

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

**Otimização de Parâmetros de Projeto de
Sistemas Mecânicos através de Algoritmo
Genético Multi-Objetivos**

Autor: Roberto Luiz Escobar
Orientadora: Prof. Dra. Kátia Lucchesi Cavalca

13/2007

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

Otimização de Parâmetros de Projeto de Sistemas Mecânicos através de Algoritmo Genético Multi-Objetivos

Autor: Roberto Luiz Escobar
Orientadora: Prof. Dra. Kátia Lucchesi Cavalca

Curso: Engenharia Mecânica
Área de concentração: Projeto Mecânico

Dissertação de mestrado acadêmico apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica, como requisito para a obtenção do título de Mestre em Engenharia Mecânica.

Campinas, 2007.
SP - Brasil

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Es18o Escobar, Roberto Luiz
Otimização de parâmetros de projeto de sistemas mecânicos através de algoritmo genético multi-objetivos / Roberto Luiz Escobar. --Campinas, SP: [s.n.], 2007.

Orientador: Kátia Lucchesi Cavalca
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Otimização matemática. 2. Algoritmos genéticos.
3. Planejamento experimental. 4. Máquinas - Vibração.
I. Cavalca, Kátia Lucchesi. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

Título em Inglês: Optimization in design parameters of mechanical systems using multi-objective genetic algorithm

Palavras-chave em Inglês: Multi-objective optimization, genetic algorithm, Response surface methodology, Rotating machines

Área de concentração: Mecânica dos Sólidos e Projeto Mecânico

Titulação: Mestre em Engenharia Mecânica

Banca examinadora: Alberto Luiz Serpa e Luiz de Paula do Nascimento

Data da defesa: 16/02/2007

Programa de Pós-Graduação: Engenharia Mecânica

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

DISSERTAÇÃO DE MESTRADO ACADEMICO

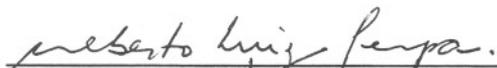
**Otimização de Parâmetros de Projeto de
Sistemas Mecânicos através de Algoritmo
Genético Multi-Objetivos**

Autor: Roberto Luiz Escobar

Orientadora: Prof. Dra. Katia Lucchesi Cavalca



Prof. Dra. Kátia Lucchesi Cavalca
Universidade Estadual de Campinas



Prof. Dr. Alberto Luiz Serpa
Universidade Estadual de Campinas



Prof. Dr. Luiz de Paula do Nascimento
Universidade Estadual Paulista – Ilha Solteira

Campinas, 16 de Fevereiro de 2007.

Dedicatória

Dedico este trabalho à minha querida esposa Ana Maria.

Agradecimentos

À Deus pela vida e pela capacidade de aprender.

Agradeço a um conjunto de pessoas, sem as quais o caminho a ser percorrido seria muito mais árduo:

À minha esposa Ana e meus filhos Isabela e Marcelo, pela alegria e pelo apoio;

Aos meus pais pelos ensinamentos e por seus valores de honestidade e humildade;

A professora Dra. Kátia Lucchesi Cavalca pela oportunidade e orientação deste trabalho;

Aos professores do DPM e do DMC pelas orientações e ensinamentos sem os quais este trabalho não seria possível;

Ao Eduardo e ao Hélio pela ajuda na implementação dos programas e análises experimentais;

Aos colegas do Laboratório de Máquinas Rotativas pelas conversas sobre a tese e pelos momentos de motivação e descontração;

A Unicamp pela infra-estrutura e a oportunidade concedidas;

À Vulkan do Brasil pelo apoio, suporte e confiança;

A todos que contribuíram para a minha evolução acadêmica e humana.

“...A Vida é feita de fases, ter sucesso é conseguir planejar uma nova fase sem saber quando ela irá começar e nem como será.”

O Autor.

Resumo

Escobar, Roberto L. **Otimização de Parâmetros de Projeto de Sistemas Mecânicos através de Algoritmo Genético Multi-Objetivos**. Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2007. 150p. Dissertação (Mestrado).

Os sistemas mecânicos são projetados para desempenhar funções específicas, e por essa razão as suas funções devem ser medidas para garantir seu desempenho dentro de uma certa precisão ou tolerância. A grande complexidade em se projetar e analisar novos projetos é a inserção de novas tecnologias, que envolvem aspectos multidisciplinares. Assim, o desenvolvimento e melhoria de projetos e produtos colocam o engenheiro projetista frente às diversas fontes de variabilidade, como por exemplo, as propriedades dos materiais, condições operacionais e ambientais e incertezas nas suposições feitas sobre seu funcionamento. Em termos de modelagem matemática, as aproximações inerentes e hipóteses feitas durante a concepção do sistema, conduzem normalmente a diferentes respostas obtidas através de simulações e/ou medidas experimentais. Dessa forma, em uma fase anterior à modelagem matemática, durante a concepção do sistema ou produto, as aplicações de ferramentas estatísticas e métodos de otimização podem fornecer estimativas sobre faixas de valores ou valores ótimos para parâmetros significativos de projeto, dentro do espaço experimental estudado. Esse tipo de abordagem estatística teve sua fundamentação teórica durante as décadas de 20 e 30 por Fisher, com a aplicação da teoria estatística sob diversos aspectos, como por exemplo: testes de hipóteses, estimativa de parâmetros, seleção de modelos, planejamento experimental e, mais tarde, no controle e melhoria de processos e produtos. Assim, este trabalho propõe um procedimento de estudo e otimização, integrando a teoria de planejamento experimental, a metodologia da superfície de resposta e otimização multi-objetivos através de algoritmos genéticos, para se obter a otimização dos parâmetros de projeto de componentes mecânicos. Em específico, foram utilizados dados de um sistema rotor-mancal e o estudo implica em minimizar as amplitudes no domínio da frequência. Outro objetivo deste trabalho, foi desenvolver um programa para otimização multi-objetivos através de algoritmos genéticos.

Palavras-chave: *otimização multi-objetivos, algoritmo genético, metodologia da superfície de resposta, máquinas rotativas*

ABSTRACT

Escobar, Roberto L. **Optimization in Design Parameters of Mechanical Systems using Multi-Objective Genetic Algorithm.** Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2007. 150p. Dissertação (Mestrado).

The mechanical systems are designed to be applied to any specific situations, and in this way their features should be measured to guarantee confidence to the systems. Their development and analysis expose the designer to a series of unknown parameters from several sources such as material properties, environmental and operational conditions. In terms of mathematical modeling, the inherent approximation and hypotheses made during system conception lead to different responses obtained by simulations and/or experimental measurements. So, in a previous phase of mathematical modeling, during the design analysis, the application of statistical tools and optimization methods is possible to estimate the values and/or ranges of the critical design parameters inside an experimental space. The connection between optimization and statistical data back at least to the early part of the 20th century and encompasses many aspects of applied and theoretical statistics, including hypothesis testing, parameter estimation, model selection, design of experiments and process and product control. So, this work proposes a link between theory of design of experiments, response surface methodology and multi-objective optimization using genetic algorithms, in order to optimize parameters for mechanical components. This study makes possible to verify the application of multi-objective optimization using genetic algorithms in design parameters and optimize them. A rotor-bearing system was used and amplitude in frequency domain was minimized. An experimental software for multi-objective optimization using genetic algorithm was developed.

Keywords: *multi-objective optimization, genetic algorithm, response surface methodology, rotating machines.*

Índice

Lista de Figuras	xiv
Lista de Tabelas	xvii
Nomenclatura	xviii
Capítulo 1	
Introdução	1
1.1 – Objetivos do Trabalho	3
1.2 – Conteúdo do Trabalho	4
1.3 – Estudo, implementação computacional e aplicação de técnicas de otimização multi-objetivos em componentes mecânicos para máquinas rotativas	5
Capítulo 2	
Revisão Bibliográfica	8
2.1 – Estatística Aplicada e Metodologia de Projeto	8
2.2 – Dinâmica de Rotores	10
2.3 – Otimização Multi-Objetivos	12
Capítulo 3	
Otimização Simples e Multi-Objetivos	16
3.1 – DOE - Delineamento de Experimentos	16
3.1.1 – Considerações sobre Projeto e Planejamento Robusto	17
3.2 – Metodologia de Superfície de Resposta – MSR	19
3.3 – Otimização	20

3.3.1 – Otimização Simples e Multi-Objetivos	21
3.3.2 – Diferenças Fundamentais entre Simples-Objetivo e Multi-Objetivos	22
3.3.3 – Otimização Multi-Objetivos – Duas abordagens	23
3.4 – Métodos Evolutivos	27
3.5 – Otimização Multi-Objetivos – Formulação	28
3.6 – Dominância e Soluções Pareto-Ótimas	29
3.7 – Objetivos em Otimização Multi-Objetivos	30
3.8 – Conceito de Dominância de Pareto	31
3.8.1 – Definição 1	31
3.8.2 – Propriedades da Relação de Dominância	32
3.8.3 – Otimalidade de Pareto	32
3.8.4 – Definição 2	33
3.8.5 – Definição 3	34
3.9 – Comentários Finais	34
 Capítulo 4	
Algoritmos Evolucionários para Otimização Multi-Objetivos	35
4.1 – Algoritmos Genéticos	36
4.2 – Representação das Soluções	37
4.2.1 – Representação Binária	38
4.2.2 – Representação Contínua ou Verdadeira	38
4.3 – Operadores Genéticos	39
4.3.1 – Operador de Seleção	39
4.3.2 – Seleção Proporcional	41
4.3.3 – Operador de Cruzamento	42
4.3.4 – Operador de Mutação	43
4.3.5 – Operador de Elitismo	43
4.4 – Passos para Implementar um Algoritmo Genético	44
4.5 – Diferenças entre os Algoritmos Genéticos e os Métodos de Otimização Tradicionais	44
4.6 – Algoritmos Genéticos para Otimização Multi-Objetivos	45

4.7 – Algoritmos Evolutivos Multi-Objetivos Não-Elististas	46
4.8 – MOGA (Multi Objective Genetic Algorithm)	47
4.8.1 – Descrição do MOGA	48
4.8.2 – Procedimento de Ajuste do MOGA	51
4.8.3 – Complexidade Computacional	52
4.8.4 – Vantagens	52
4.8.5 – Desvantagens	53
4.9 – NSGA (Non-Dominated Sorting Genetic Algorithm)	53
4.9.1 – Descrição do NSGA	53
4.9.2 – Procedimento de Ajuste do NSGA	57
4.9.3 – Complexidade Computacional	58
4.9.4 – Vantagens	58
4.9.5 – Desvantagens	59
4.10 – Algoritmos Evolutivos Multi-Objetivos Elististas	59
4.11 – NSGA-II (Non-Dominated Sorting Genetic Algorithm II)	61
4.11.1 – Descrição do NSGA-II	62
4.11.2 – Procedimento de Ajuste do NSGA-II	64
4.11.3 – Operador de Seleção por Torneio Agrupado	65
4.11.4 – Distância de Agrupamento	66
4.11.5 – Procedimento de Cálculo da Distância de Agrupamento	66
4.11.6 – Complexidade Computacional	67
4.11.7 – Vantagens	68
4.11.8 – Desvantagens	68
4.12 – SPEA (Strength Pareto Evolutionary Algorithm)	69
4.12.1 – Descrição do SPEA	69
4.12.2 – Procedimento de Ajuste do SPEA	71
4.12.3 – Algoritmo de Agrupamento	72
4.12.4 – Procedimento de Agrupamento	73
4.12.5 – Complexidade Computacional	74
4.12.6 – Vantagens	74
4.12.7 – Desvantagens	75

4.13 – Comentários Finais	76
Capítulo 5	
Simulações Teóricas e seus Resultados	77
5.1 – Delineamento de Experimentos (DOE) e Superfície de Resposta (MSR)	81
5.2 – Codificação das Variáveis	87
5.2.1 – Distância entre Mancais L	87
5.2.2 – Folga Radial F	88
5.2.3 – Temperatura do Óleo T	88
5.3 – Funções Polinomiais	88
5.4 – Aplicação do Algoritmo Genético Multi-Objetivos	92
5.4.1– Processo de Otimização Multi-Objetivos	94
5.5 – Resultados	96
Capítulo 6	
Aplicações Experimentais e seus Resultados	100
6.1 – Delineamento de Experimentos (DOE) e Superfície de Resposta (MSR)	102
6.1.1 – Medições – Amplitudes no domínio da frequência	103
6.2 – Funções Polinomiais	108
6.2.1 – Superfícies de Resposta	109
6.3 – Aplicação do Algoritmo Genético Multi-Objetivos	113
6.3.1 – Processo de Otimização Multi-Objetivos	113
6.4 – Resultados	116
Capítulo 7	
Conclusões e Propostas Futuras	120
7.1 – Conclusões	120
7.2 – Propostas Futuras	122
Bibliografia	123

Lista de Figuras

Fig. 3.1 – Exemplo que ilustra várias opções de compra de carro	23
Fig. 3.2 – Fluxo de Otimização Multi-Objetivos Ideal	25
Fig. 3.3 – Fluxo de Otimização Multi-Objetivos Ponderada	26
Fig. 3.4 – Fronteira Pareto-Ótima e Diversidade de Soluções	31
Fig. 3.5 – Soluções Pareto-Ótimas e Espaços de Objetivos	33
Fig. 3.6 – Soluções Pareto-Ótimas Globais e Locais	34
Fig. 4.1 – Fluxograma de Algoritmo Genético	37
Fig. 4.2 – Representação Binária	38
Fig. 4.3 – Operador de Cruzamento	43
Fig. 4.4 – Operador de Mutação	43
Fig. 4.5 – Espaço de procura de dois objetivos e a fronteira ótima de Pareto	48
Fig. 4.6 – Procedimento de ranking do MOGA	48
Fig. 4.7 – Fronteiras no NSGA	54
Fig. 4.8 – Agrupamento no NSGA-II	63
Fig. 4.9 – Cálculo da distância de agrupamento	66
Fig. 4.10 – Esquema de Ajuste do SPEA	71
Fig. 5.1 – Simplificação de modelo de um rotor.	78
Fig. 5.2 – Conjunto eixo-mancaal-massa do programa ROTORTEST®.	80
Fig. 5.3a – Simulação 1 ($x_1 = 460\text{mm}$; $x_2 = 180\mu\text{m}$; $x_3 = 30^\circ\text{C}$)	82
Fig. 5.3b – Simulação 2 ($x_1 = 600\text{mm}$; $x_2 = 180\mu\text{m}$; $x_3 = 30^\circ\text{C}$)	83
Fig. 5.3c – Simulação 3 ($x_1 = 460\text{mm}$; $x_2 = 250\mu\text{m}$; $x_3 = 30^\circ\text{C}$)	83

