novas propostas de aperfeiçoamento da DE para a otimização restrita, que serão referidas pelas siglas DE+APM e DUVDE+APM.

3.1 Histórico

A evolução diferencial origina-se de um algoritmo baseado em população para otimização combinatorial denominado “Genetic Annealing” proposto por Price em outubro de 1994, (Price, 1994).

Após sua publicação, Price foi contactado por Storn sobre a possibilidade do uso do algoritmo na resolução do problema de ajuste do polinômio de Chebyshev, sendo a determinação dos coeficientes deste polinômio uma tarefa considerada difícil para os otimizadores propostos de forma geral.

A aplicação do algoritmo de Price encontrou solução para o caso de cinco dimensões. Entretanto sua convergência foi lenta e de difícil determinação de parâmetros eficientes. Após a aplicação inicial, o algoritmo foi modificado de forma a utilizar valores reais no lugar da codificação por cadeias de “bits”. Além disso, operações aritméticas substituíram as operações lógicas. Com isso desenvolveu-se a operação básica de mutação da DE.

Tais modificações tornaram o “Genetic Annealing” um otimizador numérico e não mais um algoritmo combinatorial e deram origem à primeira versão da DE. Por fim, Storn propôs a separação da população dos “pais” e da “prole”, ou seja, a separação dos vetores utilizados na geração de um novo vetor e a população que armazena os novos vetores gerados. Dessa maneira, facilitou-se a utilização da DE em arquiteturas paralelas.

Com o novo algoritmo desenvolvido, o problema de Chebyshev foi resolvido

sem dificuldades para até 33 dimensões. Além disso, o algoritmo apresentou bons resultados para outras funções teste.

Os primeiros resultados foram descritos em um relatório técnico ICSI (TR-95-012) - “Differential Evolution - A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces” - (Storn e Price, 1995). O sucesso das aplicações da DE resultou em diversas outras publicações, como por exemplo, (Price, 1999; Storn e Price, 1997; Price et al., 2005).

A partir daí diversos autores realizaram trabalhos no contexto da DE, com as mais diferentes propostas e objetivos, como por exemplo (Mezura-Montes et al., 2006b; Soliman e Bui, 2008; Brest et al., 2006; Lampinen e Zelinka, 1999). Além disso, estudos sobre a evolução desta metaheurística são encontrados em (Price et al., 2005; Chakraborty, 2008; Silva e Barbosa, 2008b).

3.2 Descrição do Algoritmo

O algoritmo básico da evolução diferencial inicia-se pela geração aleatória de soluções candidatas, denominadas vetores ou indivíduos, sendo o conjunto desses indivíduos denominado população.

Outra característica da DE semelhante aos demais algoritmos evolucionistas deve-se à evolução dos indivíduos por sucessivas gerações, sendo o número de gerações utilizadas fixado originalmente no início do processo.

Uma importante característica da DE é o número relativamente pequeno de parâmetros utilizados, sendo eles a amplitude, ou ponderação, da diferença empregada (F), a probabilidade de ocorrência de recombinação (CR), o número de indivíduos/vetores mantidos na população (POP) e o número de gerações realizadas durante o processo (GEN).

Além disso, a DE é descrita basicamente por seu operador de mutação, o qual define o passo de busca realizado, sendo esse formado originalmente pela ponderação da diferença entre vetores acrescida a um vetor base. Essa operação é realizada para cada dimensão do problema, ou seja, para cada valor contido no

vetor/indivíduo, $x \in \mathbb{R}^n$.

Entretanto, o modelo básico da DE utilizado atualmente considera também um operador de recombinação, o qual realiza a intercalação de valores gerados pela mutação e valores presentes em um indivíduo alvo. Para isso, determina-se uma probabilidade CR de ocorrência da recombinação. Na geração de um novo indivíduo, a mutação é realizada com probabilidade CR, caso contrário, o valor presente no indivíduo alvo é copiado diretamente ao novo indivíduo gerado. Para garantir a realização da mutação em ao menos uma variável, seleciona-se, anteriormente à geração de um novo indivíduo, um dos componentes do mesmo, denominado *jRand*, onde a mutação é realizada independentemente da probabilidade CR.

As etapas citadas possuem diversas variações propostas, as quais caracterizam as variantes da DE. O algoritmo básico é designado por DE/rand/1/bin e seu pseudo-código é apresentado no algoritmo 2.

Algoritmo 2 Algoritmo DE/rand/1/bin

```
1: procedure DE(POP, GEN, F, CR)
2:   G = 0
3:   CREATERANDOMINITIALPOPULATION(POP)
4:   Evaluate  $f(\vec{x}_{i,G})$  ▷  $\forall i, i = 1, \dots, POP$ 
5:   for G = 1 : GEN do
6:     for i = 1 : POP do
7:       SELECTRANDOMLY( $r_1, r_2, r_3$ ) ▷  $r_1 \neq r_2 \neq r_3$ 
8:       jRand = RANDINT(1, N) ▷ N é a dimensão do problema
9:       for j = 1 : N do
10:        if (RAND(0, 1) < CR or j = jRand) then
11:           $u_{i,j,G+1} = x_{r_3,j,G} + F(x_{r_1,j,G} - x_{r_2,j,G})$ 
12:        else
13:           $u_{i,j,G+1} = x_{i,j,G}$ 
14:        end if
15:      end for
16:      if  $f(u_{i,G+1}) \leq f(x_{i,G})$  then
17:         $\vec{x}_{i,G+1} = \vec{u}_{i,G+1}$ 
18:      else
19:         $\vec{x}_{i,G+1} = \vec{x}_{i,G}$ 
20:      end if
21:    end for
22:  end for
23: end procedure
```

Existem ainda alguns pontos importantes a serem descritos, sendo o primeiro deles a nomenclatura utilizada. Sendo assim, a Tabela 3.1 identifica os termos usualmente utilizados no contexto da DE. A idéia crucial por trás da evolução diferencial é seu esquema de geração do indivíduo teste ou temporário, portanto a Figura 3.1, retirada de (Price et al., 2005), apresenta graficamente o esquema de geração descrito anteriormente.

Tabela 3.1: Nomenclaturas utilizadas na DE.

Termo	Descrição
Indivíduo/Vetor	Candidato à solução
População	Conjunto de candidatos à solução manipulados
Vetor base	Vetor pertencente à população, selecionado como base da mutação. Seus valores recebem um acréscimo determinado na operação de mutação
Vetor alvo	Vetor pertencente à população, o qual competirá com o novo vetor gerado
Vetor temporário/teste	Vetor gerado após o processo de mutação e recombinação

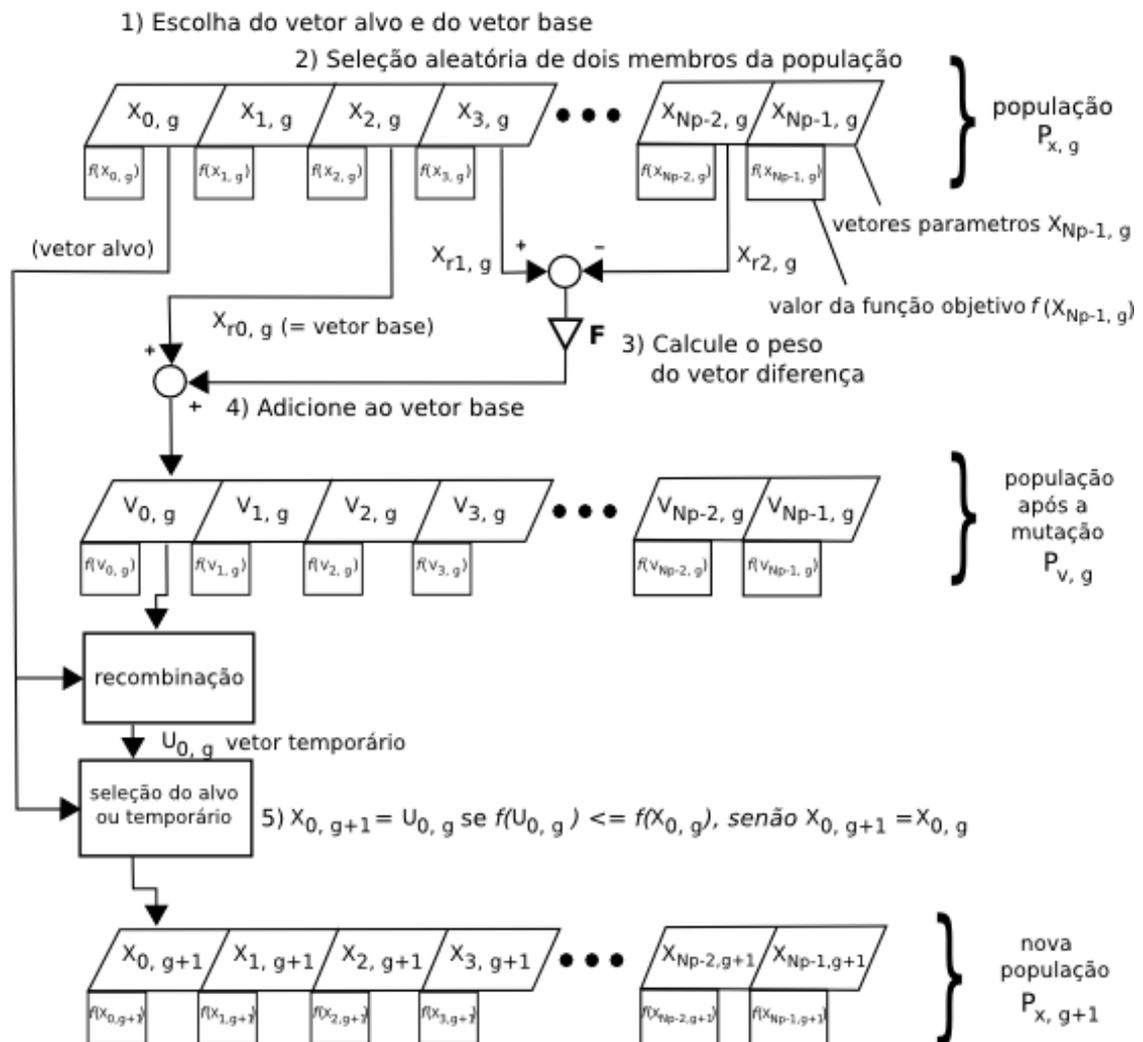


Figura 3.1: Esquema de geração do vetor teste (“trial”) - Sendo N_p o número de vetores na população

3.2.1 Um Exemplo Simples

Afim de facilitar o entendimento da evolução diferencial, a geração de um indivíduo pelo algoritmo básico da DE, apresentado anteriormente, é realizada para um problema simples de otimização, incluindo uma explanação gráfica dos movimentos realizados.

A função a ser minimizada é dada por:

$$f(x, y) = x^2 + y^2 \quad (3.2)$$

Com o intuito de facilitar a apresentação da aplicação do algoritmo, o problema alvo não é sujeito a restrições. O problema tem solução trivial, dada por $x = 0, y = 0$ e conseqüentemente $f(0, 0) = 0$.

O algoritmo da evolução diferencial foi configurado com os parâmetros: indivíduos na população (POP): 4, ponderação da diferença (F): 0.8 e probabilidade de recombinação (CR): 0.9.

Os valores aleatórios foram gerados por um gerador de números pseudo-aleatórios escrito em linguagem de programação C e os valores da população inicial gerados no intervalo $[-10, 10]$.

A população inicial gerada foi

$$\text{População inicial} = \left\{ \begin{array}{c|c|c} \text{Indivíduo} & x & y \\ \hline 1 & -4,26 & -1,56 \\ \hline 2 & 7,53 & -9,19 \\ \hline 3 & -5,28 & -3,15 \\ \hline 4 & 8,92 & -6,53 \end{array} \right.$$

A notação utilizada para a identificação de cada indivíduo na população será r_i com $i = 1, 2, 3, 4$ e o indivíduo gerado identificado como u , bem como a determinação da variável manipulada será dado por x e y , por exemplo, o primeiro vetor da população possui componentes dados por: $r_{1x} = -4,26$ e $r_{1y} = -1,56$. Contudo,

sabe-se que para efeito de implementação as diversas variáveis que compõem uma solução são armazenadas em um vetor que é também denominado na literatura de algoritmos evolucionistas como “cromossomo”. Sendo assim, a identificação de tais variáveis é efetivada por meio do índice no vetor ao qual a mesma está relacionada.

A função objetivo é apresentada graficamente, dentro do intervalo considerado para a população inicial, na figura 3.2. A figura 3.3 mostra a posição dos indivíduos sobre um gráfico de curvas de nível.

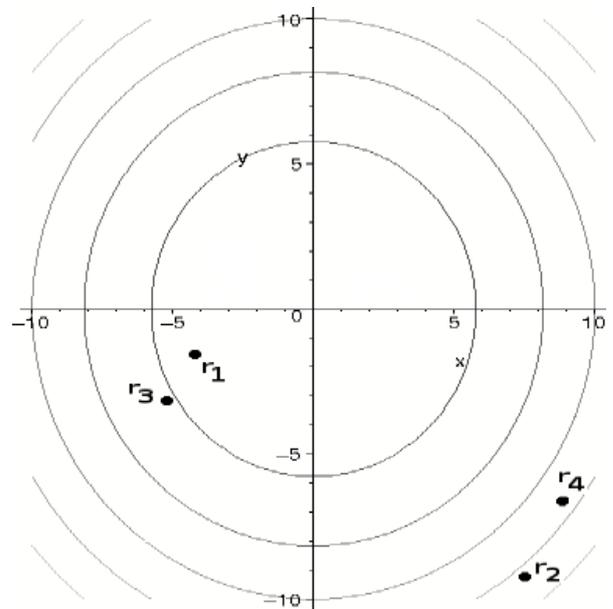
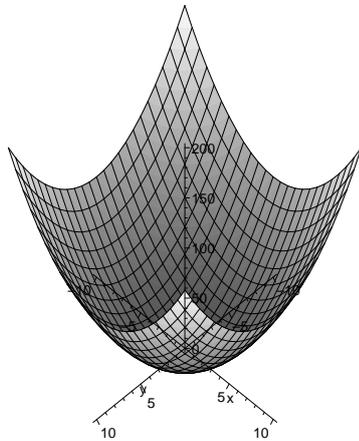


Figura 3.3: População sobre a função

Figura 3.2: Função $f(x, y) = x^2 + y^2$ $f(x, y) = x^2 + y^2$

O primeiro passo da DE é realizar para cada indivíduo da população a geração de um novo indivíduo. Portanto, para o indivíduo 1 realiza-se a geração do novo indivíduo por meio dos operadores de mutação e recombinação, sendo necessários para tanto, a seleção aleatória de três indivíduos na população. Entretanto, neste exemplo, apenas quatro indivíduos compõem a população e, conseqüentemente, são selecionados os três demais, diferentes do atual alvo, indicado como r_1 . Apesar disso, a ordem de seleção dos indivíduos é importante, pois determina, por exemplo,

qual deles será o indivíduo base.

Nesse exemplo a ordem selecionada aleatoriamente foi: (4, 2, 3). Considerando, seguindo o algoritmo 2, o indivíduo 3 como base, as operações de mutação definidas para cada variável do novo vetor gerado, u , são:

$$u_x = r_{3x} + 0,8(r_{4x} - r_{2x}) = -5,28 + 0,8(8,92 - 7,53) = -4,16$$
$$u_y = r_{3y} + 0,8(r_{4y} - r_{2y}) = -3,15 + 0,8(-6,53 - (-9,19)) = -1,02$$

A recombinação é guiada por duas seleções aleatórias, a primeira define em qual variável a mutação será garantidamente aplicada. No exemplo a dimensão selecionada foi x . A segunda escolha está relacionada à probabilidade de ocorrência para as demais variáveis. No corrente exemplo um valor aleatório foi selecionado, 0,58, valor este inferior à 0,9 (definido como CR , no início do procedimento) e portanto os valores gerados pela mutação para ambas as variáveis serão utilizadas, não ocorrendo a cópia direta do valor contido no indivíduo alvo, r_1 , para o novo indivíduo gerado.

Após a geração do novo indivíduo, $u = (-4,16, -1,02)$, o mesmo é comparado ao indivíduo alvo, no caso o indivíduo 1. A função objetivo aplicada sobre ambos resulta em: $f(u) = 18,34$ e $f(r_1) = 20,58$. Como $f(u) < f(r_1)$ o novo indivíduo gerado substituirá o indivíduo 1 na população na próxima geração. Caso a aptidão do novo indivíduo fosse inferior, o mesmo seria descartado e o atual indivíduo da população mantido.

A operação de mutação realizada é descrita graficamente na figura 3.4.

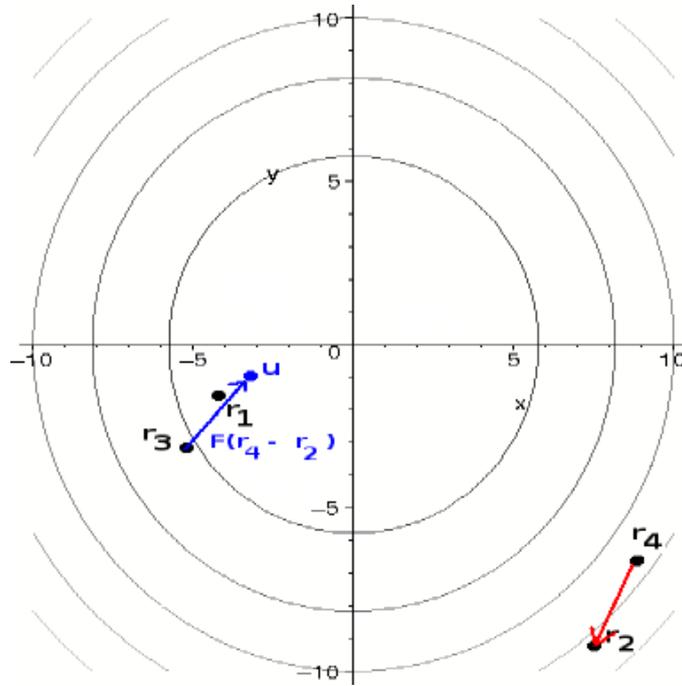


Figura 3.4: Operação de mutação

O procedimento foi repetido para os demais indivíduos da população e o resultado obtido ao final da geração é dado por:

$$\text{População} = \left\{ \begin{array}{c|c|c} \text{Indivíduo} & x & y \\ \hline 1 & -4,16 & -1,02 \\ \hline 2 & 7,53 & -9,19 \\ \hline 3 & -3,01 & -0,24 \\ \hline 4 & 4,15 & 5,45 \end{array} \right.$$

3.3 Variantes

Outro ponto importante da DE é a escolha da variante utilizada. Sendo assim, a maneira como os vetores envolvidos na mutação são selecionados, o número de diferenças realizadas e o modelo de recombinação utilizado determinam a variante descrita seguindo o padrão: *DE/mecanismo-de-seleção/número-de-diferenças/modelo-de-recombinação*.

“Mecanismo-de-seleção” indica de que forma os vetores são selecionados e no caso do algoritmo base é denominado como “rand”, dado que os vetores são selecionados de forma aleatória.

“Número-de-diferenças” indica o número de pares de vetores utilizados na subtração, ou seja, o número de diferenças ponderadas realizadas. No algoritmo apresentado apenas uma diferença é aplicada, e indica-se portanto o valor 1.

Por fim, “modelo-de-recombinação” indica como o operador de recombinação é aplicado. O modelo mais utilizado representando quase a totalidade dos trabalhos relacionados à DE é o binomial. Nesse modelo a recombinação é realizada em um componente do vetor com uma probabilidade CR, sendo assim, cada componente possui uma probabilidade CR de ser gerada por meio da mutação ou ser copiada diretamente do vetor alvo. Apesar de ser largamente aplicado e em muitos casos sua aplicação ser omitida da denominação, existem outras possibilidades, como o caso exponencial, um ponto e N-pontos. Esses casos podem ser encontrados com mais detalhes em (Price et al., 2005), e determinam quais componentes do vetor serão gerados por meio da mutação, por exemplo, no caso de um ponto os componentes anteriores ao ponto selecionado são gerados por mutação e os demais componentes são copiados do vetor alvo.

Pode-se verificar por meio das informações anteriores que a determinação da variante da DE é realizada basicamente pelo operador de mutação utilizado, descrito no algoritmo 2 na linha 11.

De posse dessas caracterizações, a seguir são apresentadas as principais variantes utilizadas neste trabalho.