Fig. 5.3d – Simulação 4 ($x_1 = 600\text{mm}$; $x_2 = 250\mu\text{m}$; $x_3 = 30^\circ\text{C}$)	84
Fig. 5.3e – Simulação 5 ($x_1 = 460\text{mm}$; $x_2 = 250\mu\text{m}$; $x_3 = 40^\circ\text{C}$)	84
Fig. 5.3f – Simulação 6 ($x_1 = 600\text{mm}$; $x_2 = 180\mu\text{m}$; $x_3 = 40^\circ\text{C}$)	85
Fig. 5.3g – Simulação 7 ($x_1 = 460\text{mm}$; $x_2 = 250\mu\text{m}$; $x_3 = 40^\circ\text{C}$)	85
Fig. 5.3h – Simulação 8 ($x_1 = 600\text{mm}$; $x_2 = 250\mu\text{m}$; $x_3 = 40^\circ\text{C}$)	86
Fig. 5.4 – Resultados do DOE mostrado no cubo dimensional.	87
Fig. 5.5 – Entrada de dados no Programa ÓTIMA®.	89
Fig. 5.6 – Superfície de Resposta f_1 em relação a x_1 e x_2	90
Fig. 5.7 – Superfície de Resposta f_1 em relação a x_1 e x_3	91
Fig. 5.8 – Superfície de Resposta f_2 em relação a x_1 e x_2	91
Fig. 5.9 – Superfície de Resposta f_2 em relação a x_1 e x_3	92
Fig. 5.10 – Entrada de dados para a otimização multi-objetivos	93
Fig. 5.11 – Geração Inicial	95
Fig. 5.12 – A linha tracejada representa a fronteira de Pareto.	95
Fig. 5.13 – Relação entre Folga Radial e Distância entre Mancais.	96
Fig. 5.14 – Relação entre Temperatura do Óleo e Distância entre Mancais.	97
Fig. 5.15 – Relação entre Temperatura do Óleo e Folga Radial	98
Fig. 5.16 – Resultado da Otimização Multi-Objetivos	99
Fig. 6.1 – Representação da bancada experimental	100
Fig. 6.2 – Esquema de sensores e instrumentos nos mancais	101
Fig. 6.3 – Esquema da instrumentação utilizada	102
Fig. 6.4 – Bancada Experimental – LAMAR – UNICAMP	103
Fig. 6.5a – Excitação Horizontal - Amplitudes – Experimento 1	104
Fig. 6.5b – Excitação Vertical - Amplitudes – Experimento 1	104
Fig. 6.5c – Excitação Horizontal - Amplitudes – Experimento 2	104
Fig. 6.5d – Excitação Vertical - Amplitudes – Experimento 2	104
Fig. 6.5e – Excitação Horizontal - Amplitudes – Experimento 3	104
Fig. 6.5f – Excitação Vertical - Amplitudes – Experimento 3	104
Fig. 6.5g – Excitação Horizontal - Amplitudes – Experimento 4	105
Fig. 6.5h – Excitação Vertical - Amplitudes – Experimento 4	105
Fig. 6.5i – Excitação Horizontal - Amplitudes – Experimento 5	105

Fig. 6.5j – Excitação Vertical - Amplitudes – Experimento 5	105
Fig. 6.5l – Excitação Horizontal - Amplitudes – Experimento 6	105
Fig. 6.5m – Excitação Vertical - Amplitudes – Experimento 6	105
Fig. 6.5n – Excitação Horizontal - Amplitudes – Experimento 7	106
Fig. 6.5o – Excitação Vertical - Amplitudes – Experimento 7	106
Fig. 6.5p – Excitação Horizontal - Amplitudes – Experimento 8	106
Fig. 6.5q – Excitação Vertical - Amplitudes – Experimento 8	106
Fig. 6.6 – Entrada de dados no Programa ÓTIMA®.	108
Fig. 6.7 – Superfície de Resposta para f2 (x1 = distância entre mancais e x3 = temperatura do óleo, x2 = constante = 180 μ m)	110
Fig. 6.8 – Superfície de Resposta para f2 (x1 = distância entre mancais e x2 = folga radial, x3 = constante = 26°C)	110
Fig. 6.9 – Superfície de Resposta para f3 (x1 = distância entre mancais e x3 = temperatura do óleo, x2 = constante = 180 μ m)	111
Fig. 6.10 – Superfície de Resposta para f3 (x1 = distância entre mancais e x2 = folga radial, x3 = constante = 26°C)	111
Fig. 6.11 – Superfície de Resposta para f1 (x1 = distância entre mancais e x2 = folga radial, x3 = constante = 26°C)	112
Fig. 6.12 – Superfície de Resposta para f1 (x1 = distância entre mancais e x3 = temperatura do óleo, x2 = constante = 180 μ m)	112
Fig. 6.13 – Geração Inicial (f1- frequência x f2 – amplitude mancal 1)	114
Fig. 6.14 – Geração Inicial (f1- frequência x f3 – amplitude mancal 2)	114
Fig. 6.15 – Soluções e Fronteira Pareto-Ótima (f1- frequência x f2 – amplitude mancal 1)	115
Fig. 6.16 – Soluções e Fronteira Pareto-Ótima (f1- frequência x f3 – amplitude mancal 2)	115
Fig. 6.17 – Relação entre Folga Radial e Distância entre Mancais	116
Fig. 6.18 – Relação entre Temperatura do Óleo e Distância entre Mancais	117
Fig. 6.19 – Relação entre Temperatura do Óleo e Folga Radial	118
Fig. 6.20 – Resultado da Otimização Multi-Objetivos	119

Lista de Tabelas

Tabela 5.1 – DOE – Ordem de ensaios	81
Tabela 5.2 – Parâmetros para simulação teórica	82
Tabela 5.3 – Valores das Simulações Teóricas.	86
Tabela 6.1 – Parâmetros para análise experimental	103
Tabela 6.2 – Resultados Experimentais	107

Nomenclatura

Letras Latinas

d_{ij} – distância entre soluções

f_1, f_2, \dots, f_M – funções objetivos

f_m – funções objetivos gerais

g_j – funções de restrição (desigualdades)

h_k – funções de restrição (igualdades)

n – número de variáveis

p – probabilidade de cada solução ser repetida

r – valor aleatório

r_i – classificação das soluções

x_1, x_2, x_3 – variáveis codificadas

C_i – número de cópias na seleção

F – valor de ajuste (fitness)

J – número de restrições (desigualdades)

K – número de restrições (igualdades)

M – número de funções objetivos

N – tamanho da população

P – população de soluções

P^o – população de soluções Pareto-Ótimas

P_t – população externa de soluções

Q_t – população de descendentes

Rt – população combinada
S – espaço de busca ou espaço de objetivos
Sh – função de divisão
Si – resistência da solução
T – somatório do valor esperado

Letras Gregas

a – coeficientes lineares e bi-lineares
a – expoente na função de divisão
b – coeficientes quadráticos
e – erro absoluto
l – número paramétrico de quantidade de elitismo
 η_i – soluções dominantes
 η_{Ci} – conta do nicho
 η_L – função polinomial de primeira ordem
 η_Q – função polinomial de segunda ordem
 σ_{SHARE} – parâmetro de controle da dispersão das soluções
 μ – número de soluções na classificação

Abreviações

ANOVA – Análise de Variância;
CEP – Controle Estatístico de Processo;
DOE – Delineamento de Experimentos;
FRF – Função de Resposta em Frequência;
MSR – Metodologia de Superfície de Resposta
SUS – Amostragem Universal Estocástica (Stochastic Universal Sampling)

Siglas

LAMAR – Laboratório de Máquinas Rotativas

DPM – Departamento de Projeto Mecânico

DMC – Departamento de Mecânica Computacional

Capítulo 1

Introdução

Recentemente, com o avanço tecnológico da ciência da computação, aliado ao desenvolvimento acelerado de computadores, a capacidade de resolver problemas de engenharia aumentou de forma significativa. Tornou-se possível a solução de sistemas de equações com número consideravelmente elevado de variáveis, novas técnicas de programação foram criadas e novos desenvolvimentos na engenharia computacional floresceram.

Os projetos de sistemas mecânicos geralmente envolvem uma ampla gama de conhecimentos multidisciplinares, que englobam desde a modelagem matemática e as simulações numéricas, até a realização de experimentos, quando possível. Da mesma forma, nem sempre é possível resolver simultaneamente todos os aspectos multidisciplinares e as tomadas de decisão, mesmo tratando-se de decisões técnicas, devendo-se muitas vezes baseá-las em análises subjetivas.

Neste sentido, muitas técnicas de otimização e de soluções de problemas multi-objetivos ou multi-disciplinares foram desenvolvidas. Em termos de modelagem matemática, muitas vezes as aproximações e hipóteses consideradas no estudo do sistema abrem margem a respostas analíticas e numéricas distintas das respostas obtidas em medições experimentais. Porém é possível, em

uma fase anterior ao modelo matemático, através da aplicação de ferramentas estatísticas e de otimização numérica, estimar os valores dos parâmetros mais críticos do sistema dentro do espaço experimental de estudo.

Um exemplo de análise é o comportamento dinâmico das máquinas rotativas. Sendo esta análise bastante complexa, uma vez que estas máquinas são constituídas por rotores, eixos, acoplamentos, mancais, carcaça, fundação e outros equipamentos auxiliares. O comportamento destas máquinas é influenciado em maior ou menor grau por estes componentes e para estudar seu comportamento dinâmico, é necessário levar em consideração a interação de todos os componentes.

O processo de otimização multi-objetivos tem atraído investimentos e pesquisas principalmente para atender à indústria aeroespacial, onde se necessita obter respostas rápidas e robustas para problemas multi-disciplinares. Vários estudos caminham para a aplicação de técnicas de otimização multi-objetivos sem a utilização de gradientes ou equações diferenciais, solucionando problemas com ótimos locais de forma robusta. Os algoritmos evolutivos são exemplos de tais técnicas.

A partir de dados experimentais, medidos ou simulados dos componentes ou de um sistema mecânico, podemos identificar os parâmetros de maior influência, assim como otimizá-los para se obter um projeto mais robusto mediante certas condições de funcionamento estudadas. O resultado proposto será uma série de soluções que representam as melhores soluções possíveis entre todas, considerando-se as relações conflitantes entre elas. Os Algoritmos Evolutivos juntamente com as teorias de Planejamento Experimental e Metodologia da Superfície de Resposta auxiliam na exploração destas soluções otimizadas.

O presente trabalho propõe o estudo da aplicação de otimização Multi-Objetivos através de Algoritmo Genético, sendo este um Algoritmo Evolutivo, integrado com a teoria sobre Planejamento Experimental e Metodologia da Superfície de Resposta, aplicado ao problema de comportamento dinâmico rotor-mancal.

Para o estudo e validação dos programas desenvolvidos neste trabalho foi utilizado um sistema vibratório composto de dois mancais hidrodinâmicos cilíndricos, um acionamento elétrico, acoplamento flexível e um rotor preso a um eixo bi-apoiado nestes mancais hidrodinâmicos. Esta configuração atualmente está montada em bancada experimental, simulando um rotor. O conjunto se encontra montado sobre uma fundação inercial, com base suficientemente rígida de modo a eliminar possíveis influências na resposta dinâmica do rotor. O modelo matemático do sistema baseia-se no método de elementos finitos para o rotor, e no modelo dos mancais, através da equação de Reynolds, por diferenças finitas. As respostas foram obtidas através de simulações numéricas realizadas no programa ROTORTEST® e métodos experimentais utilizando a bancada de teste de rotores do Laboratório de Máquinas Rotativas, LAMAR, da equipe da Profa. Dra. Kátia L. Cavalca da UNICAMP.

Optou-se pela aplicação em um sistema conhecido para facilitar a validação da proposta, mesmo sendo este sistema passível de otimização por outros métodos mais conhecidos.

1.1 Objetivos do trabalho

Este trabalho tem como objetivo o estudo da aplicação de Otimização Multi-Objetivos através de Algoritmo Genético integrado com a teoria sobre Planejamento Experimental e Metodologia da Superfície de Resposta, a um problema de comportamento dinâmico rotor-mancais, para o levantamento e otimização de parâmetros críticos dos componentes mecânicos, bem como o desenvolvimento de um programa baseado nestas teorias. Trata-se de um estudo que permite a verificação da possibilidade de aplicação dos algoritmos genéticos no estudo de sistemas mecânicos e no auxílio da identificação dos parâmetros mais críticos do sistema ou componente em estudo, podendo-se reduzir de forma significativa, equações matemáticas e medições experimentais envolvidas na combinação teórico-experimental do sistema.

Os programas desenvolvidos têm como finalidade: o aprendizado com os métodos de otimização multi-objetivos inseridos nas teorias acima descritas, e uma análise de viabilidade de aplicação destas teorias voltadas para a área experimental. As primeiras versões dos programas desenvolvidos são trabalhos de pesquisa. Em termos de aplicações, os programas propostos e implementados neste trabalho, bem como a teoria estudada tem como objetivo auxiliar o

pesquisador quanto ao desenvolvimento e estudo de sistemas, componentes e processos mecânicos, nas áreas de:

- Concepção de novos projetos;
- Desenvolvimento de processos e otimização;
- Confiabilidade e teste de vida;
- Configuração de novos produtos;
- Melhoria de processos em engenharia.

1.2 Conteúdo do trabalho

O trabalho foi organizado de forma a apresentar a seqüência dos temas necessários para o seu desenvolvimento: estudar a viabilidade da aplicação de otimização multi-objetivos evolutivos no estudo de componentes mecânicos e estender este estudo ao sistema completo. Dessa forma, utilizar métodos de otimização inadequados, por exemplo, pode gerar resultados duvidosos ou não conclusivos, o que às vezes não corresponde às soluções reais, gerando um aumento do tempo dispensado e podendo inviabilizar tais aplicações. Assim, são apresentadas brevemente noções sobre Delineamento de Experimentos e Metodologia de Superfícies de Resposta, aprofundando-se em métodos de Otimização Multi-Objetivos através de Algoritmos Evolutivos, avançando até a proposta de trabalho, a avaliação e o desenvolvimento de um programa computacional para Otimização Multi-Objetivos através de Algoritmos Genéticos.

Assim, foram propostos sete capítulos e referência bibliográfica citada apresentando o seguinte conteúdo:

1. Introdução – é apresentada a proposta de trabalho, seus objetivos e o conteúdo do trabalho;

2. Revisão Bibliográfica – é apresentada uma revisão teórica sobre as teorias estudadas;

3. *Otimização Simples e Multi-Objetivos* – é apresentada a teoria e métodos de otimização abrangendo as diferenças entre simples-objetivo e multi-objetivos;

4. *Algoritmos Genéticos para Otimização Multi-Objetivos* – são apresentados vários métodos e modelos de algoritmos genéticos, bem como suas vantagens e desvantagens tanto em relação aos métodos tradicionais quanto entre si;

5. *Simulações Teóricas e seus Resultados* – utilização do programa RotorTest ® para as simulações teóricas e aplicação dos métodos propostos, apresentando os resultados da otimização multi-objetivos;

6. *Aplicações Experimentais e seus Resultados* – aplicação em bancada dos métodos propostos, apresentando os resultados da otimização multi-objetivos;

7. *Conclusões e Propostas futuras* – são discutidas as vantagens, limitações da aplicação e trabalhos futuros.

Referências Bibliográficas;

1.3 Estudo, implementação computacional e aplicação de técnicas de otimização multi-objetivos em componentes mecânicos para máquinas rotativas.

Através da análise da dinâmica de rotores, é possível identificar diversos tipos de falhas em equipamentos mecânicos. Em rotores, ocorrem vibrações que podem ser originárias, por exemplo, de acoplamentos desalinhados, desbalanceamento de componentes mecânicos, deformação de eixos, folgas excessivas ou não nos mancais, falta de rigidez no conjunto. O monitoramento e os registros dessas vibrações podem ser feitos, através de sensores posicionados em pontos estratégicos do conjunto mecânico, transformando a energia mecânica de vibração em sinais elétricos, registrados e analisados posteriormente. Os níveis de vibrações em componentes ou conjuntos mecânicos podem ser representados de diversas formas, sendo mais comum sua

representação no domínio da frequência, ou seja, a amplitude da vibração [μm ou mm/N] é dada em função da frequência [Hz ou rpm].

Vários modelos matemáticos têm sido concebidos para simular as condições de trabalho de sistemas rotativos e avaliar seu comportamento real, isto é, a pesquisa segue tentando modelar máquinas reais como turbogeradores e bombas multi-estágios, no caso de máquinas rotativas de grande porte horizontais. Grande parte das medições em rotores é feita nos mancais, e a mobilidade mecânica determina as amplitudes de movimento relativas entre eixo, mancais e estruturas de suporte. Normalmente, as medições de deslocamento devem ser realizadas em pontos nos quais as amplitudes são maiores. Dessa forma, concentrando-se na formulação e discretização dos mancais há, geralmente, uma maior precisão dos resultados de modelos matemáticos. A aplicação da teoria de Reynolds na formulação dos fenômenos de lubrificação hidrodinâmica permite obter modelos confiáveis para mancais hidrostáticos e hidrodinâmicos de acordo com Silveira (2003).