Variante DE/best/1/bin

Esse modelo, proposto por (Price, 1999) modifica o modelo inicial (DE/rand/1/bin) apenas por utilizar como vetor base da mutação o melhor indivíduo da população (best). Dessa forma a operação de mutação é descrita por

$$u_{j,i} = x_{j,best} + F(x_{j,r_1} - x_{j,r_2}) \quad (3.3)$$

sendo $x_{j,best}$ o indivíduo de melhor aptidão na população na corrente geração e x_{j,r_1} e x_{j,r_2} indivíduos selecionados aleatoriamente como no modelo inicial descrito anteriormente.

Variante DE/target-to-best/1/bin

Esse modelo, proposto em (Price, 1999), utiliza tanto o melhor indivíduo da população quanto o vetor alvo (indivíduo que será usado na comparação com aquele gerado após a mutação). Assim, combina-se essas duas informações, ampliando o operador de mutação para

$$u_{j,i} = x_{j,i} + F(x_{j,best} - x_{j,i}) + F(x_{j,r_1} - x_{j,r_2}) \quad (3.4)$$

sendo $x_{j,i}$ o vetor alvo.

Variante Proposta por Mezura-Montes, Velázquez-Reyes e Coelho

Esse modelo, proposto por (Mezura-Montes et al., 2006b), não recebe uma classificação, mas possui semelhança com o modelo anterior (DE/target-to-best/1/bin) diferenciando-se pelo fato de utilizar como vetor base um indivíduo selecionado aleatoriamente, x_{j,r_3} , como em DE/rand/1/bin e utilizar diferentes valores para cada combinação envolvida, ou seja, dá diferentes pesos F_α e F_β a cada parcela na operação:

$$u_{j,i} = x_{j,r_3} + F_\alpha(x_{j,best} - x_{j,r_2}) + F_\beta(x_{j,i} - x_{j,r_1}) \quad (3.5)$$

Existem ainda um grande número de variantes que foram propostas na literatura, e estudos comparativos mais detalhados sobre o assunto são encontrados em (Mezura-Montes et al., 2006b; Qing, 2008).

3.4 Aplicação em Problemas Mistos

Para a expansão do domínio de aplicação da DE torna-se necessário sua adequação à problemas que envolvam variáveis inteiras ou discretas, isso porque a formulação da DE é diretamente aplicada apenas a variáveis contínuas, fato que é facilmente observado na definição de seu operador de mutação.

O problema de estender a resolução contínua não é exclusivo da DE, também surgindo em outras metaheurísticas, sendo assim, técnicas anteriormente estudadas foram aplicadas à DE. A principal solução é um simples arredondamento do valor obtido para um valor discreto mais próximo, técnica proposta por (Lampinen e Zelinka, 1999) para a DE e hoje sendo utilizada na maioria dos trabalhos que tratam de problemas mistos (discretos-inteiros-contínuos) com a aplicação da DE.

3.5 Critérios de Seleção e Tratamento de Restrições

A aplicação da DE a problemas práticos requer claramente aperfeiçoamentos, de forma a ser capaz de manipular restrições. Ou seja, além de otimizar uma determinada função objetivo, a implementação da DE deve ser capaz de satisfazer as restrições presentes no problema.

Dessa forma o espaço de busca, anteriormente totalmente livre, é reduzido a uma região, a qual compreende as soluções que satisfazem as restrições do problema considerado. Com isso podemos classificar as regiões do espaço de busca em regiões factíveis (que satisfazem as restrições) e infactíveis (que não satisfazem). Tais restrições podem ser contínuas ou não, bem como podem ser definidas por funções lineares ou não-lineares. Além disso, em muitos casos, as regiões factíveis são extremamente reduzidas e até mesmo disjuntas, tornando a busca de uma solução factível tão complexa quanto a otimização da função objetivo.

Dada a necessidade descrita, um grande número de propostas de manipulações de restrições junto à DE podem ser encontradas na literatura, sendo as de maior destaque apresentadas a seguir. Como a necessidade de técnicas de manipulação de restrições não é exclusividade da DE, técnicas anteriormente aplicadas a diferentes metaheurísticas, tais como os algoritmos genéticos, foram adaptadas para o contexto da DE.

3.5.1 Critério de Seleção de Deb

Uma das principais técnicas de manipulação de restrições foi proposta inicialmente para algoritmos genéticos em (Deb, 2000) e é empregada junto à DE por exemplo em (Kim et al., 2007; Zielinski e Laur, 2006; Mezura-Montes et al., 2006c). Nela os indivíduos são gerados normalmente, alterando-se apenas o processo de inserção, ou seja, na escolha de qual indivíduo permanecerá na população após a geração de uma nova solução. Nesse caso são mantidos na população indivíduos factíveis e infactíveis, mas um indivíduo factível sempre prevalece quando comparado à um infactível. O mecanismo de inserção é descrito a seguir:

- Se um indivíduo factível é comparado à um indivíduo infactível, mantém-se o factível;
- Se ambos os indivíduos são factíveis, mantém-se aquele que possuir melhor valor da função objetivo;
- Se ambos os indivíduos são infactíveis, mantém-se aquele que possuir a menor soma das violações nas restrições.

3.5.2 Penalização

Uma técnica largamente utilizada em algoritmos evolucionistas, e dessa forma aplicável à DE, consiste na penalização de indivíduos infactíveis. Assim, indivíduos que não satisfazem a todas as restrições sofrem uma ampliação do valor de sua aptidão (para problemas de minimização), tornando-se assim menos aptos. Entretanto, são mantidos na população e podem competir com indivíduos válidos. Essa técnica torna viável a busca da solução ótima tanto via regiões factíveis quanto via regiões infactíveis, mantendo na população, e portanto transferindo informações das duas regiões, indivíduos pertencentes a ambas as regiões do espaço de busca.

Técnicas de penalização podem ser classificadas como multiplicativas ou aditivas, sendo que no primeiro caso um fator de penalização positivo é introduzido

com o objetivo de ampliar o valor da aptidão do indivíduo infactível em um problema de minimização. No segundo caso, um funcional de penalização é adicionado ao valor da função objetivo de indivíduos infactíveis.

A penalização aditiva pode ser dividida em dois grupos: interior e exterior. Na penalização aditiva interior um funcional de barreira, $B(x)$ (o qual cresce rapidamente à medida que x se aproxima do limite do domínio viável), é adicionado no cálculo da aptidão do indivíduo, como descrito na equação 3.6.

$$F_k(x) = f(x) + \frac{1}{k}B(x) \quad (3.6)$$

Na penalização aditiva exterior introduz-se um funcional de penalização, $P(x)$, tal que $P(x) = 0$ se x é viável e $P(x) > 0$ caso contrário, no caso de um problema de minimização:

$$F_k(x) = f(x) + kP(x) \quad (3.7)$$

Nos dois grupos descritos, quando $k \rightarrow \infty$, a seqüência de minimizadores do problema irrestrito, definido utilizando-se a penalização, converge para a solução do problema restrito original.

Outro ponto importante é a definição do valor da j -ésima violação de uma solução candidata $x \in \mathfrak{R}^n$ aqui adotado como:

$$v_j(x) = \begin{cases} |h_j(x)|, & \text{para uma restrição de igualdade} \\ \max\{0, g_j(x)\}, & \text{para uma restrição de desigualdade} \end{cases}$$

A formulação mais comum de funções de penalização é dada por

$$P(x) = k \sum_{j=1}^q (v_j(x))^\beta \quad (3.8)$$

onde $v(x) \in \mathfrak{R}^q$ é o vetor de violações e $q = m + p$ é o número de restrições a serem penalizadas, sendo p o número de restrições de desigualdade e m o número

de restrições de igualdade.

A definição do parâmetro k é dependente do problema e é normalmente obtida por um processo de tentativa e erro, enquanto β é frequentemente tomado igual à 2.

Existem inúmeras propostas de penalização, assim como a definição de parâmetros mais adequados, sendo a penalização uma das principais técnicas de manipulação de restrições dentro dos algoritmos evolucionistas.

Como exemplo de penalização aplicada sobre a DE, destaca-se o trabalho (Lampinen e Zelinka, 1999), onde um indivíduo inactível tem aptidão dada por

$$F(x) = (f(x) + a) \cdot \prod_{i=1}^m c_i^{b_i}$$

onde

$$c_i = \begin{cases} 1 + s_i \cdot g_i(x), & \text{se } g_i(x) > 0 \\ 1, & \text{caso contrário} \end{cases} \quad (3.9)$$

$$s_i \geq 1$$

$$b_i \geq 1$$

$$\min(f(x)) + a > 0$$

Nesse caso são necessárias constantes que regulam a penalização sofrida pelos indivíduos inactíveis, como o valor de a , o qual é usado para garantir que somente valores positivos sejam relacionados à $F(x)$. Com o intuito de eliminar tais constantes (novos parâmetros a serem definidos pelo usuário) técnicas de penalização adaptativas foram propostas.

Um procedimento de penalização adaptativa altera os parâmetros de penalização de acordo com as informações obtidas durante o processo evolutivo e foi proposto em (Bean e Alouane, 1992), onde o parâmetro de penalização $k = \lambda(t)$ é

dado por:

$$\lambda(t+1) = \begin{cases} (1/\beta_1)\lambda(t), & \text{se } b^i \in V \text{ para todo } t-g+1 \leq i \leq t \\ \beta_2\lambda(t), & \text{se } b^i \notin V \text{ para todo } t-g+1 \leq i \leq t \\ \lambda(t), & \text{caso contrário} \end{cases}$$

onde b^i é o melhor indivíduo na geração i , V é a região viável, $\beta_i \neq \beta_2$ e $\beta_1, \beta_2 > 1$. Nesse método o parâmetro de penalização da próxima geração, $\lambda(t+1)$, decai quando todos os melhores elementos nas últimas g gerações são factíveis, aumenta quando os melhores indivíduos são infactíveis e não se altera, caso contrário.

Existem diversas propostas quanto ao uso de penalização adaptativas, tais como as presentes em (Schoenauer e Xanthakis, 1993) e (Coit et al., 1996). Entretanto, neste trabalho, destacamos a técnica identificada como “Adaptive Penalty Method” - APM, proposta originalmente em (Barbosa e Lemonge, 2003).

3.5.3 A Técnica APM (Adaptive Penalty Method)

O APM ou Método de Penalização Adaptativa, (Barbosa e Lemonge, 2003), utiliza os valores de aptidão e os valores das violações das restrições na geração corrente no cálculo da penalização de cada indivíduo, de maneira que restrições com menor grau de satisfação na população (mais difíceis de serem satisfeitas) possuem um maior peso na penalização do indivíduo. O cálculo da aptidão dos indivíduos é dado pela equação 3.10.

$$F(x) = \begin{cases} f(x) & \text{se } x \text{ é válido} \\ \bar{f}(x) + \sum_{j=1}^q k_j v_j(x) & \text{caso contrário} \end{cases} \quad (3.10)$$

com

$$\bar{f}(x) = \begin{cases} f(x) & \text{se } f(x) > \langle f(x) \rangle \\ \langle f(x) \rangle & \text{caso contrário} \end{cases}$$

O coeficiente de penalização k_j correspondente à j -ésima restrição é definido a cada geração por

$$k_j = |\langle f(x) \rangle| \frac{\langle v_j(x) \rangle}{\sum_{l=1}^q [\langle v_l(x) \rangle]^2} \quad (3.11)$$

onde $\langle f(x) \rangle$ é a média da aptidão na população corrente e $\langle v_l(x) \rangle$ é a violação média da l -ésima restrição na população corrente. Um estudo mais abrangente sobre o APM é encontrado em (Barbosa e Lemonge, 2008).

O APM foi aplicado originalmente em conjunto com algoritmos genéticos, (Lemonge e Barbosa, 2004; Barbosa e Lemonge, 2005), e sua aplicação na DE é apresentada como proposta deste trabalho, sendo sua primeira descrição realizada em (Silva et al., 2008).

3.5.3.1 Variantes do APM

Algumas alterações sobre o APM foram apresentadas em (Barbosa e Lemonge, 2008), visando uma redução do esforço computacional e uma maior suavidade na variação dos coeficientes k_j . Dessa forma, as variações propõem desde a manutenção dos valores de k_j por um número maior de gerações até o uso da média entre o valor anterior e o corrente calculado, visando a suavização da variação dos coeficientes. Também considerou-se a utilização de valores monótonos dos coeficientes k_j , ou seja, não é permitido a redução do coeficiente. Sendo assim, cada coeficiente calculado na geração corrente é utilizado somente quando maior que o atual.

A seguir apresenta-se uma breve descrição sobre as variantes consideradas em (Barbosa e Lemonge, 2008).

APM esporádico

Nesta variante do APM embora o cálculo das violações das restrições, bem como a atualização dos coeficientes de penalização, k_j , sejam realizados da mesma forma, os coeficientes são mantidos por um número fixo de gerações. A intenção dessa variante é simplesmente a redução do custo computacional.

APM esporádico com acumulação das violações

Este caso é semelhante ao anterior quanto à manutenção dos valores de k_j por um determinado número de gerações. Entretanto, nesta variante, as violações das restrições são acumuladas durante as gerações nas quais os coeficientes k_j são mantidos fixos.

APM com coeficientes de penalização monótonos

O APM monótono recebe essa determinação por não permitir a redução dos valores dos coeficientes de penalização, k_j . Assim os novos valores calculados para os coeficientes são utilizados apenas quando maiores que os atuais.

APM com amortecimento

Por fim, a busca pela redução das oscilações existentes entre as atualizações dos coeficientes originou esta variante, que utiliza uma ponderação entre o valor de k_j e o novo valor calculado. O valor resultante para o coeficiente k_j pode ser obtido por:

$$k_j^{(\text{novo})} = \Theta k_j^{(\text{novo})} + (1 - \Theta) k_j^{(\text{corrente})}$$

onde $\Theta \in [0, 1]$.

3.5.4 A Técnica de Ordenação Estocástica (Stochastic Ranking)

Outra proposta, presente em um dos mais recentes trabalhos sobre o assunto (Zhang et al., 2008), utiliza um mecanismo denominado “Stochastic Ranking” proposto para seleção dos indivíduos na otimização evolucionista restrita em (Runarsson e Yao, 2000). Para isso a DE utilizada produz um número M de indivíduos para cada indivíduo alvo (a DE utilizada difere da descrita no início deste trabalho por gerar um maior número de “filhos” por indivíduo) e realiza uma comparação entre esses indivíduos e o indivíduo alvo. Os indivíduos são classificados e o melhor colocado é mantido para a próxima geração.

A probabilidade de uso do indivíduo ineficaz na classificação é dada por

P_f , a qual decai ao longo do processo segundo a expressão

$$P_f(G) = 0,45(1 - G/GEN) \quad (3.12)$$

onde G é o número da geração corrente e GEN o número máximo de gerações na evolução. Quanto ao valor 0,45, o mesmo foi estabelecido seguindo uma análise presente em (Runarsson e Yao, 2000).

Dessa forma a busca abrange um maior número de indivíduos inactíveis no início e torna-se mais restritiva ao final.

3.5.5 Tratamento dos Limites das Variáveis

Outro ponto importante no tratamento de problemas restritos, diz respeito ao ajuste das variáveis restritas a um intervalo permitido. Isso porque, na maioria dos casos, cada variável é limitada inferiormente (L_i) e superiormente (L_s).

Os valores da população inicial são gerados de forma aleatória, dentro do intervalo estabelecido para cada variável. No entanto não é possível assegurar que os valores produzidos pela mutação não excedam tais limites. Com o intuito de evitar essas violações, as mais diferentes técnicas foram empregadas para o ajuste dos valores gerados, ou seja, o valor é gerado normalmente e, posteriormente, caso não pertença ao intervalo, o valor é realocado dentro do intervalo permitido (Zhang et al., 2008; Zielinski e Laur, 2006; Onwubolu, 2004).

Uma primeira proposta é o simples uso do limite mais próximo ao valor gerado, técnica largamente empregada na computação evolucionista, descrita para DE em (Onwubolu, 2004), a qual baseia-se na idéia de que a solução ótima está freqüentemente localizada na fronteira da região factível.

Já em (Zielinski e Laur, 2006) seleciona-se o valor central entre o valor ante-

rior (valor da variável no indivíduo alvo) e o limite mais próximo:

$$u_{j,i,G+1} = \begin{cases} \frac{1}{2}(x_{j,i,G} + L_i^j), & \text{se } u_j < L_i^j \\ \frac{1}{2}(x_{j,i,G} + L_s^j), & \text{se } u_j > L_s^j \\ u_{j,i,G+1}, & \text{caso contrário} \end{cases} \quad (3.13)$$

onde L_i^j e L_s^j são os limites superior e inferior da j -ésima variável, respectivamente. Além disso, o termo $u_{j,i,G+1}$ indica a geração de uma nova parcela (variável j) sendo calculada para a geração G , com indivíduo alvo i , assim como a indicação de $x_{j,i,G}$ indica a utilização do valor existente no indivíduo alvo.

Por fim, em (Zhang et al., 2008) um novo valor aleatório é gerado dentro do intervalo, sendo portando desconsiderado o valor gerado pela mutação.

3.6 Configuração dos Parâmetros

A eficiência da aplicação de uma técnica evolucionista sobre determinado problema depende diretamente do ajuste de seus parâmetros, sendo essa afirmação extensível à DE. Assim, um dos pontos mais estudados para o aperfeiçoamento da DE diz respeito à adaptação dos seus parâmetros, em especial a amplitude da diferença utilizada na operação de mutação (F). Com isso, uma das possibilidades é a não-fixação do parâmetro F previamente, sendo seu valor definido no decorrer do processo. Além do valor de F , alguns estudos propõem também a adaptação de CR , do número de indivíduos na população e até mesmo da variante da DE utilizada no processo.

Propostas nesse sentido foram apresentadas em (Soliman e Bui, 2008; Brest et al., 2006; Qin e Suganthan, 2005) entre outros trabalhos. Outra possibilidade é a geração aleatória de F para cada geração, como proposto em (Kim et al., 2007).

Um recente estudo sobre o tema, apresentado em (Soliman e Bui, 2008), apresenta um esquema no qual os parâmetros F e CR são calculados dinamicamente seguindo uma distribuição de Cauchy. Sendo assim F é calculado para cada geração

(t) e para cada indivíduo (i) via

$$F_{i,t+1} = \begin{cases} C(\mu, \delta_{i,t+1}), & \text{se } rand_1 \leq \pi_1 \\ C(\mu, \delta_{i,t}), & \text{caso contrário} \end{cases} \quad (3.14)$$

onde $C(\mu, \delta)$ determina uma distribuição de Cauchy e

$$\delta_{i,t+1} = \delta_l + \delta_u * rand_2 \quad (3.15)$$

O parâmetro CR é calculado seguindo

$$CR_{i,t+1} = \begin{cases} rand_3, & \text{se } rand_4 \leq \pi_2 \\ CR_{i,t}, & \text{caso contrário} \end{cases} \quad (3.16)$$

onde $rand_j \in [0, 1]$, $j = 1, 2, 3, 4$ são números aleatórios, e π_1 e π_2 representam constantes para adaptar F e CR , enquanto δ_l, δ_u representam limite inferior e limite superior do parâmetro de escala δ .

Observa-se que alguns novos parâmetros são usados na técnica proposta. A fórmula geral para a função de distribuição cumulativa de Cauchy é dada por

$$F(x) = 0.5 + \frac{1}{\pi} \arctan\left(\frac{x - \mu}{\delta}\right) \quad (3.17)$$

Tal distribuição foi adotada por possuir “caudas espessas”, permitindo mudanças maiores mais freqüentemente, aumentando assim a probabilidade de maiores “saltos” que permitam a fuga de ótimos locais.

Além disso os valores δ e CR são introduzidos no cromossomo (vetor com os valores das variáveis) de cada indivíduo e dessa forma participam da evolução, sendo adaptados ao longo do processo.

Já em (Kim et al., 2007) o parâmetro F é calculado de forma aleatória, mas sofre a influência de dois valores constantes (a e b), os quais somados não

ultrapassam o valor 1. A proposta é descrita por

$$F = a + b * rand[0, 1] \quad (3.18)$$

onde $rand[0, 1]$ é uma função que retorna um valor aleatório entre 0 e 1. Assim o valor F é calculado para cada geração. O estudo afirma ainda que o melhor valor para F , na maioria dos casos, varia entre 0.4 e 1.

3.7 Seleção da Variante

A variante usada na DE pode influenciar fortemente os resultados, como mostrado em (Qing, 2008; Mezura-Montes et al., 2006a; Silva e Barbosa, 2008b). Portanto, uma importante decisão é a escolha da variante a ser implementada quando da aplicação da DE. Além disso, uma possibilidade é o ajuste da variante, ou sua modificação, durante o processo evolutivo, ou seja, permitir diferentes variantes durante o processo.

Uma das primeiras propostas nesse sentido foi apresentada em (Qin e Suganthan, 2005), que descreve um esquema de seleção entre duas variantes, durante o processo evolutivo. As duas variantes utilizadas são identificadas por “DE/rand/1/bin” e “DE/best/2/bin” dentro da proposta. O trabalho considera o esquema de seleção como uma estratégia de aprendizado, no qual as informações obtidas a cada geração de um novo indivíduo determinam a variante a ser utilizada.

A proposta define um período de aprendizado definido por um determinado número de gerações. Nesse período acumula-se valores referentes aos fracassos (nf_1 e nf_2) e sucessos (ns_1 e ns_2) de cada variante (indicadas por 1 e 2). Esses valores são calculados da seguinte forma: Uma variante é selecionada para ser utilizada, caso o vetor gerado seja superior ao vetor alvo, isto é, permaneça na população para próxima geração, o número de sucessos é acrescido, caso contrário, o número de fracassos é acrescido.

Inicialmente, as probabilidades de utilização de cada variante são idênticas, e a cada geração de um novo indivíduo seleciona-se uma das variantes aleatoriamente,

considerando as probabilidades de cada uma. No decorrer do processo, ao final de cada “período de aprendizado”, as probabilidades são recalculadas de acordo com a equação 3.19.

$$p_1 = \frac{ns_1(ns_2 + nf_2)}{ns_2(ns_1 + nf_1) + ns_1(ns_2 + nf_2)}, \quad p_2 = 1 - p_1 \quad (3.19)$$

A proposta denominada de SaDE, foi estendida em (Huang et al., 2006) para a utilização de quatro variantes e mantendo o mesmo conceito. Porém o cálculo das novas probabilidades foi alterado, sendo descrito pela equação 3.20.

$$p_i = \frac{ns_i}{ns_i + nf_i}, \quad i = 1, 2, 3, 4 \quad (3.20)$$

Outra proposta quanto à seleção da variante da DE é apresentada neste trabalho, na seção 3.8.2, onde a seleção da variante é realizada dinamicamente durante o processo utilizando-se de uma comparação direta entre as variantes consideradas, e não mais por meio de atualização de probabilidades de seleção.

3.8 Implementações Propostas

Esta seção apresenta as proposta elaboradas para este trabalho, as quais visam o aperfeiçoamento da DE quanto ao tratamento de restrições e a melhor seleção de suas variantes.

3.8.1 DE+APM

“Differential Evolution with Adaptive Penalty Method” (DE+APM), é uma nova proposta para a aplicação da evolução diferencial em problemas restritos. A metodologia proposta pela DE é aplicada normalmente embora a avaliação dos indivíduos leve em conta a penalização dada pelo método adaptativo descrito na seção 3.5.3.

Assim, a técnica de penalização APM, que apresenta bons resultados junto aos algoritmos genéticos, foi implementada junto à DE, mantendo na população in-

divíduos factíveis e infactíveis, bem como estabelecendo um método de comparação entre eles, na tentativa de aprimorar os resultados finais obtidos.

Foram considerados duas variantes do APM, descritas na seção 3.5.3.1: a versão original e o APM monótono, na qual não é permitida uma redução dos coeficientes de penalização durante o processo. Neste trabalho as propostas mencionadas são referenciadas como DE+APM para o uso conjunto da evolução diferencial e da formulação monótona e por DE+APM* para o uso conjunto da evolução diferencial e a formulação original do APM.

Alguns dos resultados obtidos com a proposta foram apresentados em (Silva et al., 2008) e (Silva e Barbosa, 2008b), e mostram-se competitivos na otimização de treliças.

Tais resultados, bem como novos experimentos são apresentados no capítulo 4.

3.8.2 DUVDE+APM

“Dynamic Use of Variants for Differential Evolution with Adaptive Penalty Method” (DUVDE+APM) é a segunda proposta deste trabalho, sendo ela uma extensão da primeira, já que apresenta um mecanismo de seleção dinâmica de variantes da DE. Assim, a melhor variante para o problema é encontrada durante o processo evolutivo, e a mesma pode ser modificada de acordo com as informações oriundas do processo.

O esquema proposto possui, assim como a proposta descrita em (Qin e Suganthan, 2005), uma etapa de aprendizado, no qual o processo busca a melhor variante para o problema no atual momento da evolução. Entretanto, propõe-se que a seleção da variante seja realizada por meio de uma comparação direta entre as mesmas, ou seja, um novo indivíduo é gerado para cada variante considerada, sendo escolhida a variante que gerou o melhor indivíduo.

Foram consideradas quatro variantes da DE: DE/rand/1/bin, DE/best/1/bin, DE/target-to-best/1/bin e a proposta por (Mezura-Montes et al., 2006b), descritas

na seção 3.3.

Dessa forma, são gerados quatro indivíduos por vez, e não mais um como o procedimento básico, (para cada variante, um indivíduo) e os mesmos competem entre si.

Os indivíduos carregam em seu “cromossomo” o conjunto de valores componentes da solução e a informação da variante que deu origem ao mesmo.

É fácil perceber que o processo proposto, apesar de ampliar as possibilidades de busca, amplia também o esforço computacional necessário, dado principalmente pelo aumento do número de avaliações da função objetivo. Assim, após um determinado número de gerações uma das variantes é selecionada para ser utilizada com exclusividade durante um período do processo.

A variante selecionada para ser utilizada durante o novo período do processo evolutivo é aquela que gerou o melhor indivíduo na população corrente. Sendo esse período aplicado como o processo básico da DE, utilizando-se da variante selecionada no operador de mutação.

O processo evolutivo é dividido da seguinte forma: Os primeiros 10% do número total de gerações são utilizados como “período de aprendizado”, ou seja, são gerados quatro indivíduos para cada indivíduo alvo, como descrito anteriormente. Os subseqüentes 30% são realizados de forma tradicional da DE, utilizando-se da variante selecionada no período de aprendizado. Após esses dois estágios o processo é repetido, ou seja, mais 10% das gerações são aplicados como aprendizado, seguidos de mais 30% de processo normal. Por fim, o período de aprendizado é realizado mais uma vez, pelo mesmo número de gerações definidos para esse estágio e posteriormente o processo é finalizado com o processo tradicional, novamente utilizando-se da variante selecionada após o período de aprendizado.

Com o esquema descrito anteriormente, 30% das gerações executadas são utilizadas como período de aprendizado e por 70% das gerações o processo é o normal da DE. Esta divisão possibilita o ganho de qualidade das soluções sem afetar fortemente o custo computacional, como mostrado pelos experimentos apresentados

no Capítulo 4.

Além disso, de maneira a realizar experimentos comparativos entre as técnicas, o número de avaliações da função de aptidão pode ser mantido constante, ajustando-se o número de gerações necessárias para a nova proposta, fixando-se um novo número de indivíduos na população.

Sendo assim, dado um número de avaliações da função de aptidão, NA, e fixando-se um determinado número de indivíduos na população, POP, calcula-se o número de gerações, GEN, por

$$\text{GEN} = \left\lceil \frac{\text{NA}}{0,3(4 \times \text{POP}) + 0,7(\text{POP})} \right\rceil \quad (3.21)$$

Nesta proposta, após o período de aprendizado, seleciona-se a variante que gerou o melhor indivíduo da população. Entretanto, outra possibilidade é selecionar a variante que obteve maior sucesso durante esse período, ou seja, selecionar a que gerou um maior número de indivíduos que permaneceram na população. Essa proposta é identificada por DUVDE2+APM.

Capítulo 4

Experimentos Numéricos

Com a intenção de verificar experimentalmente o desempenho das propostas apresentadas no capítulo anterior, apresentam-se aqui os diversos testes realizados, com diferentes configurações e propostas sobre a DE em problemas de otimização restrita.

Os experimentos realizados são divididos em dois grupos, sendo o primeiro composto por 11 conhecidos problemas-teste para algoritmos evolucionistas. A partir daí foram focados os problemas estruturais correspondentes à minimização dos custos de uma viga soldada, um vaso de pressão e principalmente a minimização do peso de estruturas treliçadas.

4.1 Objetivos e Organização

Os objetivos dos experimentos aqui apresentados foram a comprovação da qualidade da aplicação da evolução diferencial sobre os problemas considerados, a averiguação do aprimoramento da técnica pelas propostas apresentadas e a análise do desempenho do algoritmo quando utilizado em conjunto com as propostas já mencionadas.

Entretanto, dado o grande número de combinações possíveis entre as propostas apresentadas neste trabalho, uma organização dos experimentos tornou-se necessária.

Dessa forma, o primeiro conjunto de problemas, composto por 11 funções

teste, é utilizado para a validação das propostas apresentadas. Sendo assim, diversas combinações, envolvendo todas as técnicas citadas anteriormente, são aplicadas, seguindo o seguinte critério.

Para cada problema teste, quatro etapas foram realizadas, a primeira etapa consiste na avaliação das variantes apresentadas, a segunda etapa na avaliação do tratamento de limites das variáveis, seguida da terceira etapa pela avaliação dos parâmetros empregados e, por fim, pela avaliação do tratamento de restrições implementado.

Assim, na primeira etapa avaliou-se todas as variantes propostas, fixando-se o restante do algoritmo com o tratamento de limites apresentado em (Onwubolu, 2004), onde utiliza-se o limite mais próximo quando um valor é gerado fora do intervalo permitido, restrições manipuladas pelo método apresentado por Deb (Deb, 2000) e com parâmetros fixados no início do processo (aqui $F = 0,8$ e $CR = 0,9$).

Após essa primeira etapa a variante que produziu, em média, os melhores resultados foi utilizada nas etapas subsequentes. Dessa forma, a segunda etapa avalia todas as variantes de tratamento de limites citados neste trabalho, utilizando a variante selecionada na etapa anterior e as demais variantes fixadas na primeira etapa.

Na terceira etapa foram utilizadas a variante selecionada na primeira etapa, o tratamento de limites selecionado na segunda etapa, e avaliou-se as propostas de parâmetros (estáticos ou dinâmicos), sendo a manipulação de restrições mantida como a proposta por Deb (Deb, 2000).

Na quarta e última etapa avaliam-se todas as variantes de manipulação de restrições citadas aqui, utilizando as demais variantes selecionadas nas etapas anteriores. Além disso, nesse ponto são incluídas as propostas desenvolvidas neste trabalho, para a comparação final entre os métodos.

Com base nos resultados obtidos, um conjunto de configurações foi selecionado para a continuidade dos experimentos, ou seja, as configurações que obtiveram melhor desempenho foram utilizadas nos demais experimentos realizados.

4.2 Grupo de Problemas

Como dito anteriormente, um conjunto de problemas denominado “G-Suite” (Koziel e Michalewicz, 1999), com 11 problemas teste, utilizado para a validação inicial das propostas apresentadas. Sendo assim, um conjunto de configurações e técnicas foram combinadas e selecionadas por meio de diversas execuções das implementações sobre os problemas componentes do conjunto estudado, e por fim, as melhores combinações foram comparadas às propostas deste trabalho, indicadas por DE+APM*, DE+APM, DUVDE+APM e DUVDE2+APM, apresentadas na seção 3.8. Além disso, este experimento fornece base para a seleção de configurações para a evolução diferencial para os demais experimentos realizados.

Como já descrito no início do capítulo, este experimento foi dividido em quatro etapas, sendo assim, a primeira etapa consiste na avaliação das diferentes variantes mencionadas durante este trabalho fixando-se as demais configurações da evolução diferencial. Com isso, o algoritmo base utilizado faz uso de um tratamento de limites apresentado por (Onwubolu, 2004), no qual, quando o intervalo permitido é ultrapassado, o valor gerado é substituído pelo limite mais próximo, bem como faz uso de parâmetros fixados no início do processo, e um mecanismo de manipulação de restrições apresentado por (Deb, 2000).

Foram realizadas 20 execuções independentes para cada problema e cada configuração, sendo que cada execução utilizou-se de 5000 gerações e 70 indivíduos, exceto para o caso das propostas DUVDE+APM ou DUVDE2+APM, nesses casos, fixando-se a população em 40 indivíduos e seguindo a equação 3.21, foram realizadas 4605 gerações.

As funções utilizadas são apresentadas na Tabela 4.1, os intervalos aos quais as variáveis estão submetidas na Tabela 4.2 e suas características resumidas na Tabela 4.3, sendo que “opt” indica minimização ou maximização, “n” o número de variáveis envolvidas, f^* o melhor valor conhecido na literatura, e “m” o número de restrições (sem considerar as restrições de limites das variáveis de projeto).

Os resultados obtidos na primeira etapa do experimento são apresentados

Tabela 4.1: Definição das funções teste

Função	Restrições
$G_1(x) = 5x_1 + 5x_2 + 5x_3 + 5x_4 - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i$	$\begin{aligned} 2x_1 + 2x_2 + x_10 + x_11 - 10 &\leq 0 \\ 2x_1 + 2x_3 + x_10 + x_12 - 10 &\leq 0 \\ 2x_2 + 2x_3 + x_11 + x_12 - 10 &\leq 0 \\ -2x_4 - x_5 + x_10 &\leq 0, \quad -8x_1 + x_10 \leq 0 \\ -2x_6 - x_7 + x_11 &\leq 0, \quad -8x_3 + x_12 \leq 0 \end{aligned}$
$G_2(x) = \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n ix_i^2}}$	$\begin{aligned} 0,75 - \prod_{i=1}^n x_i &\leq 0 \\ \sum_{i=1}^n x_i - 7,5n &\leq 0 \end{aligned}$
$G_3(x) = (\sqrt{n})^n \prod_{i=1}^n x_i$	$\sum_{i=1}^n x_i^2 - 1 = 0$
$G_4(x) = 5,3578547x_3^2 + 0,8356891x_1x_5 + 37,293239x_1 - 40792,141$	$\begin{aligned} 0 &\leq 85,334407 + 0,0056858x_2x_5 \\ &\quad + 0,0006262x_1x_4 - 0,0022053x_3x_5 \\ &\leq 92 \\ 90 &\leq 80,51249 + 0,0071317x_2x_5 \\ &\quad + 0,0029955x_1x_4 - 0,0021813x_3^2 \\ &\leq 110 \\ 20 &\leq 9,300961 + 0,0047026x_3x_5 \\ &\quad + 0,0012547x_1x_3 - 0,0019085x_3x_4 \\ &\leq 25 \end{aligned}$
$G_5(x) = 3x_1 + 0,000001x_1^3 + 2x_2 + 0,000002/3x_2^3$	$\begin{aligned} x_4 - x_3 + 0,55 &\geq 0, \quad x_3 - x_4 + 0,55 \geq 0 \\ 1000\text{sen}(-x_3 - 0,25) \\ +1000\text{sen}(-x_4 - 0,25) + 894,8 - x_1 &= 0 \\ 1000\text{sen}(x_3 - 0,25) \\ +1000\text{sen}(x_3 - x_4 - 0,25) + 894,8 - x_2 &= 0 \\ 1000\text{sen}(x_4 - 0,25) \\ +1000\text{sen}(x_4 - x_3 - 0,25) + 1294,8 &= 0 \end{aligned}$
$G_6(x) = (x_1 - 10)^3 + (x_2 - 20)^3$	$\begin{aligned} -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 &\leq 0 \\ (x_1 - 6)^2 - (x_2 - 5)^2 - 82.81 &\leq 0 \end{aligned}$
$G_7(x) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45$	$\begin{aligned} -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 &\leq 0 \\ 10x_1 - 8x_2 - 17x_7 + 2x_8 &\leq 0 \\ -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 &\leq 0 \\ 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 &\leq 0 \\ 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 &\leq 0 \\ x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 &\leq 0 \\ 0,5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 &\leq 0 \\ -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} &\leq 0 \end{aligned}$
$G_8(x) = \frac{\text{sen}^3(2\pi x_1)\text{sen}(2\pi x_2)}{x_1^3(x_1+x_2)}$	$\begin{aligned} x_1^2 - x_2 + 1 &\leq 0 \\ 1 - x_1 + (x_2 - 4)^2 &\leq 0 \end{aligned}$
$G_9(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$	$\begin{aligned} 127 - 2x_1^2 - 3x_2^4 - x_3 - 4x_4^2 - 5x_5 &\geq 0 \\ 282 - 7x_1 - 3x_2 - 10x_3^2 - x_4 + x_5 &\geq 0 \\ 196 - 23x_1 - x_2^2 - 6x_6^2 + 8x_7 &\geq 0 \\ -4x_1^2 - x_2^2 + 3x_1x_2 - 2x_3^2 - 5x_6 + 11x_7 &\geq 0 \end{aligned}$
$G_{10}(x) = x_1 + x_2 + x_3$	$\begin{aligned} -1 + 0,0025(x_4 + x_6) &\leq 0 \\ -1 + 0,0025(x_5 + x_7 - x_4) &\leq 0 \\ -1 + 0,01(x_8 - x_5) &\leq 0 \\ -x_1x_6 + 833,33252x_4 + 100x_1 - 83333.333 &\leq 0 \\ -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 &\leq 0 \\ -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 &\leq 0 \end{aligned}$
$G_{11}(x) = x_1^2 + (x_2 - 1)^2$	$x_2 - x_1^2 = 0$

Tabela 4.2: Definição dos limites das variáveis para as funções da G-Suite

Função	Limites
G_1	$0 \leq x_i \leq 1 (i = 1, \dots, 9), 0 \leq x_i \leq 100 (i = 10, 11, 12), 0 \leq x_{13} \leq 1$
G_2	$0 \leq x_i \leq 10 (i = 1, \dots, n) n = 20$
G_3	$0 \leq 1 (i = 1, \dots, n) n = 10$
G_4	$78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45, 27 \leq x_i \leq 45 (i = 3, 4, 5)$
G_5	$0 \leq x_1, x_2 \leq 1200, -0, 55 \leq x_3, x_4 \leq 0, 55$
G_6	$13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100$
G_7	$-10 \leq x_i \leq 10 (i = 1, \dots, 10)$
G_8	$0 \leq x_1, x_2 \leq 10$
G_9	$-10 \leq x_i \leq 10 (i = 1, \dots, 7)$
G_{10}	$100 \leq x_1 \leq 10000, 1000 \leq x_2, x_3 \leq 10000, 10 \leq x_i \leq 1000 (i = 4, \dots, 8)$
G_{11}	$-1 \leq x_1, x_2 \leq 1$

Tabela 4.3: Resumo das 11 funções da G-Suite

“opt” indica minimização ou maximização, “n” o número de variáveis envolvidas, f^* o melhor valor conhecido na literatura, e “m” o número de restrições (sem considerar as restrições de limites das variáveis de projeto).

Nome	opt	n	Tipo de $f(x)$	f^*	m
G_1	min	13	Quadrática	-15.0	9
G_2	max	20	Não linear	0.803553	2
G_3	max	10	Polinomial	1.0	1
G_4	min	5	Quadrática	-30665.5	6
G_5	min	4	Cúbica	5126.4981	5
G_6	min	2	Cúbica	-6961.8	2
G_7	min	10	Quadrática	24.306	8
G_8	max	2	Não linear	0.0958250	2
G_9	min	7	Polinomial	680.63	4
G_{10}	min	8	Linear	7049.25	6
G_{11}	min	2	Quadrática	0.75	1

nas Tabelas 4.4 e 4.5. São apresentados os resultados referentes ao melhor, pior e à média dos valores viáveis encontrados ao final das execuções, assim como o correspondente desvio padrão.

Afim de simplificar a exibição dos resultados, as variantes descritas na seção 3.3 são referenciadas simplesmente como “Rand”, “Mezura-Montes”, “Best-to-target”, “Best” e “SSaDE” respectivamente para DE/1/rand/bin, a proposta formulada por Mezura-Montes et al. (Mezura-Montes et al., 2006b), DE/1/best-to-target/bin, DE/1/best/bin e uma modificação da proposta de seleção adaptativa da variante, (Huang et al., 2006), descrita na seção 3.7. A variante identificada por “SSaDE” utiliza o critério de seleção da variante proposto em (Huang et al., 2006), o qual atribui probabilidades para a aplicação de cada uma das quatro variantes trabalhadas e ajusta essas probabilidades ao longo do processo, entretanto simplifica a proposta original por desconsiderar sua seleção dos parâmetros F e CR , focando apenas a seleção das variantes.

Os resultados da primeira etapa deste experimento (Tabelas 4.4 e 4.5) mostraram que a formulação original da DE, ou seja, a variante “Rand” obteve o melhores desempenho na média. Entretanto, para os problemas G_2 , G_3 , G_{10} e G_{11} outras variantes obtiveram melhores resultados, portanto, seguindo a organização deste experimento, as próximas etapas utilizam-se das seguintes variantes para cada problema:

- G_2 e G_3 , variante identificada por “Mezura-Montes”;
- G_{10} , variante identificada por “Best-to-target”;
- G_{11} , variante identificada por “Best”;
- Para os demais problemas a variante utilizada foi a identificada por “Rand”.