Em termos de ajuste de modelos em rotores, há diversos trabalhos que tentam corrigir os modelos obtidos pelo método de elementos finitos, através do processamento da resposta dinâmica de estruturas testadas experimentalmente. A validação de tais modelos é feita geralmente, a partir da comparação dos resultados de frequências naturais e modos próprios adquiridos em testes modais. Com esse procedimento procura-se corrigir os parâmetros imprecisos no modelo, de forma a garantir a concordância entre as predições numéricas e os testes experimentais. Portanto, sendo possível realizar simulações computacionais, estas podem prever certos comportamentos do sistema ou de componentes que são úteis para uma posterior montagem experimental. Para a simulação de máquinas rotativas, os modelos computacionais mais recentes utilizam o método de elementos finitos para a discretização do sistema, a saber, eixos, rotores e eventualmente mancais. Em se tratando de discretização, devem ser consideradas hipóteses sobre diversas situações de funcionamento, como: parametrização incorreta ou modelos matemáticos imprecisos, que geram erros de modelagem, o que é uma realidade para toda e qualquer representação matemática de sistemas físicos. Além disso, quando aplicado ao sistema físico real, uma série de efeitos inerentes à montagem e às condições externas e internas de

operação não são necessariamente passíveis de representação matemática ou numérica, sendo em algumas situações, porém, fatores de influência na resposta dinâmica do sistema.

A idéia deste trabalho em se utilizar um método estatístico de otimização multi-objetivos, é fazer um pré-estudo das faixas de operação do sistema e se obter conjuntos ótimos de parâmetros, sob os quais os modelos numéricos ou analíticos possam ser mais explorados, com maior confiança.

Assim, para o desenvolvimento do trabalho foi utilizado um programa de simulação de rotores, ROTORTEST®, desenvolvido pelo grupo de pesquisa nos últimos anos. O modelo simulado foi baseado no trabalho de Cavalcante e Cavalca (1998), cuja configuração representa um rotor horizontal apoiado sobre dois mancais hidrodinâmicos. Segundo Cavalcante e Cavalca (1998), a validação do modelo experimental apresentou algumas lacunas, devido ao fato de não haver um tratamento teórico para o experimento. A aplicação da teoria de planejamento experimental considerando Delineamento de Experimentos, da Metodologia da Superfície de Resposta (MSR) e da Otimização Multi-Objetivos através de Algoritmos Genéticos, tem como objetivo, através das simulações computacionais, tentar identificar os parâmetros críticos do sistema rotor-mancal para uma determinada faixa de trabalho e otimizá-los. A estratégia utilizada para o rotor foi escolher um planejamento experimental simples, com três parâmetros, possibilitando uma simulação teórica que permitiu que as simulações numéricas fossem realizadas experimentalmente em um período de tempo relativamente curto. As respostas estudadas foram a primeira frequência natural do sistema [rpm ou Hz] e a amplitude da vibração [μm ou mm/N]. Os fatores de projeto variados foram: comprimento do eixo (distância entre mancais), a folga radial nos mancais e a temperatura do óleo. Também foram geradas funções polinomiais através do Método de Superfície de Resposta, sendo estas funções otimizadas através do Algoritmo Genético Multi-Objetivos.

O tema do presente trabalho foi escolhido com a finalidade de complementar o programa *OTIMA^Ô*, desenvolvido durante o doutorado de Zilda de Castro Silveira. Desta forma, foi necessária uma intensa pesquisa bibliográfica, para efetuar o levantamento do que se tem disponível sobre o assunto.

Capítulo 2

Revisão Bibliográfica

2.1 Estatística Aplicada e Metodologia de Projeto

Com a evolução da tecnologia e, conseqüentemente, com o aumento da complexidade dos sistemas, a necessidade de projetos multi-disciplinares tem impactado diretamente o desenvolvimento e a avaliação de projetos englobando áreas como confiabilidade, planejamentos experimentais e otimização de sistemas. Segundo Silveira (2003), um projeto é dito factível, se o seu desempenho estiver de acordo com as especificações, podendo ser continuamente melhorado sob diversos aspectos. Entretanto, o estudo e a análise da verificação de um projeto necessitam de diversas etapas, indo desde as idéias preliminares até o desenvolvimento de modelos matemáticos incluindo diferentes técnicas de análise e comparação de resultados, podendo compreender simulações numéricas e realização de experimentos, que auxiliam no entendimento do problema físico. Neste trabalho, o enfoque é dado em relação à área de otimização multi-objetivos, que têm em comum a utilização de ferramentas estatísticas para o estudo do comportamento inicial de um sistema ou componente mecânico até a completa análise do comportamento do sistema.

Considerando ferramentas estatísticas, Hann e Hoerl (1998) citam que as aplicações nos meios de produção geraram resultados satisfatórios e de grande contribuição para o controle de diversas atividades industriais, desde a aplicação de gráficos de controle propostos por Shewart durante a década de 30, às cartas de controle de processo (CEP), até a técnica de Seis Sigma aplicada pela Motorola e pela *General Eletric*. Porém, estas ferramentas e suas aplicações forneceram a falsa idéia de que a Estatística não poderia ser aplicada em outros segmentos. Na área de desenvolvimento de produtos, os autores ressaltam que a qualidade final, normalmente atribuída aos meios de produção, é na prática determinada durante o desenvolvimento do produto, ou ainda, na fase de projeto propriamente dita. Portanto, é nesta fase que se obtêm um melhor desempenho, uma melhor qualidade do produto. Menções feitas, anteriormente por Taguchi e Deming nas décadas de 60 e 70, enfatizaram que uma grande parte da variabilidade é atribuída durante a própria concepção do sistema, na qual devem ser usadas ferramentas estatísticas adequadas especificamente voltadas ao projeto.

O termo Projeto é associado ao ato de gerar novas idéias, novas soluções para diversos problemas da sociedade como um todo, ou seja, a Engenharia utiliza conceitos das ciências puras, como por exemplo, física e matemática em conjunto com tecnologias para construir algo que ainda não existe ou melhorar produtos. Por outro lado, as ciências puras procuram identificar elementos existentes na natureza, tentando explicá-los. Silveira (2003) relaciona estas e outras definições, através da definição e evolução da linha de pesquisa em Metodologia de Projeto, desde suas primeiras definições até aplicações atuais, englobando a visão de Taguchi sobre a utilização de conceitos estatísticos na fase de desenvolvimento de projetos.

Montgomery (1999) descreve outra consideração sobre projeto robusto onde coloca o conceito de projeto robusto sob dois aspectos, sendo o primeiro associado à minimização do desvio de desempenho em função de ruídos ou variáveis não controladas e o segundo relativo à minimização do desvio de desempenho causado pela variação de algum fator ou parâmetro de projeto. Silveira e Cavalca (2004) propõem uma abordagem mais geral que tem como objetivo superar as limitações do método experimental proposto por Taguchi. Os autores aplicam o método em um sistema de irrigação acionado por energia solar.

Box et al. (1998) apresentam várias ferramentas estatísticas existentes. Apresentam no contexto experimental, as desvantagens de se utilizar “um fator de cada vez”. Neste tipo de procedimento, os fatores ou parâmetros de projeto são estudados experimentalmente variando-se um fator de cada vez, enquanto os outros são mantidos constantes. Dessa forma, os autores apresentam conceitos e ferramentas estatísticas que avançam às respostas das questões acima, utilizando planejamentos experimentais e algumas de suas variações, efeitos de bloco e a aleatorização, método da superfície de resposta, variação da média, componentes da variância, influência do ambiente. O estudo dos fatores e suas interações podem proporcionar ao engenheiro de projeto um maior número de resultados com melhor qualidade e menor consumo de recursos, do que experimentos não planejados.

Os artigos mencionados representam uma amostra da importância em se utilizar ferramentas estatísticas em Engenharia. Porém, outras áreas do conhecimento humano como a Química, Engenharia de Alimentos e Bioengenharia já fazem um uso relativamente amplo de técnicas estatísticas para desenvolvimento e análise de seus problemas, demonstrando um esforço abrangente para avançar os conhecimentos.

2.2 Dinâmica de Rotores

O objetivo desta seção é demonstrar a evolução do problema dentro do grupo de trabalho do Laboratório de Máquinas Rotativas.

Recentemente, Cavalca et al. (2006), Castro e Cavalca (2006) e Okabe e Cavalca (2006) mostram que o avanço dos estudos de análises de sistemas mecânicos tais como rotores têm crescido significativamente desde 1970, principalmente devido ao desenvolvimento eletrônico e computacional. Elementos Finitos, matrizes de transferência, análise modal, juntamente com o processamento digital de sinais, softwares de aquisição e tratamento de dados têm sido desenvolvidos e aplicados intensamente à análise dinâmica dos sistemas rotor-mancal-fundação. O principal objetivo de uma análise experimental em rotores consiste em analisar o comportamento dinâmico do sistema real, com a finalidade de se conseguir maiores capacidades e rotações. Em análise modal, buscam-se as frequências naturais do sistema e seus respectivos modos próprios. Os métodos usados para isso são bastante diversos e com resultados amplamente

discutidos na literatura. Os testes experimentais normalmente são utilizados com o objetivo de verificação sobre as condições de projeto e/ou normas foram seguidas na concepção e construção do equipamento.

Hino et al. (1987) utilizou para a análise de vibrações de rotores, o método de subestruturas, no qual o sistema completo composto por: rotor, suportes, carcaça e fundação. O sistema composto por esses elementos é dividido em subestruturas, e as funções de transferência de cada parte são medidas através de teste de impacto, sendo os parâmetros modais extraídos através de procedimentos de ajuste de curvas. Com isso, as características modais obtidas de todas as subestruturas são sintetizadas através do método de síntese modal, obtendo desta forma as características dinâmicas de toda a estrutura. Os mancais hidrodinâmicos também são elementos importantes em sistemas mecânicos rotativos, sendo utilizados amplamente nesses sistemas para transmitir potência, tornando-se um aspecto crítico de qualquer sistema de eixos. Cheli e Cavalca (1992) apresentaram um trabalho experimental, no qual a influência da estrutura de suporte do tipo flexível sobre o comportamento dinâmico do rotor foi avaliada através de técnicas de identificação de parâmetros modais.

Rajalingham e Prabhu (1987) investigaram a influência da temperatura nas características físicas dos mancais hidrodinâmicos, no qual concluíram que a variação da temperatura do óleo na direção axial pode ser desprezada sem afetar a precisão dos resultados hidrodinâmicos para o filme de óleo. Cavalca et al. (1996) apresentam os resultados de uma abordagem matemática para modelagem de mancais hidrodinâmicos baseada na linearização da solução numérica da equação de Reynolds, visando obter os coeficientes diretos e acoplados de rigidez e amortecimento de mancais hidrodinâmicos.

Cavalcante e Cavalca (1998) apresentam estudos sobre o comportamento dinâmico da fundação e sua influência no sistema rotativo. Através de parâmetros modais (massa modal, fator de amortecimento e frequência natural) e da análise da função de transferência a partir da excitação harmônica unitária da estrutura completa (aplicada em um dos nós da fundação), é possível, a partir das funções de resposta em frequência, analisar a influência do comportamento dinâmico da fundação.

Assis e Steffen (2002) colocam em seu trabalho a dificuldade em se determinar parâmetros de mancais em máquinas rotativas. A determinação desses parâmetros varia muito em rotordinâmica, mas são necessários para a análise dinâmica do sistema em relação aos diferentes tipos de projeto. Os autores propõem uma metodologia geral baseada sobre otimização híbrida (algoritmos genéticos e *Simulated Annealing*) para identificar esses parâmetros em rotores flexíveis. No trabalho, técnicas de identificação baseadas na resposta ao desbalanceamento em rotores flexíveis são utilizadas para estimar parâmetros desconhecidos nos mancais. Os autores aplicaram um programa de otimização, composto de um método clássico, algoritmo genético e *simulated annealing*, no qual a função objetivo é otimizada através de algoritmos numéricos consagrados, que são entradas dos métodos posteriores.

Tapia (2003), desenvolveu um modelo através do método de elementos finitos, considerando as modelagens simplificadas de acoplamentos existentes na literatura. O trabalho surgiu, devido à escassa pesquisa existente sobre a modelagem eixo-mancais-acoplamentos. Nas simulações teóricas, verificou-se que o comportamento dinâmico do sistema é diferente, dependendo do modelo considerado, tanto em relação às frequências naturais, quanto às amplitudes de vibração. Foram obtidas através de experimentos as FRF's (Funções da Resposta em Frequência) por desbalanceamento e excitação aleatória através de um *shaker*. As curvas experimentais foram ajustadas através do Método de Mínimos Quadrados Não-Linear. A conclusão do trabalho mostrou que os melhores modelos dos acoplamentos são aqueles representados através dos parâmetros de rigidez e amortecimento.

2.3 Otimização Multi-Objetivos

Celi (1999) mostra as dificuldades de otimização multi-objetivo aplicada ao projeto de rotores de helicópteros, considerando-se a disponibilidade de dados experimentais e modelos teóricos que representem a real condição de trabalho do sistema. Neste trabalho o autor aborda técnicas de análise de sensibilidade, projeto robusto, efeitos de não-linearidades e diversas técnicas de otimização, incluindo algoritmos genéticos e *Simulated Annealing*.

Fusato e Celi (2002) apresentam um trabalho de simulação de movimento elástico das laminas da hélice de um helicóptero considerando flexão e torção. Um estudo de otimização de projeto multidisciplinar utilizando técnicas de otimização tradicionais através de métodos diferenciais foi executado. O trabalho mostra uma necessidade de redução das equações diferenciais para equações algébricas a fim de se conseguir a completa otimização. Deve-se, deste modo, verificar o impacto das linearidades sobre a precisão e eficiência computacional.

Deb (1999) estudou vários problemas relativos à aplicação de algoritmos genéticos multi-objetivos. Neste trabalho ele mostrou as dificuldades em convergir para a Fronteira Pareto-Ótima e as dificuldades em se manter a diversidade de soluções nesta fronteira, trabalhando com problemas multi-modais, enganosos, fronteiras convexas e não-convexas, funções descontínuas e fronteiras não uniformemente representadas. O autor sugere maiores estudos e desenvolvimento de novas técnicas englobando as mais diversas situações encontradas.

Fonseca e Fleming (1993) propuseram um método de determinação de ajuste baseado em ranking para MOGA (algoritmo genético multi-objetivos). Métodos convencionais de formação de nichos são introduzidos e uma teoria para determinação do tamanho do nicho foi apresentada. Os autores concluem que algoritmos genéticos, uma vez que procuram uma população de soluções parecem apropriados para otimização multi-objetivos. A habilidade de encontrar ótimos globais enquanto possível trabalhar com descontinuidades e funções ruídos tem motivado um aumento de uso na engenharia e campos correlatos.

Chung e Alonso (2004) estudaram a aplicação de algoritmos genéticos multi-objetivos no projeto de redução de estrondo em jatos executivos supersônicos. Os autores mostraram que os algoritmos genéticos multi-objetivos possuem uma melhor robustez em relação aos algoritmos baseados em gradientes, porém necessita de uma maior capacidade computacional. Os autores também propuseram um método estatístico (Kriging) a fim de acelerar o processo de otimização melhorando a eficiência dos algoritmos genéticos.

Sellar et al. (1996) descreveram que os métodos tradicionais de otimização possuem um custo alto quando necessitam analisar sistemas complexos, acoplados ou multi-modais, tornando-

se impraticáveis. Deste modo, novas técnicas de análise e métodos de otimização multidisciplinares estão sendo desenvolvidas. Os autores propuseram um método de otimização de subespaço simultâneo (CCSO), acoplado à metodologia de superfície de resposta e otimizado por redes neurais, aplicando este estudo no problema de redução de peso de aeronaves.

Chipperfield e Fleming (1995) descreveram o uso de algoritmos genéticos multi-objetivos no projeto de um sistema de controle multi-variável de uma turbina a gás. Foi mostrado como o algoritmo genético confere uma vantagem imediata sobre os métodos convencionais de otimização multi-objetivos através de soluções Ótimas de Pareto, permitindo ao engenheiro de controle examinar as diferentes soluções conflitantes dentro dos objetivos de projeto.

Kock, Mavris e Mistree (1998) mostraram que as funções polinomiais geradas através do MSR (Método de Superfície de Resposta) prevalecem como as mais empregadas e eficientes em análise e projetos em engenharia. Eles propuseram um método de análise multi-nível, onde o problema é modelado através de superfície de resposta particionada e analisaram um turboventilador comercial da Allison Engine Company.