A segunda etapa consiste na avaliação das três propostas apresentadas na seção 3.5.5 quanto ao tratamento dos limites impostos sobre as variáveis envolvidas em cada problema, identificadas aqui como: “Próximo”, (Onwubolu, 2004), na qual o limite mais próximo ao valor gerado é utilizado. “Central”, (Zielinski e Laur,

Tabela 4.4: Resultado da primeira etapa do experimento 1, $G_1 - G_5$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_1				
Rand	-15	-14,10	-12	1,37
Mezura-Montes	-15	-9,45	-6	1,96
Best-to-target	-15	-11,42	-6	2,62
Best	-12	-9,57	-6	1,95
SSaDE	-12	-9,24	-6	1,96
Problema G_2				
Rand	0,793491	0,664171	0,514558	0,085626
Mezura-Montes	0,794897	0,725664	0,531591	0,062468
Best-to-target	0,309989	0,258663	0,223174	0,029334
Best	0,735993	0,562147	0,352932	0,113245
SSaDE	0,794897	0,585032	0,330613	0,140238
Problema G_3				
Rand	0,182	0,050	0,0	0,062
Mezura-Montes	0,681	0,146	0,0	0,189
Best-to-target	0,512	0,089	0,0	0,147
Best	1,000	0,096	0,0	0,232
SSaDE	0,040	0,002	0,0	0,008
Problema G_4				
Rand	-30665,5	-30665,5	-30665,5	0,0
Mezura-Montes	-30665,5	-30665,5	-30665,5	0,0
Best-to-target	-30665,5	-30665,5	-30665,5	0,0
Best	-30665,5	-30665,5	-30665,5	0,0
SSaDE	-30665,5	-30665,5	-30665,5	0,0
Problema G_5				
Rand	5126,49	5126,49	5126,49	0,0
Mezura-Montes	5126,49	5290,78	6112,22	367,36
Best-to-target	5126,79	5415,47	5943,95	292,60
Best	5126,49	5282,14	6112,22	359,43
SSaDE	5126,49	5126,49	5126,49	0,0

Tabela 4.5: Resultado da primeira etapa do experimento 1, $G_6 - G_{11}$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_6				
Rand	-6961,8	-6961,8	-6961,8	0,0
Mezura-Montes	-6961,8	-6961,8	-6961,8	0,0
Best-to-target	-6961,8	-6961,8	-6961,8	0,0
Best	-6961,8	-6628,6	-4174,7	790,0
SSaDE	-6961,8	-6961,8	-6961,8	0,0
Problema G_7				
Rand	24,306	24,306	24,306	0,0
Mezura-Montes	24,306	35,467	121,747	28,811
Best-to-target	24,309	24,383	25,006	0,208
Best	24,306	24,810	27,602	0,977
SSaDE	24,306	25,178	29,059	1,448
Problema G_8				
Rand	0,0958250	0,0958250	0,0958250	0,0
Mezura-Montes	0,0958250	0,0958250	0,0958250	0,0
Best-to-target	0,0958250	0,0958250	0,0958250	0,0
Best	0,0958250	0,0958250	0,0958250	0,0
SSaDE	0,0958250	0,0958250	0,0958250	0,0
Problema G_9				
Rand	680,63	680,63	680,63	0,0
Mezura-Montes	680,63	680,63	680,63	0,0
Best-to-target	680,63	680,63	680,63	0,0
Best	680,63	680,63	680,63	0,0
SSaDE	680,63	680,63	680,63	0,0
Problema G_{10}				
Rand	7049,25	7069,42	7250,96	60,51
Mezura-Montes	7049,25	8020,81	11100,00	1370,04
Best-to-target	7049,54	7055,04	7078,33	6,61
Best	7049,25	7352,64	11100,00	865,28
SSaDE	7049,25	7170,28	7250,97	98,82
Problema G_{11}				
Rand	1,0	1,0	1,0	0,0
Mezura-Montes	0,76	0,98	1,0	0,05
Best-to-target	1,0	1,0	1,0	0,0
Best	0,75	0,97	1,0	0,07
SSaDE	1,0	1,0	1,0	0,0

2006), em que um valor entre o limite mais próximo e o valor anterior da variável, dentro do intervalo permitido, é selecionado para substituir o novo valor gerado, o qual não respeita o intervalo permitido. E por fim, “Aleatório”, (Zhang et al., 2008), na qual um valor aleatório, dentro do intervalo permitido, é selecionado para substituir o valor gerado.

Os resultados da segunda etapa são apresentados na Tabela 4.6. E mostram claramente o bom desempenho da variante identificada aqui como “Aleatória”, sendo que, nos problemas em que existem diferenças nos resultados, esta variante obteve os melhores resultados, considerando-se a média. Entretanto, para os problemas em que não ocorreram diferenças, a proposta anterior foi mantida para a próxima etapa (favorecendo as propostas de maior simplicidade de aplicação), conseqüentemente, a configuração quanto a manipulação de restrições de limites das variáveis para cada problema foi estabelecida da seguinte forma: G_4 , G_6 , G_7 , G_8 e G_9 , tratamento de limites seguindo a proposta aqui identificada por “Próximo”. Os demais problemas utilizaram a proposta identificada como “Aleatório”.

A terceira etapa objetiva avaliar as propostas quanto à seleção dos parâmetros F e Cr do algoritmos. Neste ponto do experimento, as três variantes avaliadas (descritas na seção 3.6) são identificadas como: “Fixos”, na qual os parâmetros são fixados no início do processo. “Aleatórios”, seguindo (Kim et al., 2007) e “Dinâmicos”, (Soliman e Bui, 2008). Os resultados da terceira etapa são apresentados na Tabela 4.7.

A terceira etapa deste experimento apresenta as variantes identificadas por “Fixos” e “Aleatórios” como as de maior relevância, com destaque para esta última, possuindo os melhores resultados, em média, na maioria dos problemas que apresentaram diferenças entre os resultados. Observa-se que o desempenho produzido pela variante de cálculo dinâmico dos parâmetros, apesar de obter bons resultados em determinados casos, como no problema G_3 no qual obtém a solução ótima conhecida $(1, 0)$, mostra resultados instáveis, na média, para diversos outros casos, como por exemplo em G_7 e G_{10} . Com base nos resultados da terceira etapa

Tabela 4.6: Resultado da segunda etapa do experimento 1, $G_1 - G_{11}$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_1				
Próximo	-15	-14, 10	-12	1, 37
Central	-15	-14, 10	-12	1, 37
Aleatório	-14, 99	-14, 99	-14, 99	0, 00
Problema G_2				
Próximo	0, 794897	0, 725663	0, 531509	0, 062487
Central	0, 792608	0, 719219	0, 571366	0, 050351
Aleatório	0, 803553	0, 783252	0, 760764	0, 012410
Problema G_3				
Próximo	0, 681	0, 146	0, 0	0, 189
Central	0, 434	0, 138	0, 0	0, 148
Aleatório	0, 842	0, 642	0, 400	0, 112
Problema G_4				
Próximo	-30665, 5	-30665, 5	-30665, 5	0, 0
Central	-30665, 5	-30665, 5	-30665, 5	0, 0
Aleatório	-30665, 5	-30665, 5	-30665, 5	0, 0
Problema G_5				
Próximo	5126, 49	5126, 49	5126, 49	0, 0
Central	5126, 49	5126, 60	5128, 74	0, 49
Aleatório	5126, 49	5126, 49	5126, 49	0, 0
Problema G_6				
Próximo	-6961, 8	-6961, 8	-6961, 8	0, 0
Central	-6961, 8	-6961, 8	-6961, 8	0, 0
Aleatório	-6961, 8	-6961, 8	-6961, 8	0, 0
Problema G_7				
Próximo	24, 306	24, 306	24, 306	0, 0
Central	24, 306	24, 306	24, 306	0, 0
Aleatório	24, 306	24, 306	24, 306	0, 0
Problema G_8				
Próximo	0, 0958250	0, 0958250	0, 0958250	0, 0
Central	0, 0958250	0, 0958250	0, 0958250	0, 0
Aleatório	0, 0958250	0, 0958250	0, 0958250	0, 0
Problema G_9				
Próximo	680, 63	680, 63	680, 63	0, 0
Central	680, 63	680, 63	680, 63	0, 0
Aleatório	680, 63	680, 63	680, 63	0, 0
Problema G_{10}				
Próximo	7049, 54	7055, 03	7078, 33	6, 61
Central	7049, 98	7052, 90	7061, 82	2, 73
Aleatório	7049, 82	7051, 99	7062, 77	2, 98
Problema G_{11}				
Próximo	0, 75	0, 97	1, 0	0, 07
Central	1, 0	1, 0	1, 0	0, 0
Aleatório	0, 75	0, 76	0, 81	0, 01

Tabela 4.7: Resultado da terceira etapa do experimento 1, $G_1 - G_{11}$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_1				
Fixos	-14,99	-14,99	-14,99	0,00
Aleatórios	-15	-15	-15	0,0
Dinâmicos	-15	-14,88	-13,82	0,35
Problema G_2				
Fixos	0,803553	0,783252	0,760764	0,012410
Aleatórios	0,803553	0,780007	0,728531	0,018855
Dinâmicos	0,803553	0,777283	0,70754	0,022674
Problema G_3				
Fixos	0,842	0,642	0,400	0,112
Aleatórios	0,961	0,754	0,555	0,113
Dinâmicos	1,000	0,715	0,255	0,234
Problema G_4				
Fixos	-30665,5	-30665,5	-30665,5	0,0
Aleatórios	-30665,5	-30665,5	-30665,5	0,0
Dinâmicos	-30665,5	-30665,5	-30665,5	0,0
Problema G_5				
Fixos	5126,49	5126,49	5126,49	0,0
Aleatórios	5126,49	5126,49	5126,49	0,0
Dinâmicos	5130,37	5218,90	5431,67	106,04
Problema G_6				
Fixos	-6961,8	-6961,8	-6961,8	0,0
Aleatórios	-6961,8	-6961,8	-6961,8	0,0
Dinâmicos	-6961,8	-6961,8	-6961,8	0,0
Problema G_7				
Fixos	24,306	24,306	24,306	0,0
Aleatórios	24,306	24,306	24,306	0,0
Dinâmicos	24,306	24,396	25,460	0,278
Problema G_8				
Fixos	0,0958250	0,0958250	0,0958250	0,0
Aleatórios	0,0958250	0,0958250	0,0958250	0,0
Dinâmicos	0,0958250	0,0958250	0,0958250	0,0
Problema G_9				
Fixos	680,63	680,63	680,63	0,0
Aleatórios	680,63	680,63	680,63	0,0
Dinâmicos	680,63	680,63	680,63	0,0
Problema G_{10}				
Fixos	7049,82	7051,99	7062,77	2,98
Aleatórios	7049,32	7049,51	7049,95	0,15
Dinâmicos	7054,18	7138,92	7332,66	79,72
Problema G_{11}				
Fixos	0,75	0,76	0,81	0,01
Aleatórios	0,75	0,81	0,97	0,08
Dinâmicos	0,75	0,80	0,99	0,08

e considerando que nos casos em que não ocorreram diferenças nos resultados a implementação anterior foi mantida, a configuração quanto a seleção dos parâmetros para os problemas tratados foi estabelecida da seguinte forma: G_1 , G_3 , G_7 , G_9 e G_{10} com a proposta identificada por “Aleatórios” e demais problemas com a fixação dos parâmetros no início do processo, “Fixos”. É importante ressaltar que as configurações selecionadas nas etapas anteriores são mantidas, como descrito na organização deste experimento.

A quarta e última etapa deste experimento visa a avaliação dos mecanismos de manipulação de restrições apresentados neste trabalho, bem como a validação das propostas realizadas, comparando-as com as combinações formadas após as quatro sucessivas seleções realizadas, uma a cada etapa. Nesta etapa as propostas são identificadas como “Deb”, “DE+APM*”, “Lampinem”, “Ranking”, “DE+APM”, “DUVDE+APM”, “DUVDE2+APM” respectivamente referentes ao mecanismo proposto em (Deb, 2000), a proposta descrita na seção 3.8.1 utilizando-se a formulação original do APM, o uso da penalização proposta em (Lampinen e Zelinka, 1999), o uso de um “stochastic ranking” como em (Zhang et al., 2008), a proposta apresentada na seção 3.8.1 utilizando-se da variante monótona do APM, e as duas propostas de seleção dinâmica da variante em conjunto com a versão monótona do APM.

Os resultados da quarta etapa são apresentados nas Tabelas 4.8, 4.9 e 4.10. Os campos indicados por – indicam que não foram encontradas soluções viáveis ao final do processo.

Com os resultados finais deste experimento pode-se concluir que as propostas apresentadas neste trabalho são válidas, isso porque levaram a resultados competitivos com as demais variantes, sendo que na maioria dos casos obtiveram os melhores resultados. Entretanto, uma das mais importantes observações deve-se ao desempenho das propostas denominadas DUVDE+APM e DUVDE2+APM, dado que as mesmas obtiveram desempenho comparável às demais, mesmo considerando que as outras implementações agregaram configurações selecionadas espe-

Tabela 4.8: Resultado da quarta etapa do experimento 1, $G_1 - G_4$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_1				
Deb	-15	-15	-15	0,0
DE+APM*	-15	-15	-15	0,0
Lampinem	-	-	-	-
Ranking	-15	-15	-15	0,0
DE+APM	-15	-15	-15	0,0
DUVDE+APM	-15	-14,76	-12,65	0,59
DUVDE2+APM	-15	-14,88	-13,82	0,35
Problema G_2				
Deb	0,803553	0,783252	0,760764	0,012410
DE+APM*	0,803553	0,780876	0,740167	0,017645
Lampinem	-	-	-	-
Ranking	0,801716	0,772081	0,682034	0,032348
DE+APM	0,803553	0,779333	0,753214	0,014395
DUVDE+APM	0,803553	0,759774	0,611783	0,043368
DUVDE2+APM	0,803553	0,733557	0,616608	0,058322
Problema G_3				
Deb	0,961	0,754	0,555	0,113
DE+APM*	1,000	0,987	0,925	0,025
Lampinem	-	-	-	-
Ranking	1,000	1,000	1,000	0,0
DE+APM	1,000	0,913	0,756	0,072
DUVDE+APM	1,000	1,000	1,000	0,0
DUVDE2+APM	1,000	1,000	1,000	0,0
Problema G_4				
Deb	-30665,5	-30665,5	-30665,5	0,0
DE+APM*	-30665,5	-30665,5	-30665,5	0,0
Lampinem	-	-	-	-
Ranking	-32217,4	-32169,7	-32037,5	55,2
DE+APM	-30665,5	-30665,5	-30665,5	0,0
DUVDE+APM	-30665,5	-30665,5	-30665,5	0,0
DUVDE2+APM	-30665,5	-30665,5	-30665,5	0,0

Tabela 4.9: Resultado da quarta etapa do experimento 1, $G_5 - G_8$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_5				
Deb	5126,49	5126,49	5126,49	0,0
DE+APM*	5126,49	5126,49	5126,49	0,0
Lampinem	–	–	–	–
Ranking	5126,49	5126,49	5126,49	0,0
DE+APM	5126,49	5126,49	5126,49	0,0
DUVDE+APM	5126,49	5126,49	5126,49	0,0
DUVDE2+APM	5126,49	5126,49	5126,49	0,0
Problema G_6				
Deb	–6961,8	–6961,8	–6961,8	–6961,8
DE+APM*	–6961,8	–6961,8	–6961,8	–6961,8
Lampinem	–	–	–	–
Ranking	–6961,8	–6961,8	–6961,8	–6961,8
DE+APM	–6961,8	–6961,8	–6961,8	–6961,8
DUVDE+APM	–6961,8	–6961,8	–6961,8	–6961,8
DUVDE2+APM	–6961,8	–6961,8	–6961,8	–6961,8
Problema G_7				
Deb	24,306	24,306	24,306	0,0
DE+APM*	24,306	24,306	24,306	0,0
Lampinem	735,90	1894,892	3823,97	1373,277
Ranking	24,306	24,306	24,306	0,0
DE+APM	24,306	24,306	24,306	0,0
DUVDE+APM	24,306	24,306	24,306	0,0
DUVDE2+APM	24,306	24,508	27,602	0,725
Problema G_8				
Deb	0,0958250	0,0958250	0,0958250	0,0
DE+APM*	0,0958250	0,0958250	0,0958250	0,0
Lampinem	–	–	–	–
Ranking	0,0958250	0,0958250	0,0958250	0,0
DE+APM	0,0958250	0,0958250	0,0958250	0,0
DUVDE+APM	0,0958250	0,0958250	0,0958250	0,0
DUVDE2+APM	0,0958250	0,0958250	0,0958250	0,0

Tabela 4.10: Resultado da quarta etapa do experimento 1, $G_9 - G_{11}$

Variante	Melhor	Média	Pior	Desvio Padrão
Problema G_9				
Deb	680, 63	680, 63	680, 63	0, 0
DE+APM*	680, 63	680, 63	680, 63	0, 0
Lampinem	680, 63	680, 63	680, 63	0, 0
Ranking	680, 63	680, 63	680, 63	0, 0
DE+APM	680, 63	680, 63	680, 63	0, 0
DUVDE+APM	680, 63	680, 63	680, 63	0, 0
DUVDE2+APM	680, 63	680, 63	680, 63	0, 0
Problema G_{10}				
Deb	7049, 32	7049, 51	7049, 95	0, 15
DE+APM*	–	–	–	–
Lampinem	8826, 91	11488, 66	16583, 70	2110, 90
Ranking	7274, 09	7346, 57	7461, 90	54, 49
DE+APM	7049, 32	7049, 49	7050, 27	0, 21
DUVDE+APM	7049, 25	7049, 25	7049, 25	0, 0
DUVDE2+APM	7049, 25	7049, 25	7049, 25	0, 0
Problema G_{11}				
Deb	0, 750	0, 761	0, 810	0, 01
DE+APM*	0, 750	0, 750	0, 750	0, 0
Lampinem	0, 753	0, 753	0, 753	0, 0
Ranking	0, 750	0, 750	0, 750	0, 0
DE+APM	0, 750	0, 750	0, 750	0, 0
DUVDE+APM	0, 750	0, 755	0, 847	0, 02
DUVDE2+APM	0, 750	0, 750	0, 750	0, 0

cificamente para cada problema.

Os próximos experimentos foram organizados de acordo com os resultados obtidos nesta avaliação, ou seja, as configurações utilizadas e as técnicas selecionadas para comparação são baseadas nos resultados obtidos neste experimento. Sendo assim, os demais experimentos utilizaram como implementações do algoritmo da evolução diferencial as propostas que apresentaram os resultados mais relevantes.

Seguindo essa idéia, a implementação da evolução diferencial básica utilizada foi composta por:

- Variante original da evolução diferencial, DE/rand/1/bin;
- Tratamento quanto aos limites das variáveis aqui identificado como “Aleatório”, (Zhang et al., 2008);
- Parâmetros selecionados aleatoriamente, (Kim et al., 2007).

Como manipulação das restrições foram utilizados:

- O critério de seleção de Deb, (Deb, 2000);
- As propostas apresentadas neste trabalho, DE+APM, DUVDE+APM e DUVDE2+APM, todas utilizando-se da variante monótona do APM;
- Uma combinação de variantes selecionadas adaptativamente, SSaDE, e o uso do critério de seleção de Deb.

4.3 A Viga Soldada

Este exemplo é largamente utilizado para a avaliação de algoritmos na literatura, sendo apresentado de forma esquemática na Figura 4.1. A função custo é dada por

$$c(h, b, l, t) = 1.10471h^2l + 0.4811tb(14.0 + l) \quad (4.1)$$

e as restrições se escrevem

$$\begin{aligned}
 g_1(\tau) &= 13600 - \tau \geq 0, & g_4(Pc) &= Pc - 6000 \geq 0 \\
 g_2(\sigma) &= 30000 - \sigma \geq 0, & g_5(\delta) &= 0.25 - \delta \geq 0 \\
 g_3(b, h) &= b - h \geq 0
 \end{aligned} \tag{4.2}$$

onde τ , σ , Pc e δ são dadas por:

$$\begin{aligned}
 \tau &= \sqrt{(\tau')^2 + (\tau'')^2 + l\tau'\tau''/\alpha}, & \delta &= \frac{2.1952}{t^3b}, \tau' = \frac{6000}{\sqrt{2hl}} \\
 \alpha &= \sqrt{0.25(l^2 + (h+t)^2)}, & \sigma &= \frac{504000}{t^2b} \\
 Pc &= 64746.022(1 - 0.0282346t)tb^3, & \tau'' &= \frac{6000(14+0.5l)\alpha}{2(0.707hl(l^2/12+0.25(h+t)^2))}
 \end{aligned} \tag{4.3}$$

As variáveis de projeto $\{h, l, b, t\}$ são limitadas por: $0,125 \leq h \leq 10$ e $0,1 \leq l, b, t \leq 10$.