Anderson (1999) mostrou que problemas em projetos freqüentemente consistem em uma mistura de soluções numéricas, cálculos analíticos e seleções em catálogos não sendo fácil o caminho de se utilizar equações diferenciais. Portanto, conclui que métodos de otimização não-gradiente são os mais apropriados para estes tipos de problemas e como a maioria dos problemas de otimização é multi-objetivos na sua própria natureza, algoritmos genéticos mostram-se adequados nestes casos.

Fleming e Purshouse (2001) mostraram que algoritmos genéticos têm sido usados em otimizações de estruturas e parâmetros de controle. Outras aplicações incluem diagnóstico de falhas, análise de estabilidade, planejamento de caminho de robôs e problemas combinatoriais (tais com planejamentos de negócios). O artigo descreve como a metodologia de algoritmos genéticos pode ser empregada em problemas de engenharia de controle.

Nos capítulos 3 e 4 outras referências bibliográficas são citadas. Isto facilita a compreensão dos métodos de otimização evolutivos multi-objetivos e introdução aos algoritmos genéticos, uma vez que são áreas de pesquisas recentes.

Capítulo 3

Otimização Simples e Multi-Objetivos

Neste capítulo é feita uma breve revisão sobre a teoria de planejamentos experimentais, metodologia de superfície de resposta e métodos de otimização que foram implementados no programa computacional. Uma maior ênfase é dada às metodologias de solução de problemas de otimização, mais especificamente entre as diferenças entre os métodos clássicos de otimização e os métodos evolutivos. Também é feita uma revisão sobre as diferenças entre otimização simples objetivo e multi-objetivos, complementando as técnicas recentemente aplicadas em estudos de sistemas, produtos e processos em Engenharia Mecânica.

3.1 DOE - Delineamento de Experimentos

O trabalho pioneiro de Fisher (1935) na Estação Agrícola Experimental de Rothamsted, Inglaterra, durante as décadas de 20 e 30 estabeleceu solidamente o papel da estatística no planejamento experimental e vice-versa, voltado para a teoria de erros, redução de dados, estimação de parâmetros, desenvolvendo a técnica de Análise da Variância (ANOVA) conforme Montgomery (1994). Steinberg e Hunter (1984) descreveram que, inicialmente, Fisher foi admitido para analisar alguns dados a partir de estudos conduzidos na Estação de Rothamsted na área de agricultura, mas em pouco tempo algumas questões importantes não podiam ser analisadas e concluídas, devido à falta de robustez de muitos dos planejamentos existentes neste

período. Em 1933, Fisher tornou-se professor da Universidade de Londres e, posteriormente, da Universidade de Cambridge, na qual conheceu o matemático Yates. Juntos, Fisher e Yates desenvolveram muito do que se conhece sobre a teoria de Planejamento de Experimentos, contribuindo para a área de Delineamento de Experimentos.

O trabalho de Fisher foi conduzido pela percepção de que a análise estatística de dados poderia ser fundamentada de forma mais precisa, através da aplicação de idéias que conduzissem a uma correta coleta de dados em primeiro lugar. Em 1980, Box G.E.P descreveu o trabalho de Fisher sobre planejamento ou projetos experimentais, e toda sua contribuição na área de experimentação: aleatorização, blocagem e utilização de réplicas; o desenvolvimento de novas classes de planejamentos experimentais e o trabalho conjunto com pesquisadores de diversas áreas com a aplicação das técnicas que ele desenvolvia. Assim, desde a introdução dos primeiros princípios de planejamento experimental, muito têm sido feito quanto à estatística aplicada nas áreas de agricultura, química, biologia, e em diversas áreas da engenharia.

3.1.1 Considerações sobre projeto e planejamento robusto

Em estatística, o termo “robusto” ou “robustez” foi proposto por Box, G.E.P em 1953. O termo robustez está relacionado à aplicação de algum procedimento que fornece resultados satisfatórios, dentro do contexto de análise, mesmo havendo violações em algumas hipóteses pré-estabelecidas.

Seguindo a linha de pesquisa iniciada por Pearson (1931), Box em 1953 examinou os efeitos da análise da variância com o teste inicial de normalidade de Bartlett, considerando uma hipótese que englobava ambos os procedimentos. A pesquisa inicial desenvolvida por Pearson indicou que a análise da variância é robusta para hipóteses que consideram o teste de normalidade, mas também sugeriu que estas conclusões não deveriam ser estendidas para as comparações de estimativas da variância baseada em amostras independentes. Box, ainda em 1953, concluiu que certamente o teste de Bartlett é significativamente sensível ao teste de normalidade. Este resultado o conduziu a observar que o teste de Bartlett poderia ser utilizado como um teste preliminar para a análise da variância, uma prática recomendada por alguns

estatísticos, com a ressalva de que o problema não é conhecido previamente, sendo necessária aplicações dessas técnicas, para depois obterem-se conclusões, e repetir esse procedimento, quando não há respostas conclusivas. A análise de técnicas tradicionais da estatística para a determinação de sensibilidade por hipóteses e o desenvolvimento de novas técnicas menos sensíveis a diversas condições, têm sido o foco da pesquisa em estatística aplicada nas últimas décadas. O planejamento experimental é uma área na qual se concentram as questões de investigação da robustez, pois as suposições dos pesquisadores têm uma influência extremamente significativa na elaboração do projeto experimental. Entretanto, o planejamento deve ser escolhido anteriormente à coleta de dados, e estes não podem ser descartados se as suposições estiverem erradas. Assim, o interesse por planejamentos que forneçam respostas robustas pode incorporar variações não previstas, e ainda assim gerar respostas aceitáveis. A hipótese que se destaca em vários trabalhos de pesquisa em planejamento experimental está associada à equação:

$$\text{Resposta} = f(x) + e \quad (3.1)$$

O modelo representa o efeito das variáveis previstas sobre a resposta e os erros descritos de forma geral a partir deste modelo. Os pesquisadores freqüentemente utilizam modelos provenientes de experiências anteriores, cada um baseado em considerações teóricas, ou então utilizam um modelo empírico mais simples, considerando que este se adequará ao estudo. É improvável, entretanto, que o pesquisador esteja completamente certo em qualquer tentativa empreendida para se obter um modelo adequado, e a estratégia de planejamento que falhar, levando em conta essa incerteza, deve ser vista com certo ceticismo. Particularmente, são planejamentos obtidos com critérios de ótimos (planejamentos ótimos), pois são significativamente dependentes do modelo escolhido. Esses planejamentos tendem a concentrar todos os experimentos realizados sobre um número pequeno de pontos experimentais, e são normalmente adequados para a estimativa dos coeficientes do modelo assumido, que podem fornecer insuficiente ou nenhuma verificação para a falta de ajuste. Hipóteses sobre a componente erro são caracterizadas, normalmente, em termos de uma distribuição de probabilidade, também sujeita às incertezas.

Neste trabalho, adota-se o delineamento de experimentos para definir os testes a serem executados para posterior aplicação de metodologia de superfície de resposta (MSR) e, ainda posteriormente, a aplicação de otimização multi-objetivos. O delineamento de experimentos levará em conta apenas a definição dos parâmetros de estudo, não sendo feita análise estatística dos resultados por este método, uma vez que serão utilizados diretamente na otimização multi-objetivos.

3.2 Metodologia de Superfície de Resposta – MSR

Metodologia de Superfície de Resposta (MSR) de acordo com Myers e Montgomery (1995) é uma coleção de técnicas estatísticas e matemáticas úteis nos processos de otimização, melhoria e desenvolvimento e possui também importantes aplicações em projetos, desenvolvimento e formulação de novos produtos, também como na melhoria dos produtos atuais.

As mais extensivas aplicações do MSR estão na área industrial, particularmente em situações onde várias variáveis de entrada potencialmente influenciam as medidas de performance ou qualidade de um determinado serviço ou produto. Esta medida de performance ou qualidade é chamada de resposta e é tipicamente uma variável contínua, embora existam casos de variáveis discretas. Na maioria das vezes o sistema envolve mais de uma resposta e, deste modo, passa a ser um problema multi-objetivos.

MSR pode tomar respostas de funções desconhecidas e aproximá-las através de variáveis codificadas, onde estas variáveis codificadas geralmente são adimensionais com amostragem de média zero e de mesma dispersão ou mesmo desvio padrão. Usualmente um polinômio de baixa ordem é gerado em alguma região relativamente pequena do espaço de variáveis independentes. Estes polinômios são utilizados para gerar superfícies onde é possível a visualização das curvas de nível e, através de técnicas estatísticas, retiram-se, então, os parâmetros do modelo.

A criação de tais modelos de superfície de resposta é particularmente apropriada nos estágios preliminares de projetos onde co-existam variáveis conflitantes e se necessita de uma análise rápida de performance, tanto técnica como econômica. Em muitos casos, um modelo de

primeira ou segunda ordem é usado. O modelo de primeira ordem para o caso de duas variáveis independentes é:

$$\eta_L = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 \quad (3.2)$$

E no caso de levar em consideração a interação entre as duas variáveis independentes, esta interação pode ser adicionada através de um novo termo como segue:

$$\eta_L = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_{12} x_1 x_2 \quad (3.3)$$

Freqüentemente, a curvatura dentro de uma superfície de resposta é suficientemente acentuada de modo que os modelos de primeira ordem são inadequados, assim, um modelo de segunda ordem será necessário nestas situações. Para o caso de duas variáveis tem-se:

$$\eta_Q = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_1 x_1^2 + \beta_2 x_2^2 + \beta_{12} x_1 x_2 \quad (3.4)$$

3.3 Otimização

Otimização refere-se a achar uma ou mais soluções viáveis que correspondem aos valores extremos de uma ou mais funções objetivos. Quando um problema de otimização de um sistema físico envolve somente uma função objetivo, chamamos este processo de otimização simples-objetivo. Quando um problema de otimização envolve mais de uma função objetivo, o procedimento de busca de uma ou mais soluções é conhecido como otimização multi-objetivos segundo Deb (2001).

Existem vários métodos para se tratar otimizações multi-objetivos. Um destes métodos é converter o problema multi-objetivo em um problema simples objetivo. Técnicas de ponderação (Weighted Sum) e restrições (e-Constraint) são exemplos destes métodos. Os métodos acima citados utilizam parâmetros pré-definidos, devendo-se ter uma avaliação prévia do sistema em estudo a fim de se conseguir parâmetros que realmente representem este sistema. Outro processo de solução de problemas multi-objetivos é a utilização de algoritmos evolutivos.

3.3.1 Otimização Simples e Multi-Objetivos

A necessidade de encontrar soluções ótimas em um problema vem principalmente da necessidade de se obter uma solução para minimizar custos de fabricação ou maximizar a confiabilidade, entre outros.

Por causa das propriedades destas soluções otimizadas, métodos de otimização são de grande importância na prática, particularmente em projetos de engenharia, experimentos científicos e tomadas de decisão em negócios.

Os maiores esforços nesta área de pesquisa tem se concentrado em entender, desenvolver e aplicar métodos de otimização simples-objetivo. Nos dias atuais, entretanto, existem algoritmos de otimização simples-objetivo que trabalham usando técnicas baseadas em gradientes e técnicas de busca heurística. Além de princípios de busca determinísticos envolvidos num algoritmo, existem também princípios estocásticos, os quais permitem aos algoritmos de otimização encontrar soluções globais mais confiáveis. A fim de ampliar a aplicação de algoritmos de otimização em vários domínios diferentes, princípios naturais e físicos são simulados para desenvolver algoritmos de otimização robustos. Algoritmos Evolutivos e Simulated Annealing são dois exemplos de tais algoritmos.

A maioria dos problemas reais de otimização naturalmente envolve múltiplos objetivos. O princípio extremista mencionado acima não pode ser aplicado para um único objetivo quando os demais objetivos também são importantes. Diferentes soluções podem produzir discrepâncias (resultados conflitantes) entre diferentes objetivos. Uma solução que é extrema (no melhor sentido) com respeito a um objetivo requer um compromisso com outros objetivos. Isto impede a escolha de uma solução que é ótima com respeito a somente um objetivo.

Como no caso de otimização simples-objetivo, otimização multi-objetivos tem sido estudada extensivamente. Existem muitos algoritmos e estudos de casos de aplicações envolvendo múltiplos objetivos. Entretanto, existe um ponto comum na maioria destes estudos,

pois a maioria destes métodos evita a complexidade envolvida em um problema de otimização multi-objetivos verdadeiro e transformam múltiplos objetivos em uma função simples-objetivo através de parâmetros definidos. Assim, a maioria dos estudos em otimização multi-objetivos clássica não trata otimização multi-objetivos diferentemente de otimização simples-objetivo. De fato, otimização multi-objetivos é considerada uma aplicação de otimização simples objetivo para manusear múltiplos objetivos. Isto é contrário à nossa intuição de que otimização simples-objetivo é um caso específico de otimização multi-objetivos e, deste modo, otimização multi-objetivos não é uma extensão de simples-objetivo.

É verdade que as teorias e algoritmos por otimização simples-objetivo podem ser aplicados a estas funções simples-objetivos transformadas de multi-objetivos. Entretanto, existe uma diferença fundamental entre otimização simples e multi-objetivos que é ignorada quando usando métodos de transformação.

3.3.2 Diferenças Fundamentais entre Simples-Objetivo e Multi-Objetivos

Sem perda de generalidade, vamos discutir as diferenças fundamentais entre otimização simples-objetivo e multi-objetivos tomando como exemplo um problema de otimização de dois objetivos. Para dois objetivos conflitantes, cada objetivo corresponde a uma solução ótima diferente. Tomemos um exemplo de compra de um automóvel como mostrado na figura 3.1 onde existem 5 opções de compra que variam entre duas condições, respectivamente o conforto e o custo. Se o comprador pretende sacrificar o custo mínimo em troca de conforto, este comprador provavelmente não adotará a solução 3 e procurará uma alternativa que esteja de acordo com sua capacidade financeira, porém que atende ao esperado padrão de conforto. Assim, nós podemos visualizar uma série de possíveis soluções entre 3 e 4 (soluções 1, 2 e 5), onde o ganho em um objetivo leva ao sacrifício do outro objetivo. As soluções 3 e 4 são consideradas ótimas do ponto de vista que nenhuma das duas é melhor que a outra em ambos os sentidos (conforto e custo). Dizemos que estas soluções são conflitantes.

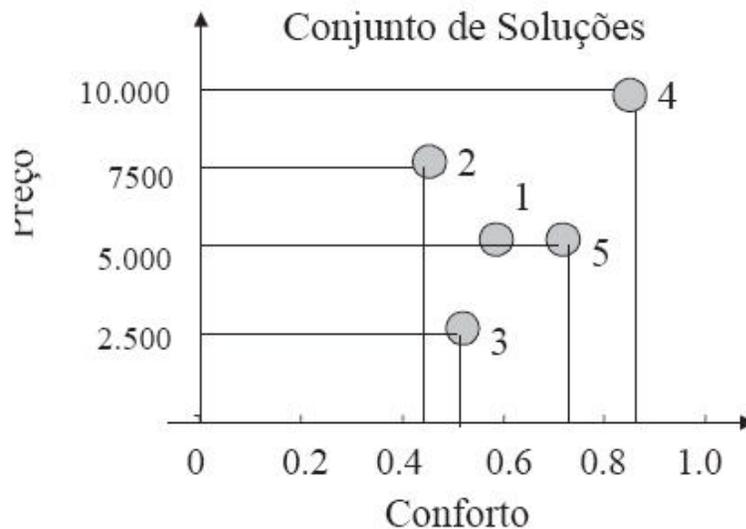


Figura 3.1: Exemplo que ilustra várias opções de compra de carro

Agora vem a grande questão. Com todas estas soluções conflitantes em mente, qual é a melhor solução entre todas com respeito aos dois objetivos? A ironia da questão é que nenhuma delas é melhor que a outra em ambos os objetivos. A razão está no fato de que nenhuma solução desta série faz ambos os objetivos (custo e conforto) parecerem melhores do que outra solução da série. Assim, em problemas com mais do que um objetivo conflitante, não existe solução ótima simples. O que existe na verdade são soluções que são consideradas ótimas (no sentido de serem as melhores possíveis). Sem qualquer informação prévia, nenhuma solução desta série de soluções ótimas pode ser dita melhor do que as outras. Esta é a diferença fundamental entre otimização simples-objetivo e otimização multi-objetivos.