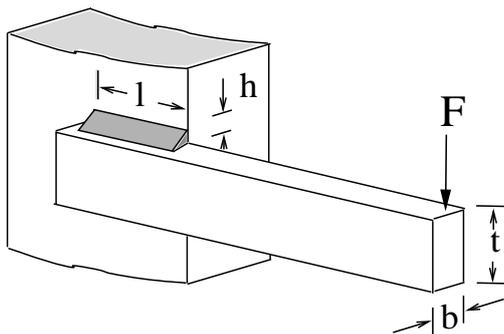


Figura 4.1: Experimento 2 - Viga Soldada. A carga aplicada na extremidade da viga é $F = 6000$ lb

Seguindo os comparativos propostos ao final do experimento anterior, a Tabela 4.11 apresenta os resultados obtidos após a realização, para cada implementação, de 20 execuções independentes que contaram com 500 gerações e 80 indivíduos na população. Para as implementações DUVDE+APM e DUVDE2+APM as configurações foram de 50 indivíduos na população e 421 gerações. A tabela apresenta ainda os resultados obtidos pela aplicação conjunta do APM e de um algoritmo genético (GA+APM), bem como uma técnica híbrida de algoritmo genético e sistema artificial imune, respectivamente apresentados em (Lemonge e Barbosa, 2004)

e (Bernadino, 2008).

Na Tabela 4.11 as implementações avaliadas são identificadas como: “Deb”, implementação do DE com manipulação de restrições proposta por (Deb, 2000). “SSaDE”, simplificação do mecanismo de seleção adaptativa de variantes proposta por (Huang et al., 2006) com uso da manipulação de restrições apresentada por (Deb, 2000). Além das propostas anteriormente apresentadas como “DE+APM”, “DUVDE+APM” e “DUVDE2+APM”. São apresentados também na Tabela 4.11 os resultados obtidos por duas técnicas que utilizam-se de algoritmos genéticos: a primeira é a sua combinação com o APM (GA+APM), (Lemonge e Barbosa, 2004), e a segunda sua hibridização com um sistema imune artificial (GA-AIS), (Bernadino, 2008), sendo considerados os melhores resultados obtidos por essa hibridização para cada problema tratado, utilizando-se de codificação binária ou contínua. Nesses casos foram permitidas respectivamente 40000 e 320000 avaliações da função objetivo. Os demais experimentos também se valeram desses trabalhos para comparação dos resultados obtidos pelas propostas de evolução diferencial implementadas.

A Tabela 4.11 apresenta ainda o número de avaliações da função objetivo (NA) suficientes para a obtenção dos resultados.

Tabela 4.11: Resultado do experimento 2, Viga Soldada

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	2,3815	2,4171	2,9553		40000
GA-AIS	2,3816	2,4359	2,7121	0,0642	320000
Deb	2,3811	2,3811	2,3811	0,0	25000
DE+APM	2,3811	2,3811	2,3811	0,0	25000
SSaDE	2,3811	2,3811	2,3811	0,0	15000
DUVDE+APM	2,3811	2,3811	2,3811	0,0	15000
DUVDE2+APM	2,3811	2,3811	2,3811	0,0	15000

O gráfico da Figura 4.2 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas, nas 12000 primeiras avaliações.

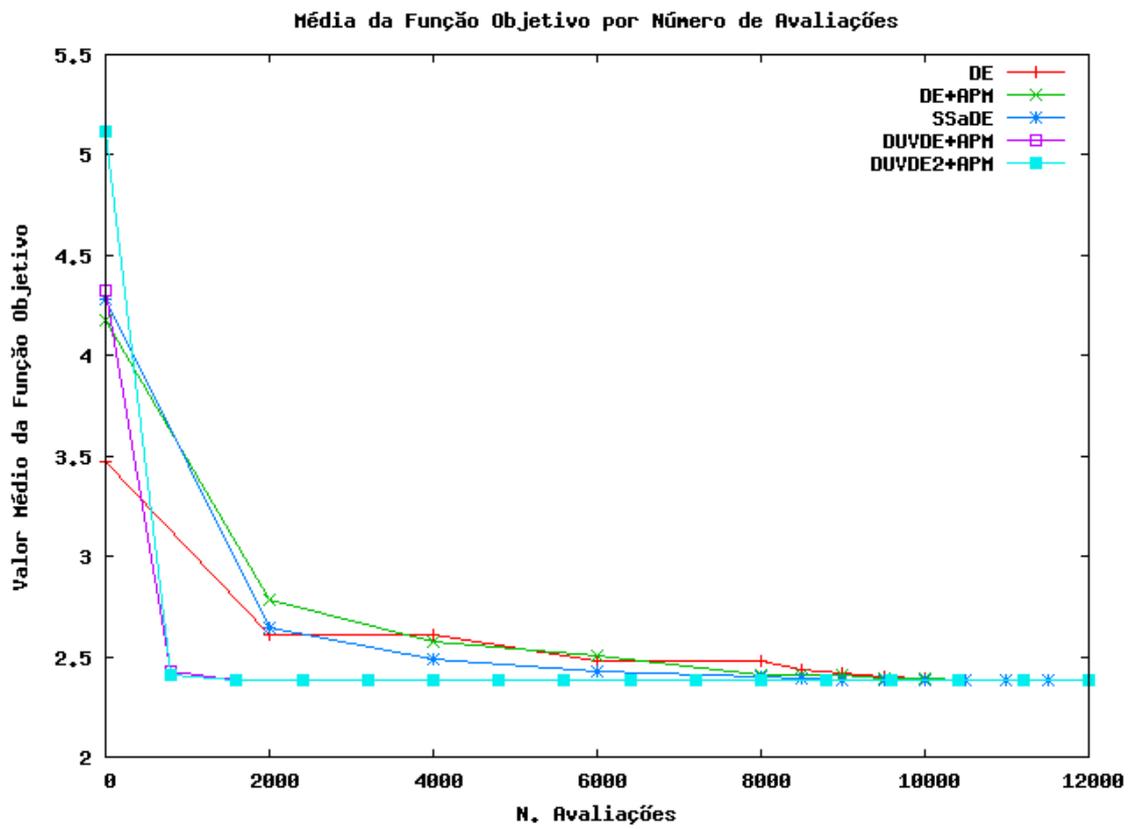


Figura 4.2: Experimento 2 - Viga Soldada. Decaimento médio da função objetivo.

Os resultados da Tabela 4.11 mostram o bom desempenho das técnicas de evolução diferencial aplicadas neste problema. Considerando as soluções encontradas e o número reduzido de avaliações da função objetivo utilizadas.

Além disso, a evolução diferencial mostrou nesse caso uma grande estabilidade, evidenciada pela convergência de todas as suas implementações para um mesmo valor, valor esse menor (e conseqüentemente melhor) do que aqueles obtidos pelas demais técnicas. Outro indicador dessa estabilidade é o desvio padrão igual a zero, mostrando que em todas as execuções, independentemente da população inicial aleatória, a DE convergiu para o valor apresentado.

Pode-se ainda observar a capacidade de rápida convergência da evolução diferencial. No gráfico da Figura 4.2, com apenas 12000 avaliações da função objetivo todas as implementações foram capazes de atingir o valor apresentado na Tabela 4.11. O gráfico mostra ainda uma importante característica das propostas desse trabalho: DUVDE+APM e DUVDE2+APM apresentaram uma curva com grande decaimento em um número reduzido de avaliações. Com isso, e considerando ainda que apesar da rápida convergência o resultado final foi superior ou igual aos demais, pode-se mostrar o melhor desempenho das propostas implementadas aqui, capazes de atingir a solução desejada, com menor custo computacional.

4.4 O Vaso de Pressão

Este experimento da literatura corresponde a um problema de otimização mista, ou seja, possui tanto variáveis de projeto discretas como contínuas. O problema é apresentado de forma esquemática na Figura 4.3 e sua função custo é dada por

$$\omega(Ts, Th, R, L) = 0.6224TsRL + 1.7781ThR^2 + 3.1661Ts^2L + 19.84Ts^2R \quad (4.4)$$

enquanto as restrições se escrevem

$$\begin{aligned}
 g_1(T_s, R) &= T_s - 0.0193R \geq 0 \\
 g_2(T_h, R) &= T_h - 0.00954R \geq 0 \\
 g_3(R, L) &= \pi R^2 L + 4/3\pi R^3 - 1296000 \geq 0
 \end{aligned}
 \tag{4.5}$$

As variáveis são limitadas por: $0,0625 \leq T_s, T_h \leq 5$, em passos constantes de $0,0625$, e $10 \leq R, L \leq 200$.

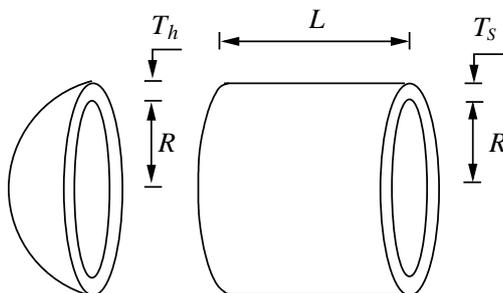


Figura 4.3: Experimento 3 - Vaso de pressão

Para este experimento foram realizadas para cada implementação um total de 20 execuções independentes, sendo a DE configurada com 1000 gerações e 80 indivíduos na população, exceto para as implementações DUVDE+APM e DUVDE2+APM, seguindo o proposto na seção 3.8.2, que utilizaram 842 gerações e 50 indivíduos na população. Para as técnicas GA+APM e GA-AIS foram permitidas, respectivamente, 80000 e 60000 avaliações da função objetivo. Os resultados obtidos são apresentados na Tabela 4.12, com os melhores valores em negrito.

O gráfico da Figura 4.4 apresenta o decaimento médio das aptidões dos indivíduos para cada implementação, em função do número de avaliações realizadas, para as 12000 primeiras avaliações.

Novamente pode-se observar o melhor desempenho da evolução diferencial. Os resultados obtidos pelas implementações da evolução diferencial foram semelhantes, entretanto, com melhor desempenho médio da proposta original e da proposta apresentada neste trabalho denominada DE+APM. O gráfico da Figura 4.4

Tabela 4.12: Resultado do experimento 3, Vaso de Pressão

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	6060, 18	6311, 76	6838, 39		80000
GA-AIS	6059, 85	6545, 12	7388, 16	124	60000
Deb	6059,71	6059,71	6059,71	0, 0	15000
DE+APM	6059,71	6059,71	6059,71	0, 0	15000
SSaDE	6059,71	6083, 39	6410, 09	75, 94	10000
DUVDE+APM	6059,71	6064, 34	6090, 53	11, 00	10000
DUVDE2+APM	6059,71	6095, 87	6410, 09	98, 81	10000

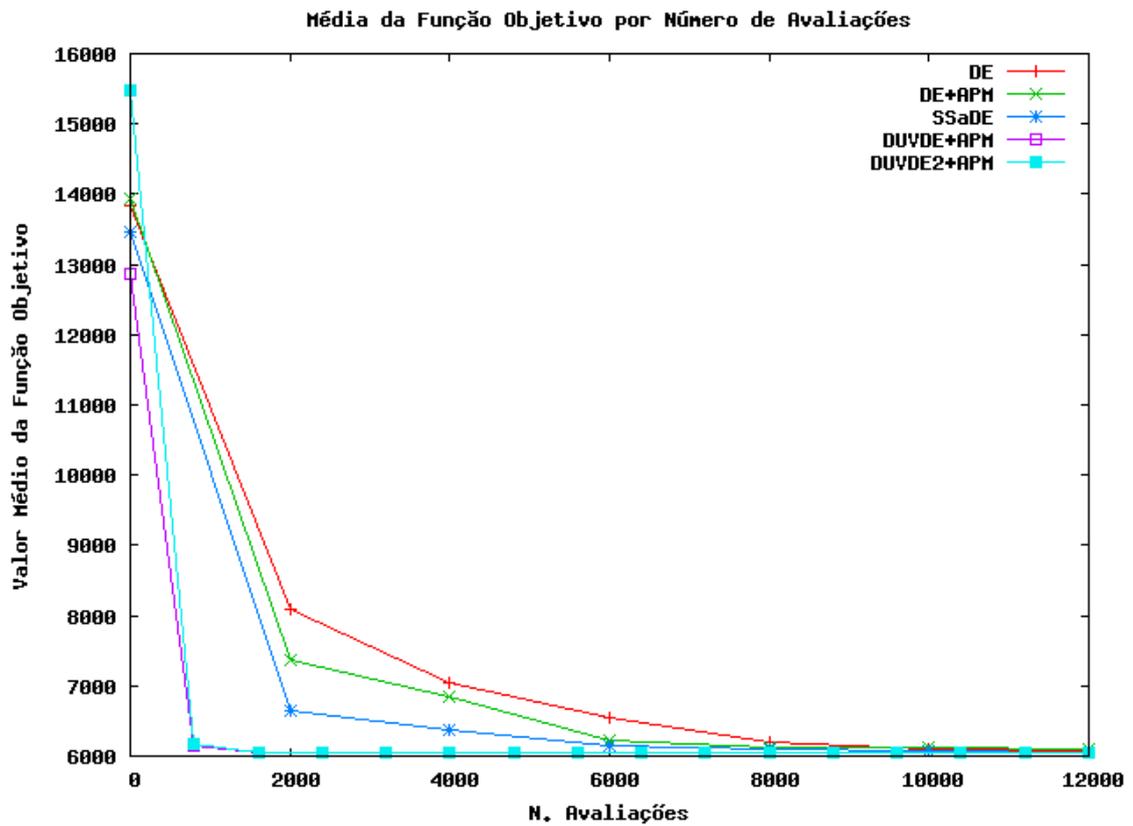


Figura 4.4: Experimento 3 - Vaso de Pressão. Decaimento médio da função objetivo.

mostra mais uma vez a rápida convergência obtida pelas propostas DUVDE+APM e DUVDE2+APM, possibilitando assim uma redução do custo computacional.

4.5 Treliça de Dez Barras

Este é um problema teste bem conhecido correspondente à minimização do peso da treliça de 10 barras apresentada na figura 4.5.

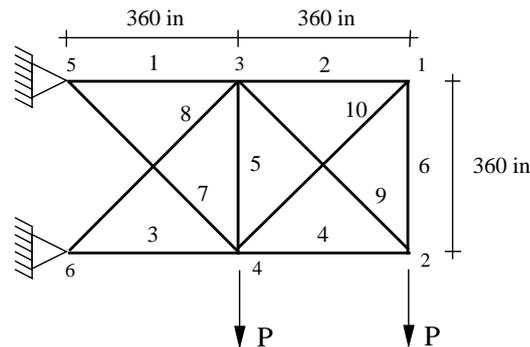


Figura 4.5: Experimento 4 - Treliça de 10 barras

Dois casos foram analisados: variáveis discretas e contínuas. Para o caso discreto os valores das áreas de seção transversal (in^2) são selecionados do conjunto ϑ com 42 opções: 1,62; 1,80; 1,99; 2,13; 2,38; 2,62; 2,63; 2,88; 2,93; 3,09; 3,13; 3,38; 3,47; 3,55; 3,63; 3,84; 3,87; 3,88; 4,18; 4,22; 4,49; 4,59; 4,80; 4,97; 5,12; 5,74; 7,22; 7,97; 11,50; 13,50; 13,90; 14,20; 15,50; 16,00; 16,90; 18,80; 19,90; 22,00; 22,90; 26,50; 30,00; 33,50. Para o caso contínuo a área mínima da seção transversal é igual a $0,1 \text{ in}^2$.

As restrições envolvem a tensão em cada membro e o deslocamento nos pontos nodais. As variáveis de projeto são as áreas de seção transversal das barras $x = \{A_1, A_2, \dots, A_{10}\}$. A tensão é limitada à faixa de $\pm 25 \text{ ksi}$ e os deslocamentos são limitados a 2 in , nas direções x e y . A densidade do material é $0,10 \text{ lb/in}^3$, o módulo de Young é $E = 10^4 \text{ ksi}$, e as cargas verticais de 100 kips são aplicadas nos nós 2 e 4.

4.5.1 Caso Contínuo

Para esse experimento, o número de avaliações da função objetivo permitido é de 280000, o mesmo para GA+APM,(Lemonge e Barbosa, 2004), e GA-AIS (Bernadino, 2008). A DE foi usada com uma população de 100 indivíduos e com 2800 gerações, exceto para as implementações DUVDE+APM e DUVDE2+APM, as quais foram configuradas com 70 indivíduos na população e 2105 gerações.

Vinte execuções independentes foram realizadas e os resultados estão presentes na Tabela 4.13. Na Tabela 4.14 as melhores soluções encontradas para cada implementação são apresentadas.

Tabela 4.13: Resultado do experimento 4, Treliça de 10 Barras - Caso Contínuo

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	5069,09	5091,43	5117,39		280000
GA-AIS	5061,16	5068,85	5084,56	7,78	280000
Deb	5060,854	5060,854	5060,854	0,0	140000
DE+APM	5060,854	5060,854	5060,854	0,0	130000
SSaDE	5060,854	5060,854	5060,854	0,0	120000
DUVDE+APM	5060,853	5060,853	5060,854	0,0002	60000
DUVDE2+APM	5060,853	5060,853	5060,854	0,0002	60000

O gráfico da Figura 4.6 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas, para as 35000 primeiras avaliações.

Bons resultados foram obtidos pela evolução diferencial, sendo as propostas DUVDE+APM e DUVDE2+APM as de melhor desempenho.

4.5.2 Caso Discreto

Para este experimento, o número de avaliações da função objetivo permitido é de 90000. A DE foi configurada com uma população de 200 indivíduos e conseqüentemente 450 gerações enquanto as implementações DUVDE+APM e DUVDE2+APM foram utilizados com 120 indivíduos na população e 394 gera-

Tabela 4.14: Melhores soluções encontradas para o experimento 4 - Caso Contínuo, Treliza de 10 Barras

A_i	GA+ APM	GA-AIS	DE	DE+ APM	SSaDE	DUVDE+ APM	DUVDE2+ APM
1	29, 2256	30, 5268	30, 5179	30, 5162	30, 5236	30, 5250	30, 5224
2	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000
3	24, 1821	22, 9157	23, 2003	23, 2009	23, 1954	23, 1973	23, 1988
4	14, 9471	15, 4829	15, 2278	15, 2257	15, 2248	15, 2229	15, 2265
5	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000
6	0, 3946	0, 5462	0, 5512	0, 5514	0, 5519	0, 5515	0, 5510
7	7, 4957	7, 4759	7, 4572	7, 4574	7, 4574	7, 4573	7, 4572
8	21, 9248	21, 0156	21, 0384	21, 0403	21, 0354	21, 0350	21, 0362
9	21, 2908	21, 5536	21, 5253	21, 5254	21, 5291	21, 5290	21, 5264
10	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000	0, 1000
Peso	5069, 09	5061, 16	5060, 854	5060, 854	5060, 854	5060, 853	5060, 853

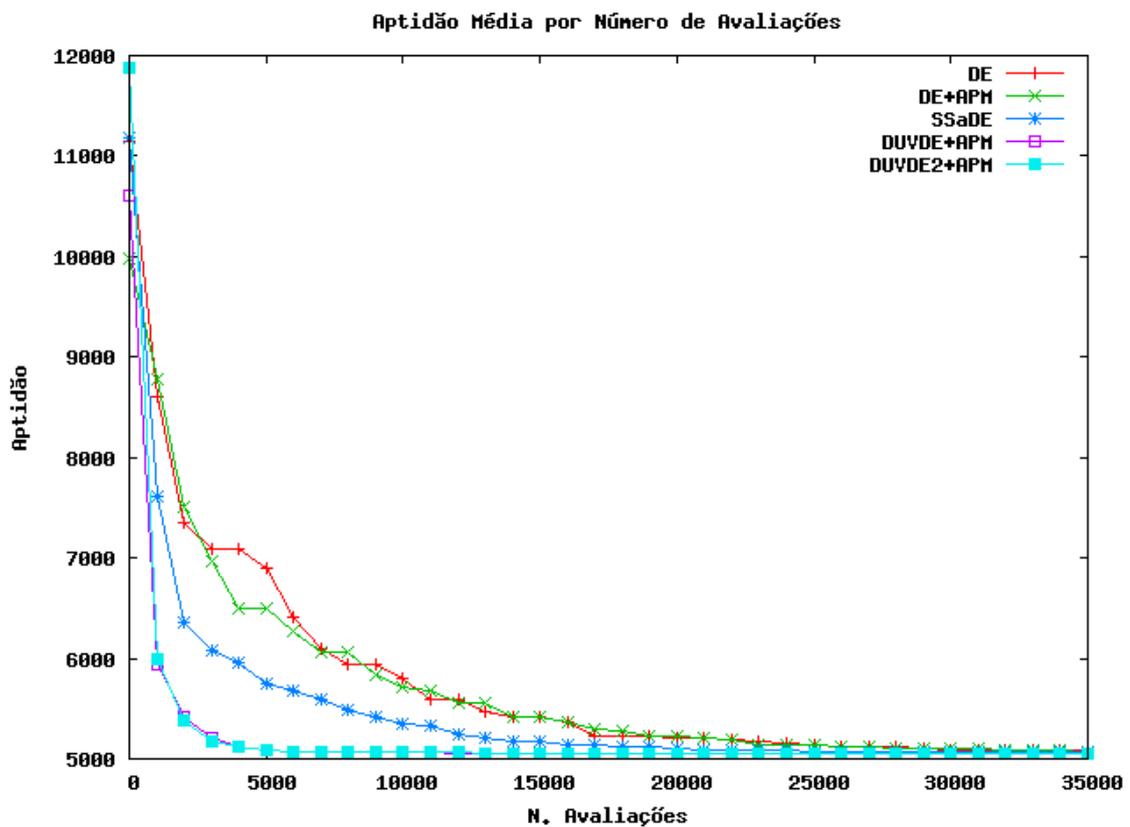


Figura 4.6: Experimento 4 - Treliza de 10 Barras, caso contínuo. Decaimento médio da função objetivo.