3.3.3 Otimização Multi-Objetivos – Duas abordagens

Embora a diferença fundamental entre estas duas otimizações permaneça na série ótima de soluções, na prática geralmente precisamos apenas de uma solução, não importando se o problema é simples ou multi-objetivos. No caso de multi-objetivos o usuário agora está em um dilema. Qual destas soluções deverá ser escolhida? Tomando-se novamente o exemplo do carro, outros parâmetros são necessários para definir a compra (financiamento, combustível, cor, aparência, etc). Frequentemente, tais informações de alto nível são subjetivas, qualitativas e

movidas pela experiência. Assim, em uma otimização multi-objetivos, idealmente os esforços devem ser direcionados para se achar as soluções ótimas conflitantes tomando-se todos os objetivos importantes. Depois de encontradas as soluções, pode-se usar outras informações para definir qual a decisão escolhida. Tendo em vista estas discussões, sugere-se, portanto, o seguinte princípio para um procedimento de otimização multi-objetivos ideal:

Passo 1 Encontrar soluções ótimas múltiplas conflitantes com uma larga faixa de valores para os objetivos

Passo 2 Escolher uma das soluções obtidas usando informação de alto nível.

A figura 3.2 mostra esquematicamente o procedimento de uma otimização multi-objetivos ideal. No passo 1 as soluções múltiplas conflitantes são encontradas. No passo 2, informação de alto nível é usada para escolher uma das soluções conflitantes. Com este procedimento em mente, é fácil perceber que otimização simples-objetivo é um caso específico de otimização multi-objetivos, como foi dito anteriormente. No caso de otimização multi-objetivos com somente uma solução global, o passo 1 encontrará somente uma solução, não necessitando, portanto, passar pelo passo 2. No caso de otimização simples-objetivo com múltiplos ótimos globais, ambos os passos são necessários para achar todos ou muitos dos ótimos globais e depois escolher um entre eles usando informação de alto nível sobre o problema.

Se pensarmos com cuidado, cada solução conflitante corresponde a uma ordem de importância dos objetivos. Voltando ao exemplo do carro, está claro da figura 3.1 que a solução 3 define mais importância para o custo do que para o conforto. Por um outro lado, a solução 4 mostra mais importância para o conforto. Assim, se cada fator de preferência relativo entre os objetivos é conhecido para um dado problema específico, não existe a necessidade de seguir o princípio acima para resolver um problema de otimização multi-objetivos.

O método de transformar um problema multi-objetivos em um simples-objetivo pode ser obtido através de uma soma ponderada dos objetivos, onde o peso de cada objetivo será proporcional a sua importância.

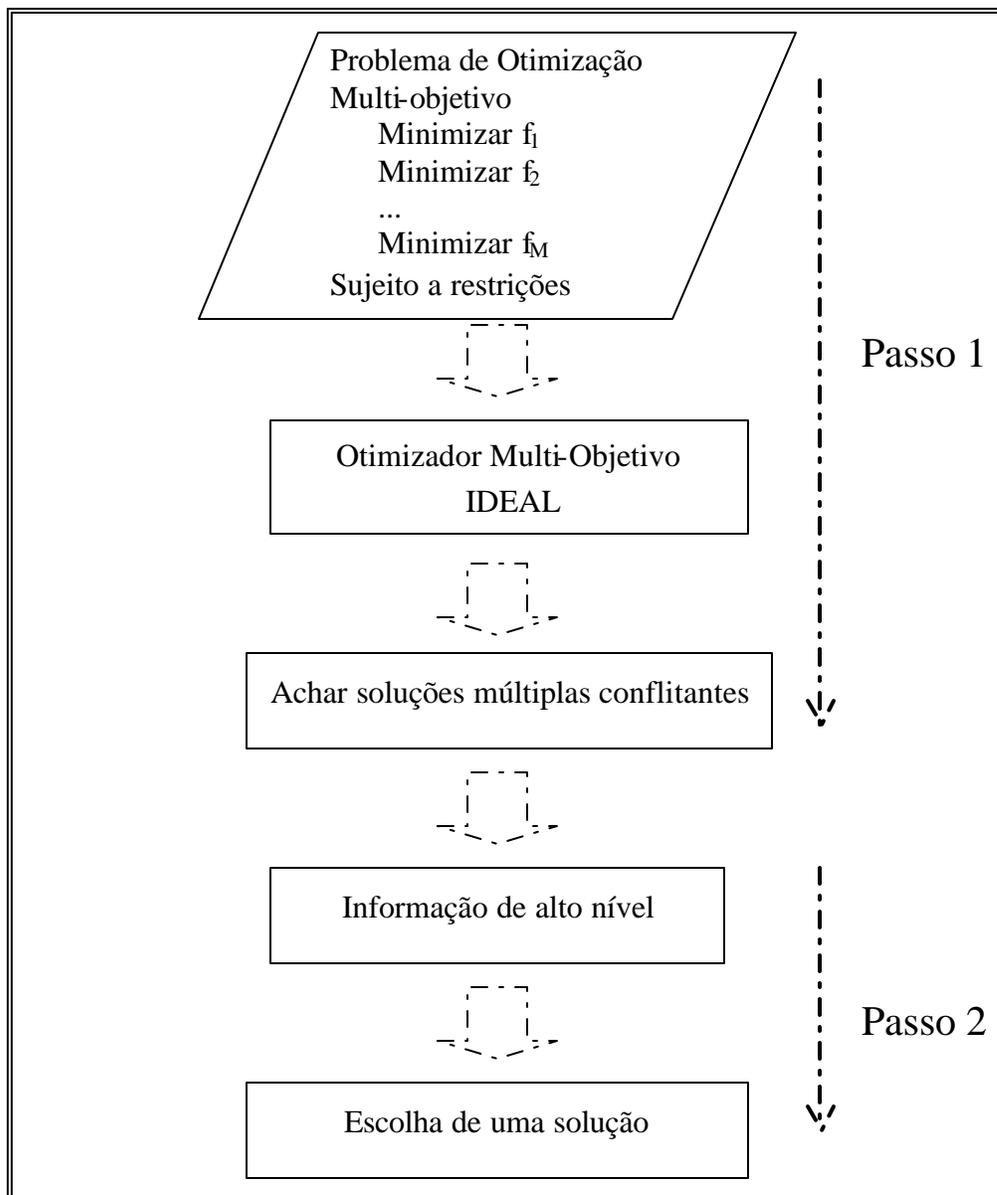


Fig. 3.2 – Fluxo de Otimização Multi-Objetivos Ideal

Este método de escalonar um vetor objetivo dentro de uma função objetivo composta converte o problema de otimização multi-objetivos para um problema de otimização simples-objetivo. Quando esta função objetivo composta é otimizada, na maioria dos casos é possível obter uma solução particular. Este procedimento de manusear problemas de otimização multi-objetivos é muito simples, porém ainda mais subjetivo do que o procedimento ideal descrito anteriormente. Denomina-se este procedimento de “otimização multi-objetivo baseado em preferência” (preference-based). Um esquema do procedimento é mostrado na figura 3.3.

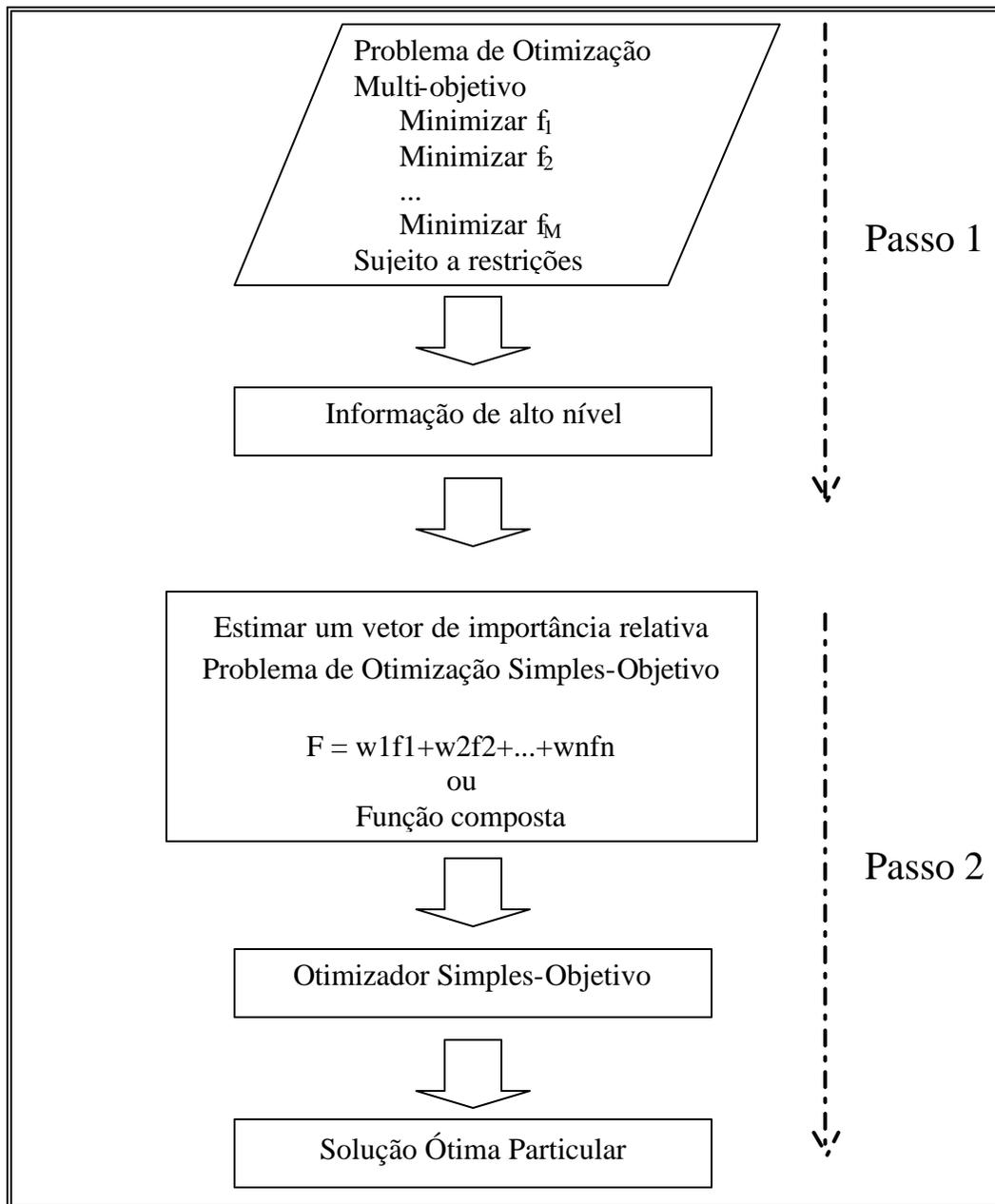


Fig. 3.3 – Fluxo de Otimização Multi-Objetivos Ponderada

É importante perceber que uma solução conflitante obtida usando estratégia baseada em preferência é fortemente sensível ao vetor de preferência utilizado na formação da função objetivo composta. Uma mudança neste vetor irá resultar em uma solução ótima diferente e nem todos os vetores geram soluções realmente ótimas. Além destas dificuldades, achar um vetor que seja adequado é altamente subjetivo e indireto, ou seja, requer uma análise prévia baseada em

dados de informações de alto nível (a priori). Portanto, sem o conhecimento do comportamento das soluções ótimas torna-se muito difícil definir um vetor de preferência adequado. Muitos métodos clássicos de otimização multi-objetivos convertem múltiplos objetivos em simples objetivo usando um vetor de preferência. Porém, a menos que um vetor de preferência confiável e preciso seja escolhido, a solução particular obtida terá caráter duvidoso quanto à verdadeira otimização.

O método de otimização multi-objetivos ideal mostrado na figura 3.2 é menos subjetivo. O usuário não necessita ter informações a priori para definir um vetor de preferência. O trabalho está em encontrar tantas soluções conflitantes quantas sejam possíveis. Uma vez que encontrada uma série bem distribuída de soluções conflitantes, necessita-se de algumas informações para definir a melhor escolha. Deste modo utilizam-se informações de alto nível para se comparar soluções ótimas conflitantes previamente definidas. Assim, no modelo ideal, as informações não são usadas para se procurar uma nova solução, ao invés disto, são usadas para escolher uma solução de uma série de soluções já obtidas. Deste modo, existe uma diferença fundamental na utilização da informação de alto nível entre ambos os métodos. No método baseado em preferência, um vetor de preferência necessita ser fornecido sem conhecimento de suas conseqüências. No método ideal, a informação é usada para se escolher uma solução entre as soluções ótimas obtidas. Assim, conclui-se, portanto, que o método ideal é mais robusto, mais prático e menos subjetivo.

3.4 Métodos Evolutivos

Foi mencionado anteriormente que o caminho clássico para resolver otimização multi-objetivos é seguir a técnica baseada em preferência. Entretanto, o campo de procura e otimização tem mudado nos últimos anos pela introdução de algoritmos de otimização multi-objetivos não-clássicos, não-ortodoxos e não-estocásticos. Destes, os algoritmos evolutivos traduzem os princípios evolutivos da natureza para direcionar as procuras em direção a soluções ótimas. Uma das mais fortes diferenças entre os algoritmos clássicos e os evolutivos é que estes utilizam uma população de soluções em cada iteração, ao invés de uma solução. E, uma vez que uma população de soluções é processada em cada iteração, a resposta do algoritmo evolutivo é

também uma população de soluções. Se o problema a ser otimizado tiver apenas uma solução ótima, todas as soluções encontradas pelo algoritmo evolutivo podem convergir para esta única solução ótima. Entretanto, se um problema de otimização tem múltiplas soluções ótimas, um algoritmo evolutivo certamente apresenta potencial para capturar tais múltiplas soluções ótimas na sua população final.

Esta habilidade dos algoritmos evolutivos de encontrar múltiplas soluções ótimas em uma simples simulação faz este algoritmo particularmente interessante no sentido de resolver problemas de otimização multi-objetivos.

3.5 Otimização Multi-Objetivos - Formulação

Como o nome sugere, um problema de otimização multi-objetivos trabalha com mais de uma função objetivo, as quais devem ser maximizadas ou minimizadas. Existem também restrições que qualquer solução viável deve satisfazer. A forma geral para um problema de otimização multi-objetivos é a seguinte:

$$\begin{array}{lll}
 \textit{maximizar/minimizar} & f_m(\mathbf{x}), & m = 1, 2, \dots, M \\
 \textit{sujeito a} & g_j(\mathbf{x}) \leq 0, & j = 1, 2, \dots, J; \\
 & h_k(\mathbf{x}) = 0, & k = 1, 2, \dots, K; \\
 & x_i^{(L)} \leq x_i \leq x_i^{(U)}, & i = 1, 2, \dots, n.
 \end{array} \quad (3.5)$$

onde \mathbf{x} é o vetor de n variáveis de decisão $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$. Os valores $x_i^{(L)}$ e $x_i^{(U)}$ representam os valores mínimo e máximo respectivamente para a variável x_i . Estes limites definem o espaço de variáveis de decisão. O vetor \mathbf{x}_i representa as soluções e as funções de restrição são as J desigualdades (g_j) e as K igualdades (h_k).

Uma solução \mathbf{x} viável será aquela que satisfaça as $J+K$ funções de restrição e os $2n$ limites. Caso contrário, a solução não será viável. O conjunto de todas as soluções viáveis forma o espaço de busca \mathbf{S} . Cada uma das m funções objetivo $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})$ pode ser maximizada ou minimizada. Porém, para trabalhar com os algoritmos de otimização, é necessário convertê-las de

forma que sejam todas maximizadas ou minimizadas. Esta é uma diferença fundamental em relação à otimização simples objetivo, cujo espaço de objetivos é unidimensional.

O mapeamento acontece então entre um vetor \mathbf{x} (n -dimensional) e um vetor $f(\mathbf{x})$ (M -dimensional). Por exemplo, se cada elemento de \mathbf{x} e $f(\mathbf{x})$ são números reais, então $f(\mathbf{x})$ estaria mapeada como $f(\mathbf{x}): \mathbb{R}^n \rightarrow \mathbb{R}^M$.

3.6 Dominância e Soluções Pareto-Ótimas

Como foi mostrado anteriormente, tomar decisões implica um processo que consiste em vários fatores, com o objetivo de encontrar a melhor solução. Em alguns casos, podem aparecer várias soluções boas, das quais nenhuma é melhor que a outra. A Figura 3.1 mostrou o exemplo de comprar um carro em que se está procurando a melhor opção considerando o preço e o conforto simultaneamente. Existe então uma “relação conflitante” entre os objetivos. Quanto maior o conforto, maior o preço e vice-versa.

Diz-se que uma solução domina uma outra se seus valores são melhores em todos os objetivos. Por exemplo, a solução 5 domina a solução 1. E a solução 5 é não-dominada por nenhuma outra. O mesmo acontece com as soluções 3 e 4. Se não se conhece a priori a importância relativa de cada objetivo, pode-se dizer que as soluções 3, 4 e 5 são igualmente boas. Portanto, existe um conjunto de soluções ótimas, este conjunto é chamado de soluções não-dominadas 3, 4 e 5. As outras soluções (1 e 2) formam o conjunto dominado. Estes conjuntos têm as seguintes propriedades:

1. Qualquer par de soluções não-dominadas deve ser não-dominadas uma em relação à outra.
2. Quaisquer das soluções não contidas no conjunto não-dominado, devem ser dominadas por no mínimo uma solução do conjunto não-dominado.

Pode-se desenhar uma curva das soluções não-dominadas que estão em um espaço contínuo. Esta curva representa a Fronteira de Pareto.

3.7 Objetivos em Otimização Multi-Objetivos

Quando a informação adicional sobre importância dos objetivos é desconhecida, todas as soluções Pareto-Ótimas são igualmente importantes. Deb (2001) assinala dois importantes objetivos em Otimização Multi-Objetivos:

1. Encontrar um conjunto de soluções o mais próximo possível da Fronteira de Pareto.
2. Encontrar um conjunto de soluções com a maior diversidade possível.

A primeira meta é comum para qualquer processo de otimização. Soluções muito distantes da Fronteira de Pareto não são desejáveis. Porém, encontrar a maior diversidade dentro das soluções é uma meta específica para Otimização Multi-Objetivos.

A Figura 3.4a mostra-se uma boa distribuição de soluções na Fronteira de Pareto. No entanto, na Figura 3.4b as soluções estão distribuídas apenas em algumas regiões. Torna-se necessário assegurar a maior abrangência possível da Fronteira Pareto-Ótima. Normalmente, uma boa diversidade no espaço de decisão garante também a diversidade no espaço de procura, porém em alguns problemas isto não acontece.

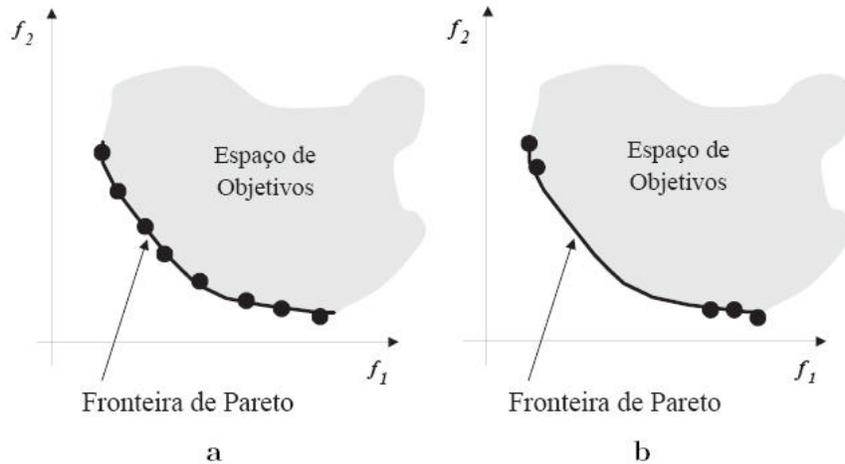


Fig. 3.4 – Fronteira Pareto-Ótima e Diversidade de Soluções

3.8 Conceito de Dominância de Pareto

Vamos examinar mais precisamente os conceitos de dominância.. Se existem M funções objetivo $f_j, j = 1, \dots, M$, o operador \prec entre duas soluções, $\mathbf{x} \prec \mathbf{y}$, significa que a solução \mathbf{x} é melhor que \mathbf{y} em um objetivo em particular.

Reciprocamente, $\mathbf{x} \succ \mathbf{y}$ denota que a solução \mathbf{x} é pior que \mathbf{y} para algum objetivo. Este operador é usado na definição a seguir conforme Deb (2001).

3.8.1 Definição 1

Uma solução $\mathbf{x}^{(1)}$ domina uma outra solução $\mathbf{x}^{(2)}$ (representado como $\mathbf{x}^{(1)} \preceq \mathbf{x}^{(2)}$) se as condições seguintes são satisfeitas :

1. A solução $\mathbf{x}^{(1)}$ não é pior que $\mathbf{x}^{(2)}$ em todos os objetivos, ou seja, $f_j(\mathbf{x}^{(1)}) \not\prec f_j(\mathbf{x}^{(2)})$ para todo $j = 1, 2, \dots, M$.
2. A solução $\mathbf{x}^{(1)}$ é estritamente melhor que $\mathbf{x}^{(2)}$ pelo menos em um objetivo, ou seja $f_j(\mathbf{x}^{(1)}) \prec f_j(\mathbf{x}^{(2)})$ pelo menos em um $j = 1, 2, \dots, M$.

Se ambas as condições são satisfeitas, pode-se dizer que:

1. $\mathbf{x}^{(2)}$ é dominada por $\mathbf{x}^{(1)}$
2. $\mathbf{x}^{(1)}$ é não dominada por $\mathbf{x}^{(2)}$
3. $\mathbf{x}^{(1)}$ é não inferior a $\mathbf{x}^{(2)}$

Na Figura 3.1, a solução 5 domina a solução 1 ($5 \preceq 1$), e a solução 3 domina a solução 2 ($3 \preceq 2$). Portanto o conceito de dominância permite comparar soluções com múltiplos objetivos.

3.8.2 Propriedades da Relação de Dominância

A relação de dominância satisfaz seguintes propriedades:

1. Não é reflexiva. Conforme a definição 2 uma solução não pode ser dominada por si mesma.
2. Não é simétrica, porque se $\mathbf{p} \preceq \mathbf{q}$ não implica que $\mathbf{q} \preceq \mathbf{p}$.
3. É transitiva, dado que se $\mathbf{p} \preceq \mathbf{q}$ e $\mathbf{q} \preceq \mathbf{r}$ então $\mathbf{p} \preceq \mathbf{r}$.

Estas propriedades caracterizam a relação de dominância como uma relação de ordem parcial estrita.

3.8.3 Otimalidade de Pareto

Quando o conjunto de soluções é finito, é possível fazer comparação das soluções duas a duas e pode-se dividir o conjunto em soluções dominadas e não dominadas. Portanto, tem-se um conjunto não dominado e um conjunto dominado.

3.8.4 Definição 2

Dado conjunto de soluções P , o conjunto não-dominado P^0 é formado por aquelas soluções que são não-dominadas por qualquer elemento de P .

Quando o conjunto P é o espaço completo de busca, o conjunto não dominado P^0 é chamado de conjunto Ótimo de Pareto. A Figura 3.5 mostra vários exemplos de conjuntos Ótimos de Pareto, conforme várias combinações de objetivos para as funções f_1 e f_2 . A curva indica onde o conjunto está localizado. Então, é possível ter conjuntos Pareto-Ótimos formando uma região contínua ou formada pela união de regiões descontínuas.

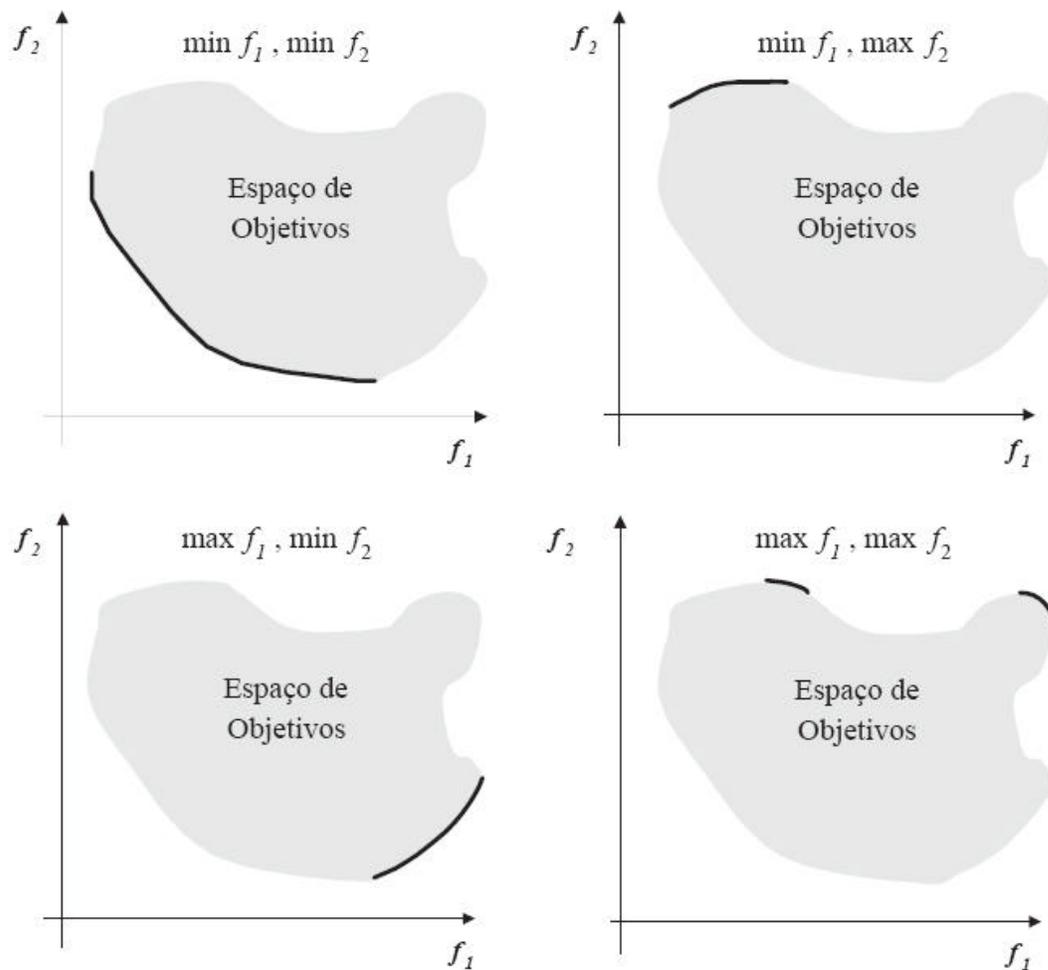


Fig. 3.5 – Soluções Pareto-Ótimas e Espaços de Objetivos

3.8.5 Definição 3

Se cada elemento x do conjunto P não é dominado por alguma solução y na vizinhança de x tal que $\|y - x\|_8 = e$, onde e é um número positivo arbitrariamente pequeno, então o conjunto P é chamado de conjunto Pareto-Ótimo local.

A Figura 3.6 mostra dois conjuntos Pareto-Ótimos locais que são não-dominados, mostrando a sua vizinhança no seu espaço de objetivos e no espaço de variáveis (à direita).

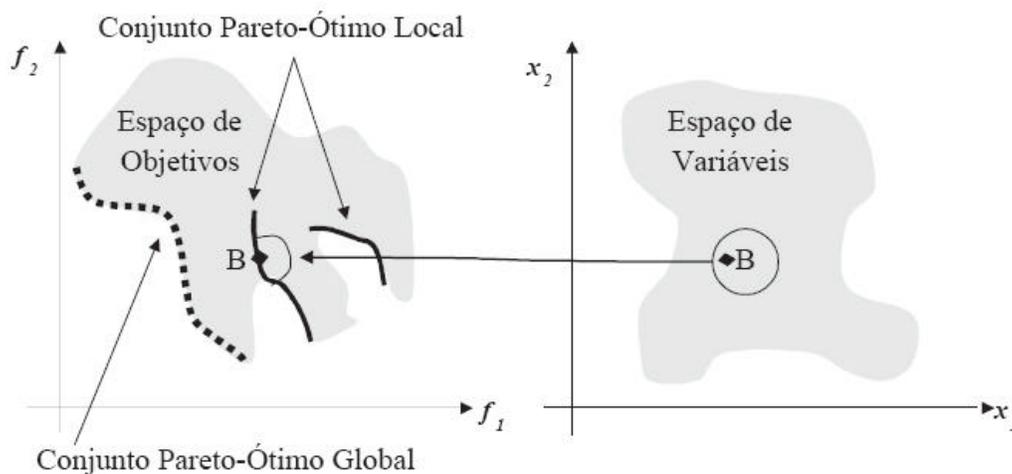


Fig. 3.6 – Soluções Pareto-Ótimas Globais e Locais

3.9 Comentários Finais

Neste capítulo foram mostradas noções de DOE (Delineamento de Experimentos), MSR (Metodologia de Superfície de Resposta) e Problemas de Otimização Multi-objetivos. Apresentou-se o modelo geral e as principais diferenças entre os problemas de otimização de simples objetivo e multi-objetivos. Finalmente, foram descritas as definições de dominância de Pareto e considerações importantes sobre as características e objetivos na otimização multi-objetivos. No próximo capítulo serão apresentados os Algoritmos enfatizando sua utilidade para encontrar um conjunto de soluções de Pareto-Ótimas simultaneamente.

Capítulo 4

Algoritmos Evolutivos para Otimização Multi-Objetivos

Os algoritmos evolutivos simulam os princípios naturais evolutivos com a finalidade de criar procedimentos de procura e otimização. Visto que nos últimos anos tem-se observado uma expansão na utilização destes algoritmos evolutivos, em especial os algoritmos genéticos, enfocaremos neste capítulo a aplicação destes em problemas de otimização multi-objetivos, descrevendo diferentes tipos de modelos atuais, bem como caracterizando vantagens e desvantagens entre eles e em relação aos métodos tradicionais ou clássicos de otimização.

Como discutido no capítulo anterior, existem dois objetivos em algoritmos de otimização multi-objetivos, onde um deles é achar soluções largamente dispersas dentro do espaço de busca a fim de se obter uma fronteira Pareto-Ótima. Achar e manter múltiplas soluções em uma rodada simples do programa é uma característica única das técnicas de otimização evolutivas. Os tópicos tratados neste capítulo abordam os conceitos básicos para entender o funcionamento dos algoritmos genéticos, vantagens e desvantagens sobre as técnicas tradicionais de otimização e os procedimentos de otimização multi-objetivos.

4.1 Algoritmos Genéticos

O conceito de algoritmo genético foi primeiramente apresentado por Holland (1975) e, como o nome sugere, os algoritmos genéticos são técnicas de busca inspiradas em mecanismos de seleção e evolução genética. Dada uma população inicial de soluções, esta evolui até convergir para uma solução ou série de soluções, por meio da aplicação de operadores genéticos de seleção, cruzamento, mutação e elitismo.

Considerando um problema de otimização qualquer, os algoritmos genéticos iniciam a busca da melhor solução a partir de um conjunto inicial de soluções, geralmente aleatórias. Denominamos cada solução do conjunto inicial como indivíduo ou cromossomo. Um indivíduo pode ser representado por uma cadeia de símbolos, por exemplo, uma cadeia binária ou números reais. É importante que cada indivíduo da população seja capaz de representar completamente uma possível solução do problema tratado. Em seguida, uma nova população ou geração é gerada a partir da população inicial.