ções. Novamente 20 execuções independentes foram realizadas e os resultados são apresentados na Tabela 4.15. As melhores soluções encontradas por cada implementação são apresentadas na Tabela 4.16.

Tabela 4.15: Resultado do experimento 4, Treliça de 10 Barras - Caso Discreto

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	5490,738	5585,980	6443,23		90000
AG-SIA	5490,738	5509,04	5574,54	18,20	90000
Deb	5490,738	5493,583	5499,354	3,233	90000
DE+APM	5490,738	5494,582	5504,058	4,449	90000
SSaDE	5490,738	5490,738	5490,738	0,0	90000
DUVDE+APM	5490,738	5497,988	5537,718	13,717	50000
DUVDE2+APM	5490,738	5496,878	5521,438	9,906	50000

Tabela 4.16: Melhor solução encontrada para o experimento 4, Treliça de 10 Barras - Caso Discreto. Todas as propostas encontraram a mesma solução.

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	Peso
33,50	1,62	22,90	14,20	1,62	1,62	7,97	22,90	22,00	1,62	5490,738

O gráfico da Figura 4.7 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas.

Neste caso, considerando principalmente a média, verifica-se o melhor desempenho da evolução diferencial, que apresentou não só um importante resultado quanto ao melhor valor final produzido, mas também apresentou resultados médios e referentes à pior aptidão, superiores às demais técnicas. Além disso, foi capaz de obter os resultados com um número reduzido de avaliações da função objetivo, como é descrito no gráfico da Figura 4.7. Ressalta-se ainda que os resultados foram obtidos para o caso discreto, apesar da formulação original da DE ser aplicada diretamente apenas a formulações contínuas.

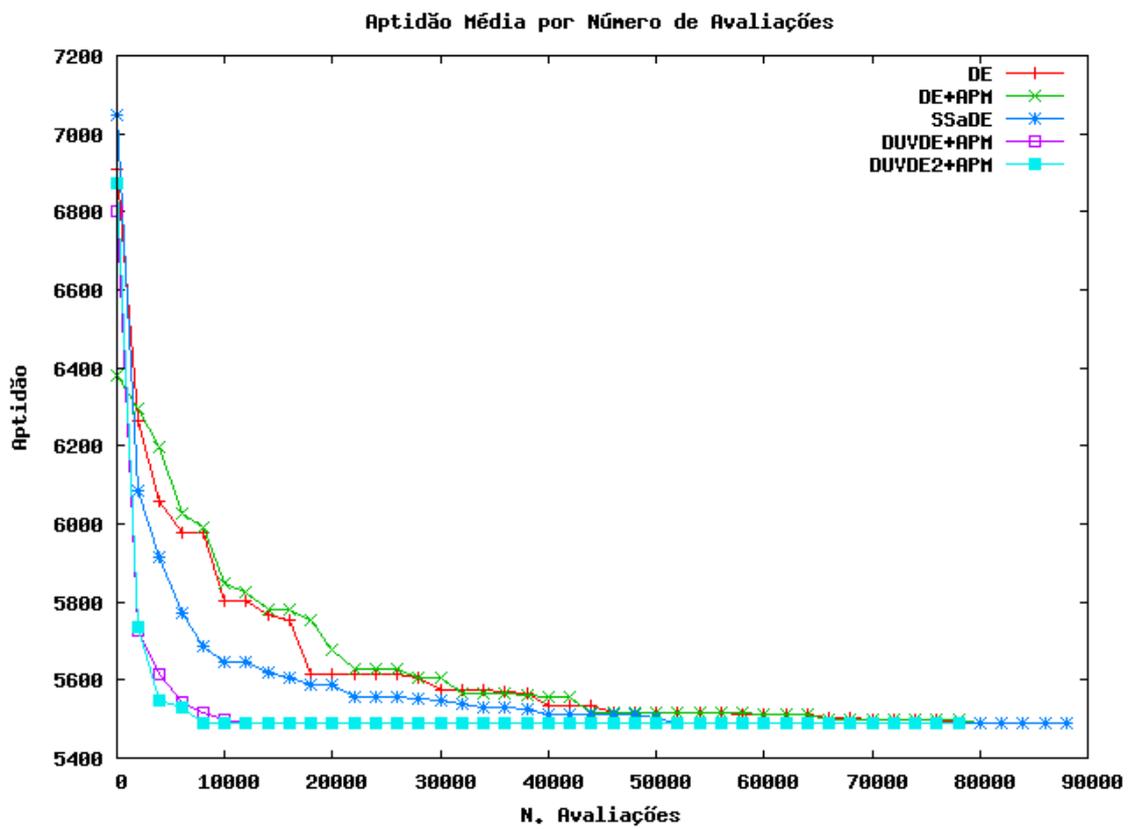


Figura 4.7: Experimento 4 - Treliça de 10 Barras, caso discreto. Decaimento médio da função objetivo.

4.6 Treliça de Vinte e Cinco Barras

Esse é também um bem conhecido problema de minimização de peso estrutural, mostrado na figura 4.8.

O material tem densidade igual a $0,11\text{lb/in}^3$ e módulo de Young igual a 10^4 ksi. Os dados de carregamento são listados na Tabela 4.18. As restrições requerem que a máxima tensão nos membros esteja na faixa de ± 40 ksi e que os deslocamentos máximos nos nós 1 e 2 sejam limitados à $0,35\text{in}$ nas direções x e y . As variáveis de projeto correspondem aos 8 grupos de barras mostrados na Tabela 4.17.

O caso discreto, bem como o contínuo, são considerados. No caso discreto os valores das áreas de seção transversal (in^2) são escolhidos do conjunto $\{0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9; 1,0; 1,1; 1,2; 1,3; 1,4; 1,5; 1,6; 1,7; 1,8; 1,9; 2,0; 2,1; 2,2; 2,3; 2,4; 2,5; 2,6; 2,8; 3,0; 3,2; 3,4\}$. No caso contínuo, as áreas da seção transversal devem pertencer ao intervalo $0,1 \leq A_i \leq 3,4; i = 1, \dots, 8$.

Tabela 4.17: Grupos de barras para a treliça de 25 barras

Grupo	Conectividades
A ₁	1-2
A ₂	1-4, 2-3, 1-5, 2-6
A ₃	2-5, 2-4, 1-3, 1-6
A ₄	3-6, 4-5
A ₅	3-4, 5-6
A ₆	3-10, 6-7, 4-9, 5-8
A ₇	3-8, 4-7, 6-9, 5-10
A ₈	3-7, 4-8, 5-9, 6-10

4.6.1 Caso Contínuo

No caso contínuo, 240000 avaliações da função objetivo foram permitidas, e as implementações da DE foram configuradas com 200 indivíduos e 1200 gerações. Nas implementações DUVDE+APM e DUVDE2+APM foram utilizados 150 indivíduos e 842 gerações, levando as mesmas 240000 avaliações. Os resultados obtidos após

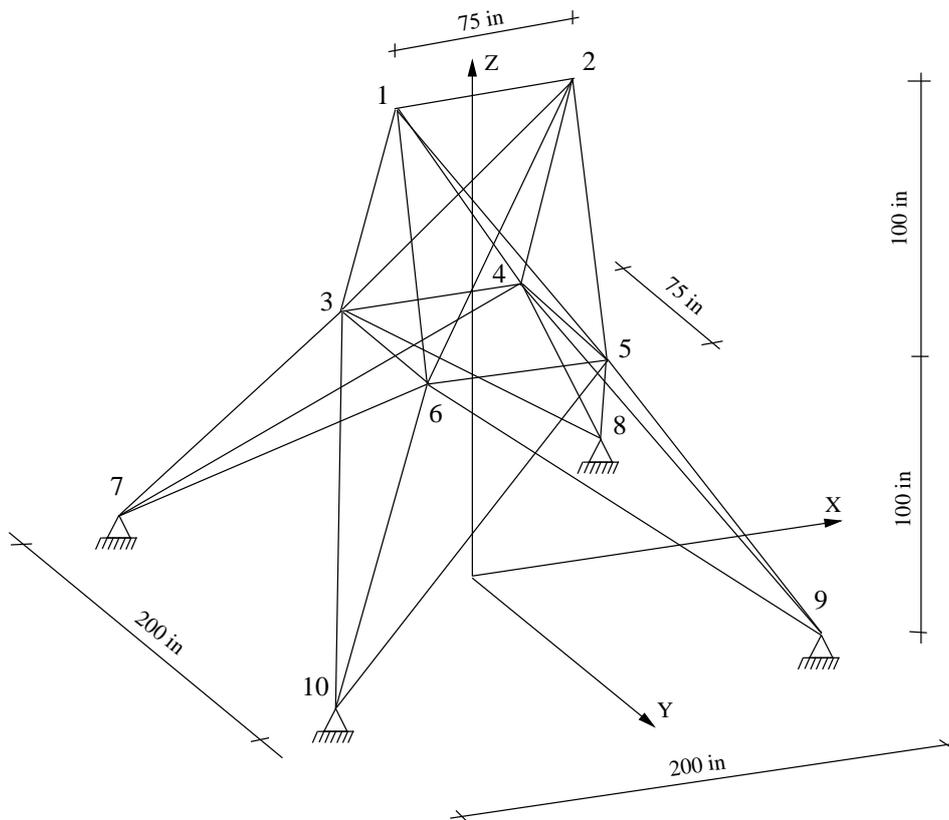


Figura 4.8: Experimento 5 - Trelça de 25 barras

Tabela 4.18: Dados do carregamento para a trelça de 25 barras (kips)

Nó	F_x	F_y	F_z
1	1	-10,0	-10,0
2	0	-10,0	-10,0
3	0,5	0	0
6	0,6	0	0

as 20 execuções independentes são apresentados na Tabela 4.19, e as melhores soluções encontradas por cada implementação são apresentadas na Tabela 4.20.

Tabela 4.19: Resultado do experimento 5, Treliça de 25 Barras - Caso Contínuo

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	484,736	487,665	493,039		240000
Deb	484.051	484.051	484.051	0,0	150000
DE+APM	484.051	484.051	484.051	0,0	150000
SSaDE	484.051	484.051	484.051	0,0	100000
DUVDE+APM	484.051	484.051	484.051	0,0	100000
DUVDE2+APM	484.051	484.051	484.051	0,0	100000

O gráfico da Figura 4.9 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas, para as 80000 primeiras avaliações.

Novamente observa-se os bons resultados obtidos pela evolução diferencial, tendo em vista o resultado final, sua estabilidade e a velocidade com a qual a solução foi encontrada. Destaca-se também a proposta DUVDE2+APM nesse último quesito.

4.6.2 Caso Discreto

No caso discreto, 20000 avaliações da função objetivo foram permitidas, e as implementações da DE foram configuradas com 50 indivíduos e 400 gerações, sendo 20 execuções independentes realizadas. Os resultados são apresentados na Tabela 4.21, e as melhores soluções encontradas são apresentadas na Tabela 4.22. Para as implementações DUVDE+APM e DUVDE2+APM foram utilizados 50 indivíduos na população e 210 gerações.

O gráfico da Figura 4.10 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas, para as 17000 primeiras avaliações.

Novamente observa-se para um problema de formulação discreta, original-

Tabela 4.20: Melhores soluções encontradas para o experimento 5, Treliça de 25 Barras - Caso Contínuo

A_i	GA+ APM	DE	DE+ APM	SSaDE	DUVDE+ APM	DUVDE2+ APM
1	0,10901	0,10000	0,10000	0,10000	0,10000	0,10000
2	0,47443	0,42209	0,42277	0,42235	0,42257	0,42291
3	3,39863	3,40000	3,40000	3,40000	3,40000	3,40000
4	0,10000	0,10000	0,10000	0,10000	0,10000	0,10000
5	1,73260	1,92409	1,92366	1,92357	1,92308	1,92472
6	1,01379	0,96575	0,96575	0,96554	0,96587	0,96589
7	0,43589	0,47163	0,47122	0,47177	0,47138	0,47077
8	3,39859	3,40000	3,40000	3,40000	3,40000	3,40000
Peso	484,736	484,051	484,051	484,051	484,051	484,051

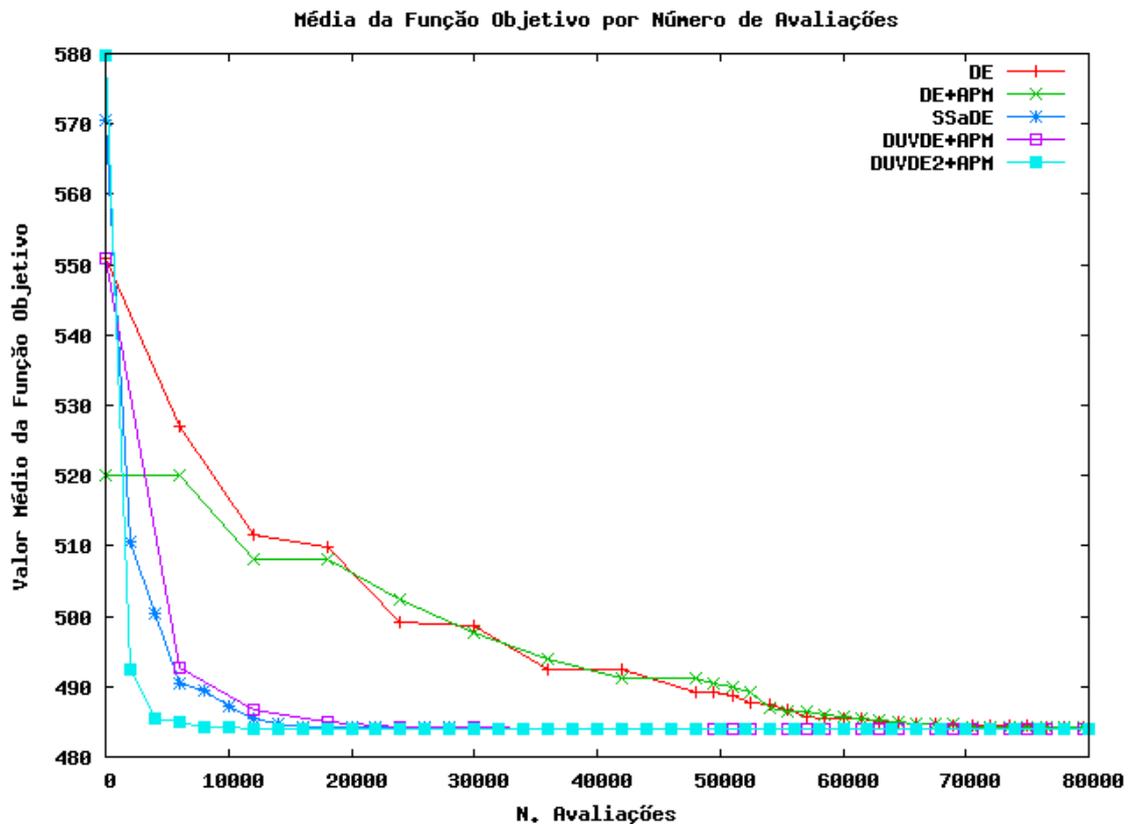


Figura 4.9: Experimento 5 - Treliça de 25 Barras, caso contínuo. Decaimento médio da função objetivo

Tabela 4.21: Resultado do experimento 5, Treliça de 25 Barras - Caso Discreto

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	484,85	485,97	490,74		20000
GA-AIS	484,85	486,20	492,55	1,49	20000
Deb	484,85	484,85	484,85	0,0	15000
DE+APM	484,85	484,88	485,048	0,06	20000
SSaDE	484,85	484,93	485,04	0,09	10000
DUVDE+APM	484,85	485,10	485,76	0,32	8000
DUVDE2+APM	484,85	485,15	487,74	0,63	10000

Tabela 4.22: Melhor solução encontrada para o experimento 5, Treliça de 25 Barras - Caso Discreto. Todas as propostas encontraram a mesma solução.

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	Peso
0,1	0,3	3,4	0,1	2,1	1,0	0,5	3,4	484,85

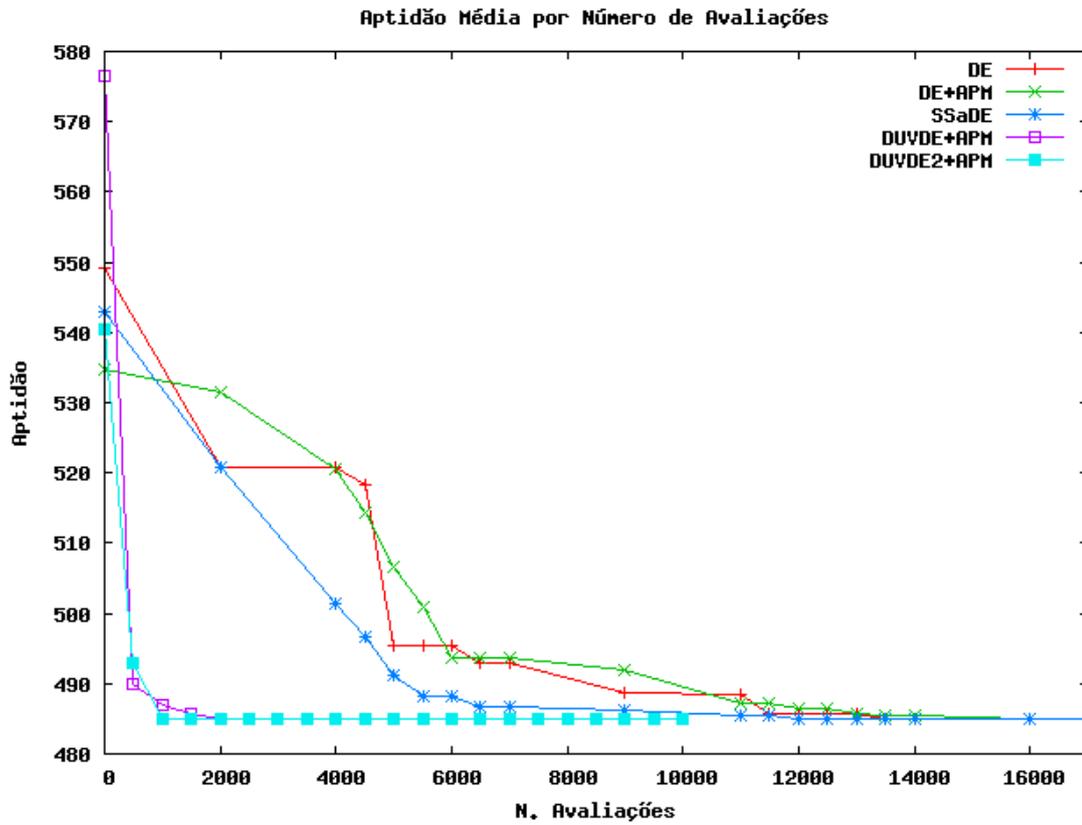


Figura 4.10: Experimento 5 - Treliça de 25 Barras, caso discreto. Decaimento médio da função objetivo

mente fora do domínio de aplicação da evolução diferencial, um melhor desempenho da DE. Isso porque, apesar de encontrar o mesmo resultado, em termos de melhor solução, a técnica apresentou-se superior às demais avaliando-se a média, as piores soluções e até mesmo o número de avaliações da função objetivo, de acordo com o gráfico da Figura 4.10.

4.7 Treliça de Sessenta Barras

Este experimento corresponde à minimização do peso de uma treliça anelar de 60 barras, proposto por Patnaik et al. (Patnaik et al., 1996), esquematizada na Figura 4.11 (fora de escala) sob três condições de carregamento dadas pela Tabela 4.23. O raio externo do anel é de 100in, o raio interno de 90in e o material possui módulo de Young igual a 10^4 ksi e densidade de $0,11\text{lb/in}^3$. Existem 198 restrições, onde 180 referem-se à tensão admissível ($\sigma_i = 60 \text{ ksi}$, $i = 1, \dots, 60$), e 18 referem-se ao deslocamento ao longo das direções x e y , de magnitude: 1,75in para o nó 4, 2,25in para o nó 13 e 2,75 para o nó 19.

As áreas da seção transversal das barras são tomadas como contínuas, com um valor mínimo de $0,5\text{in}^2$, e são unidas em 25 grupos como apresentado na Tabela 4.24.

Tabela 4.23: Casos de carregamento para estrutura em anel de 60 barras (kips)

Caso de Carregamento	Nó	F_x	F_y
1	1	-10,0	0
	7	9,0	0
2	15	-8,0	3,0
	18	-8,0	3,0
3	22	-20,0	10,0

A Tabela 4.25 mostra uma comparação dos resultados. Para as implementações do DE foram utilizados 100 indivíduos na população e 1500 gerações, exceto para as implementações DUVDE+APM e DUVDE2+APM, que foram configuradas com 70 indivíduos e 1339 gerações. O detalhamento das melhores soluções

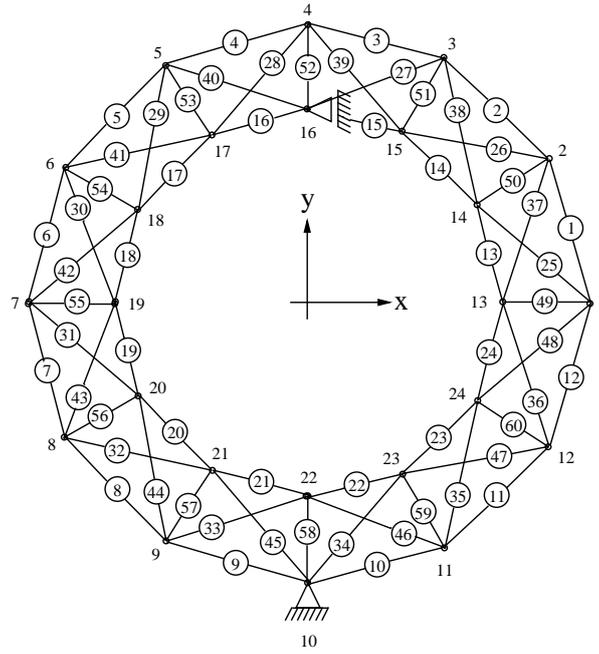


Figura 4.11: Experimento 6 - Treliça anelar de 60 barras

Tabela 4.24: Variáveis de projeto e correspondentes barras para estrutura anelar de 60 barras

Grupo	Barras	Grupo	Barras
A ₁	49 até 60	A ₁₄	25, 37
A ₂	1, 13	A ₁₅	26, 38
A ₃	2, 14	A ₁₆	27, 39
A ₄	3, 15	A ₁₇	28, 40
A ₅	4, 16	A ₁₈	29, 41
A ₆	5, 17	A ₁₉	30, 42
A ₇	6, 18	A ₂₀	31, 41
A ₈	7, 19	A ₂₁	32, 42
A ₉	8, 20	A ₂₂	33, 45
A ₁₀	9, 21	A ₂₃	34, 46
A ₁₁	10, 22	A ₂₄	35, 47
A ₁₂	11, 23	A ₂₅	36, 48
A ₁₃	12, 24		

encontradas por cada implementação é apresentado na Tabela 4.26. As implementações GA+APM, (Barbosa e Lemonge, 2003), e GA-AIS foram avaliadas permitindo a realização de 800000 avaliações da função objetivo.

Tabela 4.25: Resultado do experimento 6, Treliça de 60 Barras

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	311, 88	333, 02	384, 20		800000
GA-AIS	310, 88	313, 55	315, 20	1, 19	800000
Deb	309, 984	310, 807	311, 333	0, 388	150000
DE+APM	310, 011	310, 692	312, 451	0, 620	150000
SSaDE	308, 643	308, 814	309, 282	0, 136	150000
DUVDE+APM	308,613	308, 706	309, 009	0, 103	150000
DUVDE2+APM	308, 619	308,682	308,788	0, 047	150000

O gráfico da Figura 4.12 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas.

Com os resultados deste experimento ressalta-se a qualidade da evolução diferencial e principalmente das implementações DUVDE+APM e DUVDE2+APM, aqui propostas.

4.8 Treliça de Setenta e Duas Barras

Um estrutura de 72 barras, representada na Figura 4.13, é considerada para este experimento. A densidade do material é $0,11\text{lb/in}^3$ e o módulo de Young é igual a 10^4ksi . Dois casos de carregamento, definidos na Tabela 4.28, são considerados para essa estrutura.

As variáveis de projeto são novamente as áreas das seções transversais das barras e seu valor mínimo permitido é de $0,1\text{in}^2$. As 72 variáveis são repartidas em 16 grupos detalhados na Tabela 4.27. As restrições envolvem o deslocamento permitido de $0,25\text{in}$ para os nós 1 a 16 ao longo das direções x e y , e uma máxima tensão em cada barra, restrita à faixa de $\pm 25\text{ksi}$. A DE foi configurada com 70 indivíduos e 500 gerações, enquanto as implementações DUVDE+APM e DUVDE2+APM empregaram 45 indivíduos e 409 gerações. Foram novamente

Tabela 4.26: Melhores soluções encontradas para o experimento 6, Treliça de 60 Barras

A_i	GA+ APM	GA-AIS	DE	DE+ APM	SSaDE	DUVDE+ APM	DUVDE2+ APM
1	1,12023	1,17268	1,16569	1,15785	1,15103	1,14627	1,14711
2	2,02199	2,06219	2,07651	2,10401	2,03461	2,02499	2,02634
3	0,50879	0,51007	0,51292	0,52079	0,50000	0,50004	0,50035
4	1,72727	1,76546	1,75174	1,73472	1,74625	1,79182	1,79797
5	1,52052	1,66584	1,73341	1,70207	1,74977	1,76283	1,76592
6	0,52639	0,57206	0,57607	0,57791	0,58224	0,57580	0,58047
7	1,90322	1,90620	1,85410	1,92730	1,82881	1,82522	1,83638
8	2,12756	1,94845	1,85239	1,86624	1,79093	1,81643	1,81022
9	0,98826	1,05106	0,99358	1,01036	0,98283	0,98505	0,98619
10	2,05278	1,74661	1,77758	1,80944	1,89876	1,89233	1,87775
11	2,05278	1,64572	2,07288	1,77349	1,93229	1,94706	1,94268
12	0,72434	0,51535	0,51875	0,52303	0,50000	0,50015	0,50008
13	1,96041	2,07976	2,03143	2,03855	2,01958	2,01158	2,01419
14	1,23020	1,26165	1,26364	1,25462	1,24355	1,24407	1,24464
15	0,99706	1,13989	1,07431	1,01819	1,03164	1,01590	1,01918
16	0,60557	0,68887	0,69882	0,66129	0,73569	0,66676	0,68587
17	0,72873	0,78881	0,70572	0,73028	0,73297	0,72000	0,72820
18	1,09384	1,04016	1,00009	1,11575	1,04248	1,06180	1,02996
19	1,11583	1,16052	1,12277	1,13181	1,12187	1,12295	1,12329
20	1,16862	1,14993	1,17644	1,15112	1,15174	1,15101	1,15071
21	1,06744	1,00977	1,02770	1,10919	1,06334	1,09139	1,08394
22	1,06304	1,06800	1,05759	1,05288	1,04636	1,04796	1,04787
23	0,58797	0,81993	0,64524	0,72981	0,71892	0,69884	0,70681
24	1,06744	1,06936	1,06927	1,06223	1,03042	1,03610	1,03097
25	1,26979	1,27434	1,27142	1,26831	1,25959	1,25856	1,25836
Peso	311,875	310,880	309,984	310,011	308,643	308,613	308,619

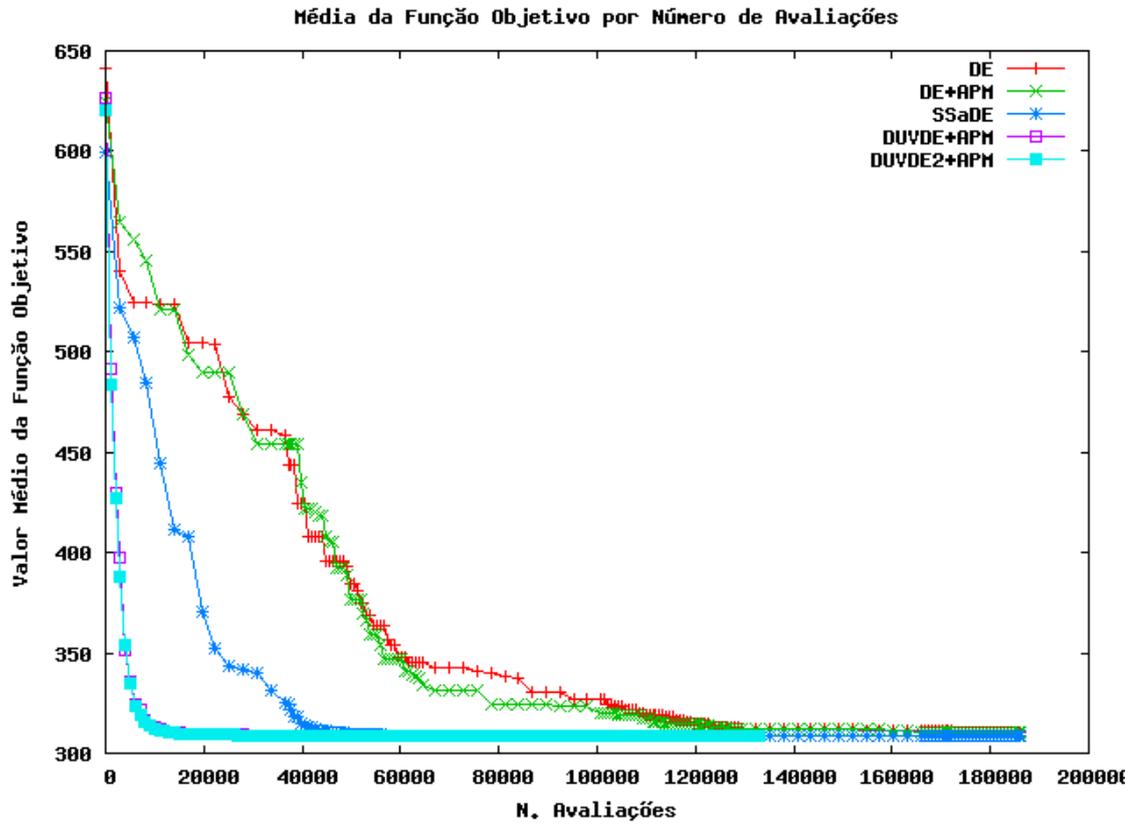


Figura 4.12: Experimento 6 - Treliça de 60 Barras. Decaimento médio da função objetivo.

Tabela 4.27: Grupamentos para a treliça de 72 barras

Grupo	Membros
A ₁	1, 2, 3, 4
A ₂	5, 6, 7, 8, 9, 10, 11, 12
A ₃	13, 14, 15, 16
A ₄	17,18
A ₅	19, 20, 21, 22
A ₆	23, 24, 25, 26, 27, 28, 29, 30
A ₇	31, 32, 33, 34
A ₈	35,36
A ₉	37, 38, 39, 40
A ₁₀	41, 42, 43, 44, 45, 46, 47, 48
A ₁₁	49, 50, 51, 52
A ₁₂	53,54
A ₁₃	55, 56, 57, 58
A ₁₄	59, 60, 61, 62, 63, 64, 65, 66
A ₁₅	67, 68, 69, 70
A ₁₆	71,72

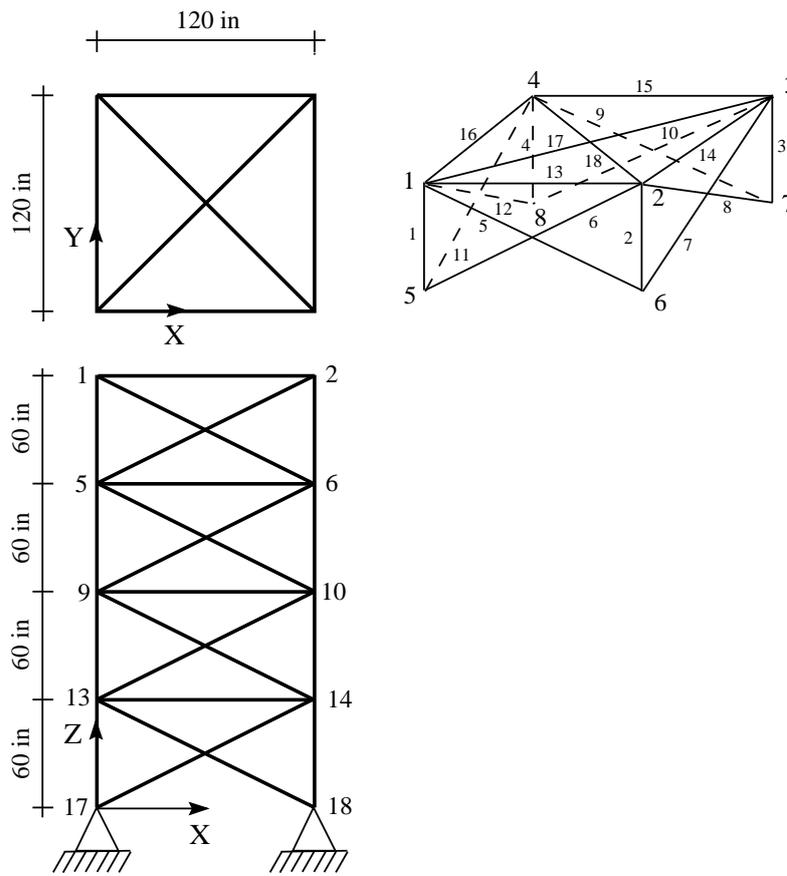


Figura 4.13: Experimento 7 - Treliza 72 barras

Tabela 4.28: Dados do carregamento para estrutura 72 barras (kips)

Caso de Carregamento	Nó	F_x	F_y	F_z
1	1	5	5	-5
2	1	0	0	-5
	2	0	0	-5
	3	0	0	-5
	4	0	0	-5

realizadas 20 execuções independentes para cada implementação e as melhores soluções obtidas para cada uma delas são apresentadas na Tabela 4.30. Todas as implementações avaliadas permitiram 35000 avaliações da função objetivo.

A Tabela 4.29 mostra uma comparação dos resultados obtidos. O gráfico da

Tabela 4.29: Resultado do experimento 7, Treliça de 72 Barras

Proposta	Melhor	Média	Pior	Desvio Padrão	NA
GA+APM	387,04	402,59	432,95	-	35000
GA-AIS	381,29	385,16	391,49	2,61	35000
Deb	383,756	386,384	389,499	1,378	35000
DE+APM	384,311	386,596	389,563	1,407	35000
SSaDE	379,642	379,755	379,974	0,093	35000
DUVDE+APM	379,646	379,822	380,565	0,205	35000
DUVDE2+APM	379,639	379,812	380,040	0,140	35000

Figura 4.14 apresenta o decaimento médio das aptidões dos indivíduos, para cada implementação, em função do número de avaliações realizadas.

Neste último experimento destacam-se os resultados obtidos por implementações da evolução diferencial que fazem a seleção da variante durante o processo evolutivo (SSaDE, DUVDE+APM e DUVDE2+APM). E, novamente, observa-se no gráfico da Figura 4.14 uma rápida convergência da proposta DUVDE+APM.

Tabela 4.30: Melhores soluções encontradas para o experimento 7, Treliça de 72 Barras

A_i	GA+ APM	GA-AIS	DE	DE+ APM	SSaDE	DUVDE+ APM	DUVDE2+ APM
1	0,15500	0,15745	0,15793	0,15614	0,15663	0,15635	0,15636
2	0,54534	0,54583	0,53106	0,50627	0,54272	0,54957	0,54687
3	0,27496	0,35137	0,36328	0,40823	0,41471	0,40420	0,41695
4	0,51853	0,58946	0,56432	0,68303	0,57669	0,56815	0,57505
5	0,60365	0,54317	0,57547	0,57854	0,51555	0,52273	0,52133
6	0,66607	0,53262	0,51302	0,50724	0,51233	0,51773	0,51171
7	0,10159	0,10192	0,11145	0,11629	0,10000	0,10026	0,10001
8	0,13008	0,10358	0,12756	0,13827	0,10037	0,10018	0,10035
9	1,19954	1,32186	1,32888	1,26333	1,27830	1,26833	1,26499
10	0,47368	0,51468	0,52091	0,49949	0,51606	0,51097	0,51036
11	0,10059	0,10250	0,11042	0,11793	0,10000	0,10005	0,10009
12	0,10945	0,11831	0,10381	0,10914	0,10001	0,10022	0,10003
13	1,95307	1,92280	1,86695	1,89375	1,87087	1,87777	1,89388
14	0,51653	0,49524	0,52568	0,53508	0,51453	0,51380	0,51273
15	0,10000	0,10014	0,10465	0,10764	0,10000	0,10002	0,10005
16	0,10105	0,10120	0,14565	0,10321	0,10000	0,10001	0,10000
Peso	387,036	381,293	383,756	384,311	379,642	379,646	379,639

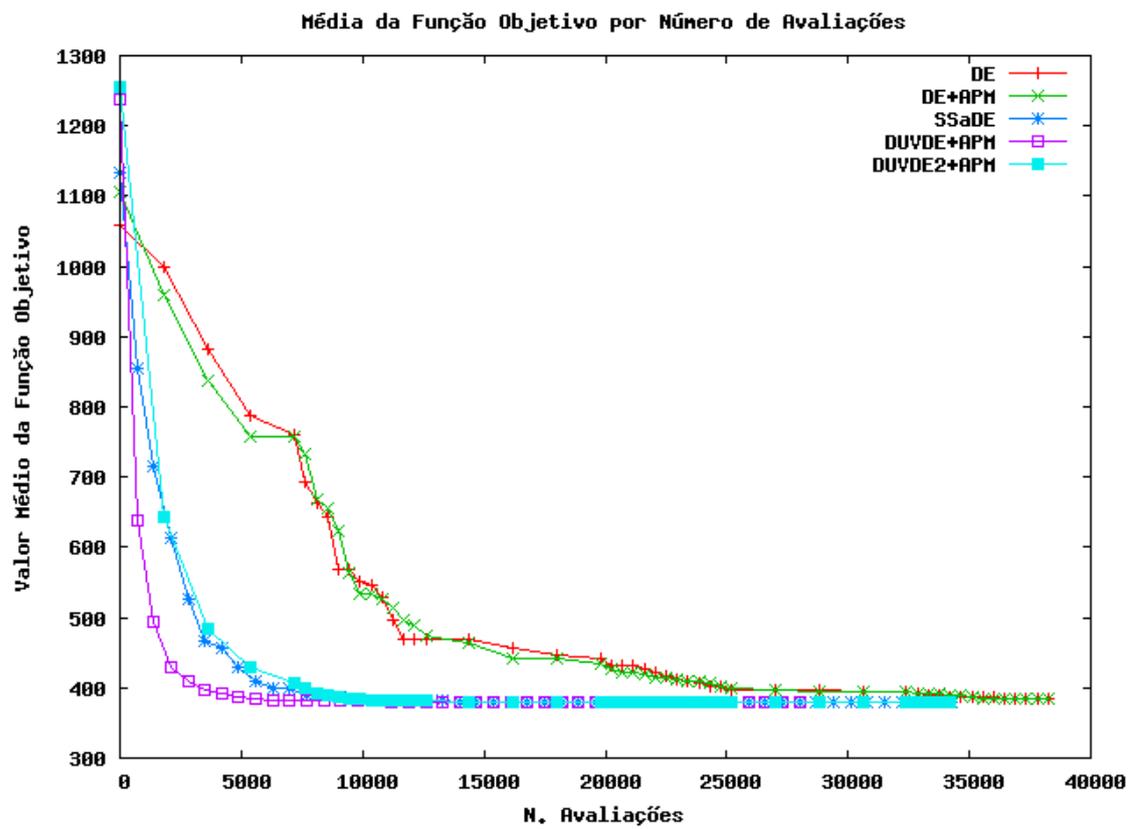


Figura 4.14: Experimento 7 - Treliça de 72 Barras. Decaimento médio da função objetivo

Capítulo 5

Conclusão

A evolução diferencial é uma técnica relativamente nova, aplicada originalmente para otimização irrestrita e sobre espaços contínuos. Este estudo foi capaz de avaliar a qualidade da técnica para um domínio de aplicação mais amplo, principalmente na otimização restrita.