Para criar os indivíduos da nova população, são utilizados operadores genéticos de seleção, elitismo, cruzamento e mutação. Uma população é obtida a partir da anterior, aplicando-se a seleção para aqueles indivíduos com um maior valor de aptidão, simulando o processo de seleção natural.

No caso de uma otimização simples objetivo, o valor de aptidão é calculado para cada indivíduo mediante uma função chamada aptidão ou objetivo. Para o caso de otimização multi-objetivos, teremos um processo similar, porém a técnica mais usada é fazer uma ordenação por meio de ranking. O processo de geração de novas populações é repetido iterativamente até que o algoritmo genético chegue a uma solução ou série de soluções viáveis, ou satisfaça alguma condição de parada. A representação do algoritmo genético na forma de fluxograma está ilustrada na Figura 4.1.

4.2 Representação das Soluções

Cada indivíduo de uma população, gerada pelo algoritmo genético, representa uma possível solução do problema que se deseja resolver. Assim, as variáveis da função objetivo devem ser representadas por um indivíduo.



Figura 4.1 – Fluxograma de Algoritmo Genético

Uma possível representação para as variáveis seria uma cadeia de bits de tamanho n que representa os indivíduos. Outra possibilidade seria um vetor de valores contínuos ou verdadeiros. Várias representações como matrizes e árvores também são possíveis. Escolher uma representação adequada para a solução de um problema é uma área de estudo fundamental para os algoritmos genéticos.

4.2.1 Representação Binária

A representação dos indivíduos através de uma cadeia de caracteres binários é uma das mais utilizadas nos algoritmos genéticos. Definindo-se os limites inferior e superior, pode-se fazer uma representação discreta através de uma seqüência binária. Cada variável é representada por uma parte do cromossomo sendo chamada de alelo ou gene. Assim se tivermos três variáveis de tamanho 8 bits, teremos um cromossomo de 24 bits.

8 bits	8 bits	8 bits	
01001001	11010110	00111101	= total de 24 bits
Variável 1	Variável 2	Variável 3	

Figura 4.2 – Representação Binária

O procedimento de mapeamento acima descrito caracteriza as variáveis de decisão da seguinte forma:

1. Qualquer precisão arbitrária (embora finita) pode ser alcançada através do uso de uma cadeia de caracteres suficientemente longa.
2. Diferentes variáveis de decisão podem ter diferentes precisões simplesmente alterando o comprimento da cadeia de caracteres.
3. Os valores podem ser negativos ou positivos, dependendo apenas dos limites inferior e superior.

4.2.2 Representação Contínua ou Verdadeira

Para casos onde o erro ocasionado pela representação binária é crítico, os algoritmos genéticos podem utilizar diretamente os valores contínuos ou verdadeiros. Nesse caso, nenhuma conversão é necessária. No entanto devem-se implementar operadores genéticos de cruzamento e mutação adequados à representação.

Não existem tantos estudos de algoritmos genéticos com representação contínua quanto os de representação binária. Neste trabalho serão utilizados os algoritmos genéticos com representação binária.

4.3 Operadores Genéticos

Esta seção aborda os principais operadores genéticos e suas variações mais utilizadas.

4.3.1 Operador de Seleção

Depois de decidir o formato do código (binário ou real), a segunda decisão a fazer é escolher o método de seleção a ser utilizado no algoritmo genético, isto é, como escolher os indivíduos na população que irão gerar os descendentes para a próxima geração e quantos descendentes serão criados.

Seleção é o processo onde os indivíduos com melhor valor de aptidão têm a maior probabilidade de gerar um ou mais descendentes para a geração seguinte. Este operador é a versão artificial da seleção das espécies do Darwinismo, que estabelece que os seres mais aptos têm maiores chances de sobreviver, ou seja, os mais fortes e menos vulneráveis aos predadores e doenças (entenda-se com maior robustez).

O propósito da seleção, portanto, é acentuar os ajustes (fitness) individuais na esperança que os descendentes tenham melhores valores de ajuste (fitness). A seleção deve ser balanceada com variações de cruzamento (crossover) e mutação (mutation), definindo um balanceamento entre exploração e utilização: seleção muito forte significa que um indivíduo com ajuste muito alto terá posição enfatizada na seleção, reduzindo a diversidade necessária para manter o progresso do algoritmo; seleção fraca demais irá resultar em uma evolução lenta demais.

Segundo Deb (2001), o objetivo principal do operador de seleção é copiar boas soluções, eliminando soluções de baixa aptidão, enquanto o tamanho da população é constante. Isto é realizado seguindo os seguintes passos:

1. Identificar boas soluções na população;
2. Realizar múltiplas cópias das boas soluções;
3. Eliminar soluções de baixa aptidão da população o que permite que várias cópias de boas soluções possam ser inseridas na população.

As melhores soluções são guardadas em uma lista de soluções escolhidas ou lista de soluções, que será utilizada para realizar as operações de cruzamento e mutação. Existe um grande número de estratégias de seleção. As mais comuns são seleção pelo torneio, seleção proporcional e seleção por ranking.

Na seleção por torneio, são realizadas várias competições entre duas soluções, e a melhor solução é copiada na lista de soluções. Este processo é repetido até preencher a lista. Goldberg (1989) e Deb (2001) mostraram que este método possui uma convergência igual, ou melhor, que outras estratégias de seleção, além de possuir uma complexidade computacional menor.

Na estratégia de seleção proporcional, o número de cópias de uma solução na lista de soluções escolhidas é proporcional ao seu valor de aptidão. Para calcular o número de cópias esperado é necessário obter a probabilidade de cada solução:

$$p_i = \frac{F_i}{\sum_{i=1}^N F_i} \quad (4.1)$$

sendo que F_i é a aptidão da solução i . N é o tamanho da população. O número de cópias na lista de soluções é calculado por $C_i = p_i N$. Ou seja, as soluções com melhor valor de aptidão terão mais cópias na lista de soluções. O escalonamento é um problema associado a esta estratégia. Quando existe uma solução com um valor de aptidão muito maior comparado ao restante da população, esta super-solução terá uma probabilidade de escolha perto de 1, e terá muitas cópias na lista de soluções. Caso todas as soluções possuam valores similares de aptidão, terão a mesma probabilidade de serem escolhidas e cada uma será copiada na lista de soluções. Isto é equivalente a não realizar operação de seleção.

A estratégia de seleção pelo ranking ordena as soluções população conforme ao seu valor de aptidão, desde a pior solução (ranking 1) até a melhor (ranking N). Depois, o número de cópias das soluções é proporcional ao valor de ranking.

4.3.2 Seleção Proporcional

O algoritmo original de Holland usou seleção de ajuste proporcional, no qual o “valor esperado” de um indivíduo (isto é, o número de vezes que o indivíduo será selecionado para reproduzir) é o valor do ajuste do indivíduo dividido pelo ajuste médio da população. O método mais comum de implementação é a amostragem tipo “roleta”, onde cada indivíduo é representado por uma fatia da roleta, de modo que o tamanho desta fatia seja proporcional ao ajuste do indivíduo. A roleta é girada N vezes, onde N é o tamanho da população. Para cada giro, o indivíduo que cair no marcador da roleta será selecionado para pertencer ao grupo de pais da próxima geração.

A seguir demonstramos o algoritmo do método:

1. Somar o “valor esperado” total da população. Chamar esta soma de **T**.
2. Repetir **N** vezes

Escolher um valor aleatório **r** entre **0** e **T**

Somar os valores esperados dos indivíduos até que a soma seja maior ou igual a **r**. O último indivíduo na somatória de **r** será selecionado.

Este método estocástico estatisticamente resulta em um número esperado de descendentes para cada indivíduo. Portanto, com uma população relativamente pequena, tipicamente usada em algoritmos genéticos, o número real de descendentes gerados por um indivíduo é frequentemente longe do valor esperado (numa situação extrema, poderemos ter todos os descendentes gerados pelo pior ajuste individual).

Baker (1987) propôs um método de amostragem diferente (SUS – stochastic universal sampling – amostragem universal estocástica) para minimizar a dispersão (a faixa de valores reais

possíveis, para um valor esperado). Em lugar da roleta ser girada N vezes para selecionar N pais, o método SUS gira a roleta apenas uma vez, mas com N ponteiros igualmente espaçados, os quais são usados para selecionar N pais.

Importante ressaltar que o método SUS não resolve a maioria dos problemas de seleção por ajuste proporcional. Tipicamente, inicialmente a variância do ajuste na população é alta e um número relativamente pequeno de indivíduos terá um ajuste melhor que os outros. Estes e seus descendentes serão multiplicados mais rapidamente na população, tendo um efeito preventivo do algoritmo genético no processo de exploração. Isto é conhecido como convergência prematura.

Em outras palavras, seleção de ajuste proporcional coloca ênfase demais em utilização de altos ajustes ao custo da exploração de outras partes do espaço de procura. Posteriormente, quando todos os indivíduos na população são muito similares (a variância do ajuste é pequena), não haverá diferenças verdadeiras de ajuste para a seleção ser utilizada e a evolução torna-se dispersa. Assim, a taxa de evolução depende da variância do ajuste na população.

4.3.3 Operador de Cruzamento

O Operador Genético de Cruzamento (Crossover) altera as soluções pelo método de trocar parte dos genes entre dois pais selecionados. Assim, este operador gera novas soluções (descendentes) a partir de soluções escolhidas da lista de soluções (pais). O operador de cruzamento possui diferentes variações, sendo a forma mais simples de cruzamento conhecida como cruzamento em um ponto. Normalmente, seleciona-se dois pais e escolhe-se aleatoriamente um ponto de troca ou ponto de cruzamento. A seguir mostramos este procedimento:

1. Escolher arbitrariamente dois indivíduos na lista de soluções;
2. Escolher dentro da cadeia do indivíduo uma posição k chamada ponto de cruzamento;
3. Criar novos descendentes trocando as cadeias parciais de cada um dos indivíduos.

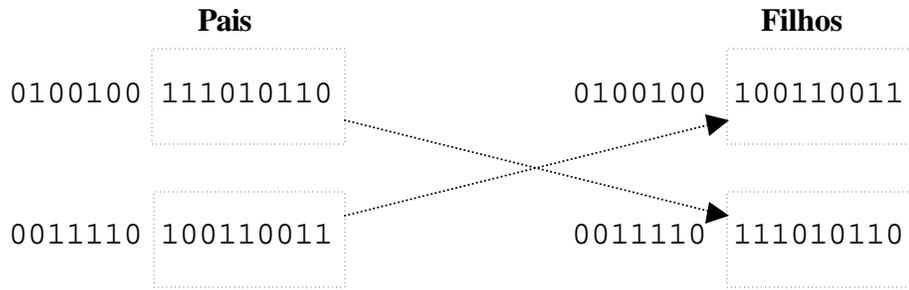


Figura 4.3 – Operador de Cruzamento

É importante notar que o cruzamento possui uma probabilidade associada, geralmente um valor próximo de 1. De acordo com Goldberg (1989), apesar das operações de seleção e cruzamento parecerem muito simples, é nelas que se concentra o poder dos algoritmos genéticos.

4.3.4 Operador de Mutação

A mutação é um operador que produz uma alteração aleatória em uma posição de um pequeno número de indivíduos. A mutação é a segunda maneira dos algoritmos genéticos explorarem o espaço de busca. Esta pequena alteração impede que o algoritmo tenha convergência muito rápida, evitando sua estabilização em regiões de mínimos locais.



Figura 4.4 – Operador de Mutação

4.3.5 Operador de Elitismo

O operador de elitismo mantém as melhores soluções encontradas previamente nas gerações seguintes. Isto implica que a melhor solução não se deteriora nas gerações seguintes. Uma forma de implementar elitismo é copiar diretamente as $n\%$ das soluções da população atual

à população seguinte. O resto das $(100 - n)\%$ das soluções é gerada usando os operadores genéticos usuais sobre a população atual. Desta forma, as melhores soluções passam diretamente para população seguinte, além de participar da criação do resto de soluções da população seguinte.

Outra forma de elitismo consiste em criar a população seguinte a partir da população atual usando os operadores genéticos usuais, e escolher as melhores N soluções de ambas populações.

4.4 Passos para Implementar um Algoritmo Genético

Para resolver um determinado problema utilizando algoritmos genéticos, os seguintes passos devem ser seguidos:

1. Definir uma representação a ser utilizada para indivíduo de maneira que uma solução completa possa ser representada por ele;
2. Definir as estratégias de seleção, cruzamento e mutação;
3. Definir a função de aptidão ou função objetivo;
4. Ajustar os seguintes parâmetros: tamanho da população, probabilidade de cruzamento, probabilidade da mutação e número de gerações.

Geralmente, estes últimos parâmetros variam de acordo com o problema e são ajustados manualmente.

4.5 Diferenças entre os Algoritmos Genéticos e os Métodos de Otimização Tradicionais

Os algoritmos genéticos trabalham com uma população de soluções em cada iteração ao invés de uma única solução. Em cada iteração os algoritmos genéticos processam um conjunto de soluções, esta característica é denominada de paralelismo implícito.

Os algoritmos genéticos não precisam de informação adicional a não ser o valor de aptidão das soluções. Isto faz com que sejam aplicáveis a uma grande variedade de problemas, alguns dos quais não se têm informações a priori.

Os algoritmos genéticos empregam regras probabilísticas para guiar a sua busca. Por exemplo, o processo de seleção é baseado na aleatoriedade de duas soluções (seleção pelo torneio ou por roleta), ou na probabilidade de escolha (seleção proporcional) dessas soluções. O operador de mutação permite que os algoritmos genéticos não recaiam nos ótimos locais, mudando a busca para outra região do espaço. Além do fato que as soluções da população inicial são escolhidas arbitrariamente. Em contrapartida, uma técnica de regras fixas pode não ter como escapar de ótimos locais, em caso de uma má decisão sobre a direção da busca.

A possibilidade de se trabalhar com várias soluções simultaneamente, de não precisar de informações adicionais e poder escapar de ótimos locais fazem dos algoritmos genéticos uma técnica promissora a ser empregada nos problemas de otimização multi-objetivos. Como desvantagem em relação às técnicas tradicionais, existe a grande capacidade computacional necessária. Deste modo, grande ênfase tem sido dado a esta área de estudos a fim de reduzir esta desvantagem.

4.6 Algoritmos Genéticos para Otimização Multi-Objetivos

A primeira implementação de um algoritmo evolutivo para otimização multi-objetivos foi proposta por Schaffer (1984), mostrada em Deb (2001). O modelo sugerido foi denominado VEGA (Vector Evaluated Genetic Algorithm). Schaffer fez uma modificação nos algoritmos genéticos para avaliar cada objetivo separadamente. Um dos problemas do VEGA, é que este algoritmo não obtém boa diversidade nas soluções da Fronteira de Pareto. Goldberg (1989) criou um procedimento para ordenação de soluções baseado no conceito de dominância. Este método fornece um valor de aptidão para uma solução i proporcional ao número de soluções que i domina. Desta forma, as soluções não dominadas são enfatizadas e terão maior quantidade de cópias na lista de soluções. Para manter a diversidade das soluções, Goldberg sugeriu o emprego

de um método de compartilhamento. Com base nas idéias iniciais propostas por Goldberg, foram elaborados vários modelos de algoritmos multi-objetivos evolutivos.

A diferença fundamental dos algoritmos multi-objetivos evolutivos em relação aos algoritmos evolutivos tradicionais é operador de seleção, dado que a comparação entre duas soluções deve realizar-se conforme o conceito de dominância de Pareto. Em alguns métodos, como MOGA e NSGA, o valor de aptidão é proporcional à dominância da solução. Outros métodos, como NPGA, utilizam apenas a dominância de Pareto e não calculam um valor de aptidão.