Este trabalho propôs não só a avaliação da DE para a otimização restrita, mas também sua avaliação em problemas com variáveis de projeto discretas e apresentou três novas propostas para o método. As propostas apresentadas visaram o aperfeiçoamento da DE para a sua aplicação em problemas de otimização restrita, bem como a seleção da variante adequada do método para o problema tratado.

As propostas apresentadas são caracterizadas principalmente por não requerer parâmetros adicionais para sua aplicação, sendo que as configurações são adaptadas automaticamente durante o processo evolutivo.

Com a avaliação dos experimentos realizados pode-se concluir que a DE obteve resultados relevantes, com destaque para a otimização de estruturas treliçadas, onde não só os resultados finais foram bons, mas também por uma significativa redução do custo computacional.

Para os problemas de variáveis de projeto contínuas a DE mostrou-se superior aos demais métodos, sendo capaz de obter melhores soluções com menor custo computacional.

Para os casos de variáveis de projeto discretas a DE foi capaz de obter bons

resultados, considerando que sua aplicação original não objetivava essa formulação e, apesar disso, os resultados obtidos foram melhores ou iguais aos dos demais métodos avaliados. Observa-se novamente a redução do número de avaliações da função objetivo necessárias para a produção desses resultados.

Para ambos os casos, comparando as diversas propostas para a DE aqui avaliadas, observa-se o ganho resultante da aplicação conjunta de um mecanismo de penalização adaptativa e uma seleção dinâmica de variantes, identificadas por DUVDE+APM e DUVDE2+APM. Estas propostas mostraram de convergência rápida sem perda de qualidade.

Observa-se ainda que a DE mostrou-se muito estável, ou seja, em todos os casos obteve-se pequenas variações dos resultados em cada execução, independente da população inicial aleatória utilizada.

Por fim, os resultados obtidos por este trabalho, principalmente pelas propostas DUVDE+APM e DUVDE2+APM, motivam a continuação de pesquisas quanto à aplicação e aperfeiçoamento da evolução diferencial, bem como sua utilização em outros problemas de interesse.

Referências Bibliográficas

- H. J. C. Barbosa e A. C. C. Lemonge. A new adaptive penalty scheme for genetic algorithms. **Information Sciences**, 156(3-4):215–251, 2003.
- H. J. C. Barbosa e A. C. C. Lemonge. A genetic algorithm encoding for a class of cardinality constraints. **GECCO '05: Proc. of the 2005 Conference on Genetic and Evolutionary Computation**, páginas 1193–1200, 2005.
- H.J.C. Barbosa e A.C.C. Lemonge. **Frontiers in Evolutionary Robotics**. Capítulo An Adaptive Penalty Method for Genetic Algorithms in Constrained Optimization Problems, I-Tech Education and Publishing, 2008.
- J. C. Bean e A. B. Alouane. A dual genetic algorithm for bounded integer programs. Relatório Técnico **TR 92-53**, Department of Industrial and Operator Engineering, The University of Michigan, 1992.
- H. S. Bernadino. Hibridização de algoritmos genéticos e sistemas imunológicos artificiais para problemas de otimização com restrições em engenharia. Dissertação de Mestrado, Universidade Federal de Juiz de Fora - MG - Brasil, 2008.
- J. Brest, S. Greiner, B. Boskovic, M. Mernik, e V. Zumer. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. **Evolutionary Computation, IEEE Transactions on**, 10(6):646–657, Dec. 2006.
- C. G. Broyden. The convergence of a class of double-rank minimization algorithms. **Journal of the Institute of Mathematics and Its Applications**, 6:76–90, 1970.

- T. Bäck. **Evolutionary Algorithms in Theory and Practice**. Oxford University Press, 1996.
- V. Cerný. A thermodynamical approach to the traveling salesman problem. **Journal of Optimization Theory and Applications**, 45:41–51, 1985.
- U. K. Chakraborty. **Advances in Differential Evolution**. Springer-Verlag, Inc., 2008.
- D. W. Coit, A. E. Smith, e D. M. Tate. Adaptive penalty methods for genetic optimization of constrained combinatorial problems. **INFORMS Journal on Computing**, 6(2):173–182, 1996.
- G. B. Dantzig. **Linear Programming and Extensions**. Princeton University Press, 1963.
- K. Deb. An efficient constraint handling method for genetic algorithms. **Computer Methods in Applied Mechanics and Engineering**, (186):311–338, 2000.
- A. B. H. Ferreira. **Novo Aurélio Século XXI: o dicionário da língua portuguesa**. Editora Nova Fronteira S.A., 3 edição, 1999.
- R. Fletcher. A new approach to variable metric algorithms. **Computer Journal**, 13:317–322, 1970.
- A. Friedlander. **Elementos de Programação Não-Linear**. Unicamp, 1994.
- J. M. Gere e W. Weaver. **Análise de Estruturas Reticuladas**. Editora Guanabara S.A., 1987.
- D. Goldfarb. A family of variable metric updates derived by variational means. **Mathematics of Computation**, 24:23–26, 1970.
- J. R. Harland e P. Salamon. Simulated annealing: A review of the thermodynamic approach. **Nuclear Physics B Proceedings Supplements**, 5:109–115, 1988.

- M. R. Hestenes e E. Stiefek. Methods of conjugate gradients for solving linear systems. **Journal of Research of the National Bureau of Standards**, 49 (6), 1952.
- R. Hooke e T. A. Jeeves. A simplex method for function minimization. **J. Ass. Comput. Mach.**, 8:212–241, 1961.
- H. H. Hoos e T. Stützle. **Stochastic Local Search: Foundations and Applications**. Morgan Kaufmann Publishers, 2005.
- V.L. Huang, A.K. Qin, e P.N Suganthan. Self-adaptive differential evolution algorithm for constrained real-parameter optimization. In: **IEEE Congress on Evolutionary Computation**, 2006.
- K. A. D. Jong. **Analysis of Behavior of a Class of Genetic Adaptive System**. Tese de Doutorado, University of Michigan, 1975.
- H. Kim, J. Chong, K. Park, e D. A. Lowther. Differential evolution strategy for constrained global optimization and application to practical engineering problems. **IEEE Transactions on Magnetics**, 43(4):1565–1568, 2007.
- S. Kirkpatrick, C. D. Gelatt, e M. P. Vecchi. Optimization by simulated annealing. **Science**, 220:671–680, 1983.
- S. Koziel e Z. Michalewicz. Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization. **Evolutionary Computation**, 7(1): 19–44, 1999.
- L. Lamberti. An efficient simulated annealing algorithm for design optimization of truss structures. **Computers and Structures**, 86(19-20):1936–1953, 2008.
- J. Lampinen e I. Zelinka. Mixed integer-discrete-continuous optimization by differential evolution, part 1: The optimization method. **Proceedings of the 5th Intl. Mendel Conf. on Soft Computing**, páginas 71–76, 1999.

- A. C. C. Lemonge. **Aplicação de Algoritmos Genéticos em Otimização Estrutural**. Tese de Doutorado, Programa de Engenharia Civil?COPPE - Universidade Federal do Rio de Janeiro / Brasil, 1999.
- A. C. C. Lemonge e H. J. C. Barbosa. An adaptive penalty scheme for genetic algorithms in structural optimization. **Intl. J. For Numerical Methods in Engineering**, 59(5):703–736, 2004.
- D. G. Luenberger. **Linear and Nonlinear Programing**. Wesley Publishing Company, 2 edição, 1986.
- E. Mezura-Montes, J. Velázquez-Reyes, e C. A. C. Coelho. A comparative study of differential evolution variants for global optimization. **GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation**, páginas 485–492, 2006a.
- E. Mezura-Montes, J. Velázquez-Reyes, e C.A.C. Coelho. A comparative study of differential evolution variants for global optimization. **Proceedings of the GECCO'06**, páginas 8–12, 2006b.
- E. Mezura-Montes, J. Velázquez-Reyes, e C.A.C. Coelho. Modified differential evolution for constrained optimization. **Proceedings of the IEEE Congress on Evolutionary Computation**, páginas 25–32, 2006c.
- M. Mitchell. **An Introduction to Genetic Algorithms**. MIT Press, 1996.
- J. A. Nelder e R. Mead. A simplex method for function minimization. **The Computer Journal**, 7:308–131, 1965.
- G. C. Onwubolu. Differential evolution for the flow shop scheduling problem. páginas 585–611, 2004.
- S. Patnaik, D. Hopkins, e R. Coroneos. Structural optimization with approximate sensitivities. **Computer & Structures**, 58(2):407–418, 1996.
- K. V. Price. Genetic annealing. **j-DDJ**, 19(11):127, 1994.

- K. V. Price. An introduction to differential evolution. **New Ideas in Optimization**, páginas 79–108, 1999.
- K. V. Price, R. M. Storn, e J. A. Lampinen. **Differential Evolution: A Pratical Approach to Global Optimization**. Springer-Verlag Berlin Heidelberg, 2005.
- A. K. Qin e P. N. Suganthan. Self-adaptive differential evolution algorithm for numerical optimization. **Evolutionary Computation, 2005. The 2005 IEEE Congress on**, 2, 2005.
- A. Qing. A study on base vector for differential evolution. **Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2008)**, 2008.
- T. P. Runarsson e X. Yao. Stochastic ranking for constrained evolutionary optimization. **Evolutionary Computation, IEEE Transactions on**, 4(3):284–294, 2000.
- M. Schoenauer e S. Xanthakis. Constrained ga optimization. **Proc. of the Fifth Int. Conf. on Genetic Algorithms**, páginas 573–580, 1993.
- D. F. Shanno. Conditioning of quasi-newton methods for function minimization. **Mathematics of Computation**, 24:647–656, 1970.
- E. K. Silva e H. J. C. Barbosa. Um algoritmo genético para corte guilhotinado retangular em duas dimensões. In: **VIII Simpósio Mecânica Computacional - SIMMEC**, página 85, Belo Horizonte, MG, 2008a.
- E. K. Silva e H. J. C. Barbosa. Um estudo sobre os avanços da evolução diferencial para otimização restrita mista. In: **Anais do XXIX CILAMCE**, página 605, Maceio, AL, 2008b.
- E. K. Silva, H.J.C. Barbosa, e A.C.C. Lemonge. An adaptive constraint handling technique for differential evolution in engineering optimization. **International Conference on Engineering Optimization - EngOpt 2008**, página 46, 2008.

- O. S. Soliman e L. T. Bui. A self-adaptive strategy for controlling parameters in differential evolution. **Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2008)**, 2008.
- R. Storn e K. V. Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Relatório Técnico **TR-95-012**, ICSI, 1995.
- R. Storn e K. V. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. **Journal of Global Optimization**, 11:341–359, 1997.
- C. Tsallis e D. A. Stariolo. Generalized simulated annealing. **Physica A: Statistical and Theoretical Physics**, 233(1-2):395–406, 1996.
- M. Zhang, W. Luo, e X. Wang. Differential evolution with dynamic stochastic selection for constrained optimization. **Information Sciences, the special issue on Nature Inspired Problem Solving: Theories and Applications**, (178):3043–3074, 2008.
- K. Zielinski e R. Laur. Constrained single-objective optimization using differential evolution. In: **2006 IEEE Congress on Evolutionary Computation**, 2006.

Apêndice A

Otimização Estrutural

Neste apêndice é feita uma sucinta apresentação da otimização estrutural. É evidente que a otimização é fundamental em diversas áreas do conhecimento, e que a otimização estrutural mostra-se de grande importância na engenharia.

A busca pela melhoria de um projeto, bem como a redução de seus custos, correspondente à sua otimização, sendo esta modelada como uma função objetivo a ser minimizada ou maximizada. Da mesma forma, as necessidades e limitações vinculadas ao projeto podem ser representadas por restrições às quais a solução está sujeita.

Em projetos mecânicos e estruturais, são fundamentais as definições da topologia, dimensões das estruturas, materiais utilizados e demais componentes do projeto, assim como a configuração estrutural do mesmo. Além disso, as definições são limitadas muitas vezes por padrões existentes, sendo necessário a seleção do componente desejado dentre os disponíveis para aquisição. Claramente o projeto é também limitado pelas exigências geradas por sua funcionalidade, por exemplo, as solicitações aplicadas sobre a estrutura.

Na elaboração de um projeto mecânico ou estrutural existem diversos objetivos que podem ser considerados como, por exemplo, o custo final, a durabilidade, a confiabilidade, a eficiência, entre outros. Um exemplo é o projeto de treliças metálicas, como torres de transmissão, nas quais uma pequena redução do custo pode acarretar em uma grande economia devido à necessidade de um grande número de

unidades.

Pode-se imaginar também a otimização da forma da asa de uma aeronave, otimização da seção transversal de uma viga de uma ponte, otimização da espessura e geometria de um vaso de pressão, otimização de estruturas tridimensionais de edificações, localização ótima da posição de pilares numa edificação e maximização das frequências de vibração da estrutura de uma ponte.

De forma geral, a otimização estrutural leva a um problema de otimização restrita, dado que na maioria dos casos as variáveis de projeto e medidas de desempenho da estrutura estão sujeitas a diversas limitações.

A.1 Exemplo de otimização estrutural

Afim de exemplificar um problema estrutural, considera-se aqui a otimização de uma treliça metálica, principal foco dos experimentos apresentados no trabalho, esquematizada na figura A.1, formada por três barras de eixos retos.

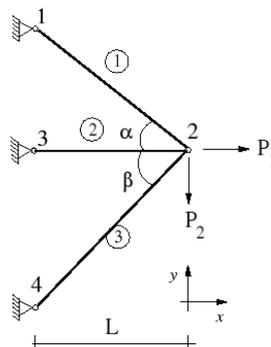


Figura A.1: Treliça de três barras

Nas extremidades de cada barra são definidos os nós 1, 2, 3 e 4. Sobre o nó 2 é aplicado um carregamento externo constituído de duas cargas concentradas de intensidade P_1 e P_2 nas direções dos eixos x e y , respectivamente.

O objetivo do projetista é a minimização do volume da treliça, consequentemente reduzindo o custo do material empregado, satisfazendo as condições de segurança com o máximo de economia.

Designando-se por $A_i, i = 1, \dots, 3$, as áreas das seções transversais das bar-

ras, definidas como variáveis de projeto, o volume $V(A_i)$ a ser minimizado é escrito como

$$V(A_i) = A_1L_1 + A_2L_2 + A_3L_3 = \sum_{i=1}^3 A_iL_i \quad (\text{A.1})$$

onde $L_i, i = 1, 2, 3$, são os comprimentos das barras.

Restrições devem ser consideradas para o problema, afim de evitar soluções inaceitáveis, considerando, por exemplo, tensões internas nos elementos da estrutura, deslocamentos, frequências de vibração, cargas críticas de flambagem, entre outras. Também é comum as restrições não serem escritas explicitamente como funções das variáveis de projeto envolvidas na função objetivo e isso pode acarretar dificuldades na avaliação de uma solução candidata quanto à sua factibilidade.

A.1.1 Determinação das restrições

A aplicação das cargas P_1 e P_2 sobre o nó 2 da estrutura causará deslocamentos nas direções dos eixos x e y , sendo uma possível configuração geométrica da treliça após a aplicação do carregamento dada pela figura A.2. L^* representa o

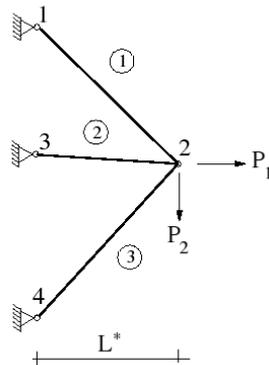


Figura A.2: Possível configuração da treliça de três barras após aplicação do carregamento

valor modificado da dimensão L da treliça apresentada na figura A.1, além disso as barras 1, 2 e 3 sofrerão deformações e estarão submetidas a tensões internas produzidas por esforços normais internos, as quais possuem direções normais às seções transversais das barras. O valor dessa tensão normal em cada barra é a

razão entre o esforço normal e a área da seção transversal da barra. Tal valor pode ser obtido após a determinação dos deslocamentos ocorridos no nó 2 da estrutura.

Definindo-se $\sigma_i, i = 1, 2, 3$ como sendo as tensões normais nas barras e u_{2x} e u_{2y} como os deslocamentos do nó 2, nas direções x e y , respectivamente, escreve-se:

$$\begin{aligned}\sigma_i &\leq \bar{\sigma}_i & i = 1, 2, 3 \\ u_{2x}, u_{2y} &\leq \bar{u}_2\end{aligned}$$

onde $\bar{\sigma}_i$ e \bar{u}_2 são os limites máximos para as tensões normais do material nas barras e para os deslocamentos do nó 2, respectivamente.

Com isso, pode-se formular o problema de otimização, como

$$\begin{aligned}\text{minimizar } V(A_i) &= \sum_{i=1}^3 A_i L_i \\ \text{sujeito a} & \\ \sigma_i &\leq \bar{\sigma}_i \quad i = 1, 2, 3 \\ u_{2x}, u_{2y} &\leq \bar{u}_2\end{aligned}\tag{A.2}$$

Existem ainda, normalmente, restrições adicionais inseridas no problema afim de reduzir o espaço de busca: as variáveis de projeto são limitadas inferior e superiormente, como por exemplo: $A_i^{\min} \leq A_i \leq A_i^{\max} \quad i = 1, 2, 3$.

A.1.2 Determinação dos deslocamentos e tensões

Determinados os valores das áreas A_i torna-se possível encontrar os deslocamentos e tensões através da resolução de um sistema de equações lineares, o qual envolve as características elásticas e geométricas da estrutura, assim como o carregamento aplicado. O sistema representa a equação de equilíbrio da estrutura, em termos dos deslocamentos dada por

$$[K]\{u\} = \{F\}\tag{A.3}$$

onde $[K]$ é uma matriz quadrada de dimensões $n \times n$ chamada de matriz de rigidez da estrutura. $\{F\}$ é o vetor de cargas com componentes calculadas em função do carregamento aplicado. O sistema de equações para o exemplo utilizado possui dimensão 2×2 e pode ser escrito como:

$$\begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{Bmatrix} u_{2x} \\ u_{2y} \end{Bmatrix} = \begin{Bmatrix} F_{2x} \\ F_{2y} \end{Bmatrix}$$

onde k_{ij} são os coeficientes de rigidez, que dependem de características elásticas e geométricas das barras. As componentes F_{2x} e F_{2y} são os valores dos carregamentos P_1 e P_2 aplicados diretamente sobre o nó 2 nas direções x e y , respectivamente. Assim como u_{2x} e u_{2y} são os valores dos deslocamentos a serem determinados.

Para o exemplo tratado, os coeficientes de rigidez podem ser calculados por:

$$\begin{aligned} k_{11} &= \frac{E_1 A_1}{L_1} \cos \alpha + \frac{E_2 A_2}{L_2} + \frac{E_3 A_3}{L_3} \cos \beta \\ k_{12} &= k_{21} = \frac{E_1 A_1}{L_1} \sin \alpha \cos \alpha + \frac{E_3 A_3}{L_3} \sin \beta \cos \beta \\ k_{22} &= \frac{E_1 A_1}{L_1} \sin \alpha + \frac{E_2 A_2}{L_2} + \frac{E_3 A_3}{L_3} \sin \beta \end{aligned}$$

onde E_1, E_2 e E_3 são os módulos de elasticidade dos materiais de cada uma das três barras, L_1, L_2 e L_3 são os comprimentos de cada uma delas, α e β são os ângulos entre as barras da figura A.1.

A formulação apresentada evidencia o relacionamento dos deslocamentos, u_{2x} e u_{2y} , com os valores das variáveis de projeto, e das áreas da seção transversal das barras (A_i), dada a utilização dessas variáveis no cálculo da matriz de rigidez da estrutura.

Após a determinação dos deslocamentos, é possível obter os esforços e tensões normais em cada ponto ao longo das barras da treliça.

A obtenção dessas informações envolve operações matriciais que consideram características elásticas e geométricas de cada barra, além dos deslocamentos dos nós de suas extremidades. Tais operações são apresentadas em detalhes em (Gere e Weaver, 1987).

Observa-se que mesmo quando a função objetivo é linear (volume/peso da estrutura no caso) as restrições (de tensões e deslocamentos no caso) são funções não-lineares das variáveis de projeto.

Determinar se uma dada solução/projeto candidato atende às restrições envolve resolver o sistema de equações A.3. O que é muito mais custoso do que o cálculo da função objetivo neste caso.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)