Os modelos de algoritmos multi-objetivos evolutivos são classificados por Deb (2001) em dois tipos:

1. Não elitistas, são aqueles modelos que como indica o próprio nome, não utilizam alguma forma de elitismo nas suas iterações.
2. Elitistas, modelos que usam de alguma forma o elitismo. Alguns modelos como SPEA, utilizam uma população externa onde são guardadas as soluções não dominadas encontradas até o momento. Outros métodos como NSGA-II combinam a população atual com a população posterior para preservar as melhores soluções de ambas. O estudo realizado por Zitzler et al. (2000) conclui que o elitismo melhora as soluções encontradas. A partir deste trabalho, os novos modelos incorporam alguma estratégia de elitismo.

4.7 Algoritmos Evolutivos Multi-Objetivos Não-Elitistas

Uma forte característica de um método de otimização evolutivo é que uma população de soluções é processada em cada iteração (ou geração). Esta característica proporciona ao algoritmo evolutivo uma excepcional vantagem na solução de problemas de otimização multi-objetivos, pois um dos objetivos de um procedimento ideal de otimização multi-objetivos é achar tantas soluções Pareto-ótimas quantas forem possíveis.

Um algoritmo evolutivo pode ser direcionado para encontrar todas as soluções não dominadas em uma população igualmente e, simultaneamente, preservar uma série diversificada de soluções não-dominadas múltiplas usando um operador de preservação de nicho.

Neste sentido, múltiplas boas soluções podem ser encontradas e mantidas em uma população. Depois de algumas gerações, este processo pode levar a população a convergir para uma Fronteira Pareto-Ótima com uma boa dispersão de soluções.

Deb (2001) apresentou uma lista de possíveis algoritmos evolutivos não-elitistas onde listamos alguns dentre os mais comuns estudados:

- **VEGA** – *Vector Evaluated Genetic Algorithm* – Schafer (1984)
- **VOES** – *Vector Optimized Evolution Strategy* – Kursawe (1990)
- **WBGA** – *Weight Based Genetic Algorithm* – Hajela e Lin (1992)
- **RWGA** – *Random Weighted Genetic Algorithm* – Murata e Ishibuchi (1995)
- **MOGA** – *Multiple Objective Genetic Algorithm* – Fonseca e Fleming (1993)
- **NSGA** – *Non-Dominated Sorting Genetic Algorithm* – Srinivas e Deb (1994)
- **NPGA** – *Niched Pareto Genetic Algorithm* – Horn et all (1994)
- **PPES** – *Predator Prey Evolution Strategy* – Laumans et all (1998)

Neste trabalho faremos a aplicação de dois dos algoritmos acima propostos, sendo o algoritmo genético MOGA proposto por Fonseca e Fleming (1993) e o NSGA proposto por Srinivas e Deb (1994).

4.8 MOGA (Multi-Objective Genetic Algorithm)

Fonseca e Fleming (1993) foram os primeiros a introduzir um algoritmo genético multi-objetivo que usava a classificação de não-dominância de uma população. Este algoritmo foi chamado de MOGA (Multi-Objective Genetic Algorithm). Estes pesquisadores foram os primeiros a sugerir um algoritmo genético multi-objetivo que explicitamente encontra soluções

não-dominadas e simultaneamente mantém a diversidade necessária nestas soluções não-dominadas.

O MOGA se difere de um algoritmo genético clássico porque o ajuste (fitness) é determinado para cada solução da população. O restante do algoritmo (seleção estocástica universal, recombinação, cruzamento e mutação) é o mesmo do algoritmo genético clássico.

4.8.1 Descrição do MOGA

Primeiro, cada solução é checada em relação à sua dominância na população. Para uma solução i , define-se um rank r_i (classificação) igual ao número de soluções h_i que dominam a solução i somando-se 1 . Assim,

$$r_i = 1 + h_i \quad (4.2)$$

Neste caminho, soluções não-dominadas terão rank igual a 1 , uma vez que nenhuma outra solução dominará esta solução não-dominada. Assim, para uma dada população, o rank mínimo será igual a 1 e o valor máximo do rank não passará de N (tamanho da população).

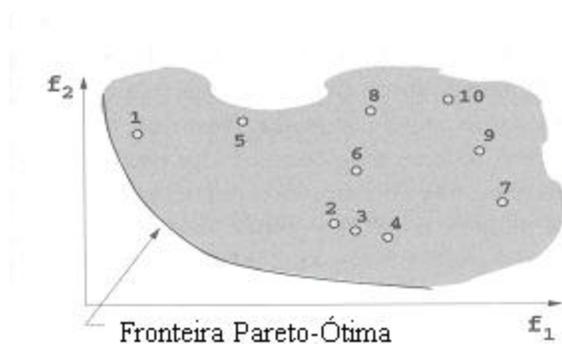


Fig. 4.5 – Espaço de procura de dois objetivos e a fronteira ótima de Pareto

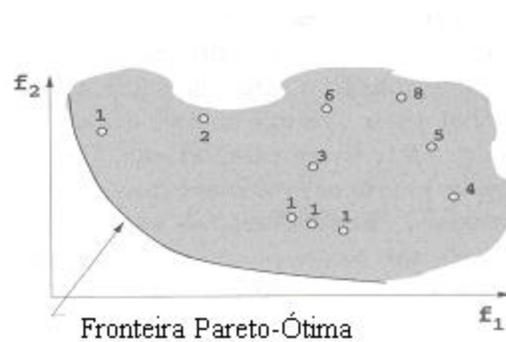


Fig. 4.6 – Procedimento de ranking do MOGA

A fig. 4.5, Deb (2001), mostra um problema com dois objetivos e 10 soluções enquanto a fig. 4.6 mostra a classificação (rank) das soluções. A região hachurada representa o espaço de

procura viável. É claro que o procedimento de ranking pode não determinar todos os valores de ranking entre **1** e **N**. Por exemplo, os rankings 7, 9 e 10 não aparecem na figura 4.6.

Uma vez executado o ranking, uma linha de ajuste é definida de acordo com o ranking. Para isto, primeiro os ranks são posicionados em ordem crescente de magnitude. A linha de ajuste é criada usando uma função de mapeamento que pode ou não ser linear. Normalmente a função de mapeamento é escolhida de modo a definir um ajuste entre **N** (para a melhor solução) e **1** (para a pior solução).

Portanto, soluções de cada rank são consideradas separadamente e se definem ajustes médios chamados “**raw fitness**”. Este ajuste médio é definido para cada solução do rank que está sendo analisado.

Deste modo, o **raw fitness** máximo e o ajuste (fitness) máximo permanecem idênticos. Além disso, os procedimentos de mapeamento e ajuste médio asseguram que as melhores soluções classificadas têm os maiores ajustes **raw fitness**. Assim, as soluções não dominadas são enfatizadas na população.

Lee e Esbensen (1997) também usaram a classificação por rank para definir cópias em uma seleção proporcional. Embora a idéia de converter valores de múltiplos objetivos em métricas de ranqueamento usando o princípio de não-dominância seja um passo eficiente na direção de otimização multi-objetivo, a segunda meta de manter a diversidade de soluções não pode ser atingida com este simples procedimento.

A fim de manter a diversidade entre as soluções não-dominadas, Fonseca e Fleming (1993) introduziram nichos entre as soluções de cada ranking. Utiliza-se a chamada conta do nicho (niche account) com uma dispersão de resultado S_{share} . Uma função de divisão com $\alpha = 1$ é usada, mas a métrica da distância é calculada com os valores da função objetivo ao invés dos valores dos parâmetros. Assim, a distância normalizada entre qualquer duas soluções **i** e **j** em um determinado rank **r** será calculada como:

$$d_{ij} = \sqrt{\sum_{k=1}^M \frac{(f_k^i - f_k^j)^2}{(f_k^{\max} - f_k^{\min})^2}} \quad (4.3)$$

onde f_k^{\max} e f_k^{\min} são os valores máximos e mínimos da função objetivo e M é o número total de funções objetivo.

Para a solução i , d_{ij} é calculada para cada j (incluindo i) dentro do mesmo rank r .

A função de divisão utilizada será:

$$Sh(d_{ij}) = \begin{cases} \frac{1}{1 + \frac{d_{ij}^a}{\sigma_{share}}} & \text{Se } d_{ij} = \sigma_{share} \\ 0 & \text{De outro modo} \end{cases} \quad (4.4)$$

Normalmente, utiliza-se $a = 1$. Com os valores das funções de divisão podemos calcular a conta do nicho.

$$hc_i = \frac{1}{m(r_i)} \sum_{j=1}^{m(r_i)} Sh(d_{ij}) \quad (4.5)$$

onde $m(r_i)$ é o número de soluções no rank r_i . Em um MOGA, a técnica da função de divisão é utilizada e o valor de ajuste relativo é calculado dividindo-se o ajuste de uma solução pela sua conta do nicho. Embora todas as soluções de um rank particular tenham ajustes idênticos, o valor de ajuste relativo de uma solução pertencente a uma região menos “povoada” tem um ajuste relativo melhor. Isto produz uma grande pressão de seleção para soluções pobremente representadas em qualquer rank.

Dividindo o ajuste relativo pelo valor da conta do nicho (sempre igual ou maior que 1) reduz-se o ajuste de cada solução. A fim de manter o mesmo ajuste médio das soluções em um rank antes da divisão, estes valores de ajuste são escalonados de modo que o valor de ajuste médio relativo é o mesmo que o valor do ajuste médio definido.

Depois destes cálculos, o foco é deslocado para as soluções do próximo rank e um procedimento idêntico é executado. Este procedimento é repetido até que todos os ranks sejam processados. Após isto, executa-se o procedimento de algoritmo genético clássico.

4.8.2 Procedimento de ajuste do MOGA

Passo 1 Escolher um S_{share} . Iniciar $m(\mathbf{j}) = 0$ para todos os ranks possíveis $\mathbf{j} = 1, \dots, N$. Fixar o contador de solução $\mathbf{i}=1$.

Passo 2 Calcular o número de soluções (h_i) que dominam a solução \mathbf{i} . Calcular o rank da \mathbf{i} -ésima solução como $r_i = 1 + h_i$, ou seja $m(r_i) = m(r_i) + 1$. Incrementar o contador de soluções no rank r_i por 1.

Passo 3 Se $\mathbf{i} < N$, incrementar \mathbf{i} por 1 e ir para o *Passo 1*. De outro modo ir para *Passo 4*.

Passo 4 Identificar o máximo rank r^* verificando o maior r_i que tem $m(r_i) > 0$. A ordenação de acordo com o rank e com a média dos ajustes produz o seguinte cálculo de ajuste médio para qualquer solução $\mathbf{i} = 1, \dots, N$.

$$F_i = N - \frac{1}{r_i - 1} \sum_{k=1}^{r_i - 1} m(k) - 0,5(m(r_i) - 1) \quad (4.6)$$

Para cada solução \mathbf{i} com rank $r_i = 1$, a equação (4.6) calcula um ajuste igual a $F_i = N - 0,5(m(r_i) - 1)$, o qual é a média de $m(\mathbf{1})$ inteiros consecutivos de N até $N - m(\mathbf{1}) + 1$.

Fixar o contador de rank $\mathbf{r} = 1$.

Passo 5 Para cada solução \mathbf{i} no rank \mathbf{r} , calcular a conta do nicho $h\mathbf{c}_i$ com outras soluções ($\mathbf{m}(\mathbf{r})$ delas) de mesmo rank usando a equação (4.5). Calcular o ajuste relativo usando $\mathbf{F}'_j = \mathbf{F}_j / h\mathbf{c}_i$. Para preservar o mesmo ajuste médio, escalonar o ajuste relativo como segue:

$$\mathbf{F}'_j \leftarrow \frac{\mathbf{F}_j \mathbf{m}(\mathbf{r})}{\sum_{k=1}^{\mathbf{m}(\mathbf{r})} \mathbf{F}'_k} \mathbf{F}'_j \quad (4.7)$$

Passo 6 Se $\mathbf{r} < \mathbf{r}^*$ incrementar \mathbf{r} por $\mathbf{1}$ e ir para o *Passo 5*. De outro modo o procedimento está completo.

4.8.3 Complexidade Computacional

Para obter os valores de r_i , cada uma das \mathbf{N} soluções deve ser comparada com $\mathbf{N} - \mathbf{1}$ soluções, em cada um dos \mathbf{M} objetivos. O número de comparações necessárias é da ordem $\mathbf{O}(\mathbf{MN}^2)$. O cálculo da aptidão média, dos nichos e da aptidão final, no pior caso, quando todas as soluções estão no mesmo ranking e no mesmo nicho, é da ordem $\mathbf{O}(\mathbf{N}^2)$. Portanto, a complexidade de MOGA é de $\mathbf{O}(\mathbf{MN}^2)$, segundo Deb (2001).

4.8.4 Vantagens

A principal vantagem do MOGA é a sua simplicidade no cálculo do valor de aptidão. Dado que o cálculo de nichos é realizado no espaço de objetivos, este algoritmo é facilmente aplicável a outros problemas de acordo com Deb (2001). Este algoritmo é adequado quando se deseja obter uma dispersão grande nas soluções encontradas.

4.8.5 Desvantagens

O valor n não fornece os mesmos valores para soluções que se encontram na mesma fronteira, exceto a primeira. Isto pode gerar um ruído não desejado em algumas soluções do espaço de busca. Deb (2001) afirma que este algoritmo é sensível à forma da Fronteira de Pareto e à densidade das soluções no espaço de busca. O cálculo do compartilhamento não garante que soluções com altos valores de n terão uma pior aptidão que soluções com n baixos. Isto acontece quando existem muitas soluções próximas entre si com n baixos. O valor de contador de nicho para estas soluções é alto e o ajuste definido é pequeno. Neste caso, não existe uma boa seleção das soluções de melhor n e, então, a convergência do algoritmo é degradada.

4.9 NSGA (Non-Dominated Sorting Genetic Algorithm)

A idéia de Goldberg (1989) de usar o conceito de classificação não-dominada em algoritmo genético foi mais diretamente implementada por Srinivas e Deb (1994). Uma vez mais, o duplo objetivo em algoritmo de otimização multi-objetivo foi mantido através do esquema de definição de ajuste (fitness), que prefere soluções não-dominadas, e pelo uso de uma divisão estratégica, que preserva a diversidade entre as soluções de cada fronteira não-dominada. Nos seguintes parágrafos nós iremos discutir o algoritmo NSGA (Non-Dominated Sorting Genetic Algorithm) em detalhes.

4.9.1 Descrição do NSGA

O primeiro passo de um algoritmo NSGA é classificar a população \mathbf{P} de acordo com a não-dominância. Isto classifica a população dentro de um número de classes equivalentes exclusivas (séries não-dominadas) \mathbf{P}_j :

$$\mathbf{P} = \hat{\cup}_{j=1}^r \mathbf{P}_j \quad (4.8)$$

É importante lembrar que qualquer um dos dois membros de uma mesma classe não pode ser dito melhor do que o outro com respeito a todos os objetivos. O número total de classes (ou fronteiras) denotadas por r na equação acima, depende da população \mathbf{P} e do problema subjacente. Como foi discutido antes, o procedimento de classificar completamente a população de acordo com a não-dominância é no máximo $O(MN^2)$. Vamos considerar a população mostrada na figura 4.5. Depois da classificação não-dominada, os membros da população estarão distribuídos em quatro séries não-dominadas distintas, como mostrado na figura 4.7.

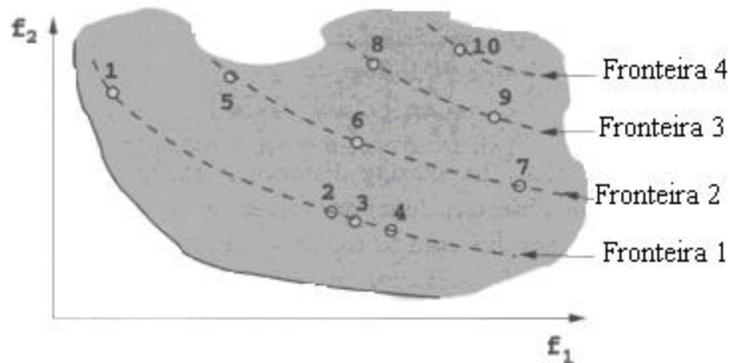


Figura 4.7 – Fronteiras no NSGA

Uma vez que a fase de classificação está finalizada, fica claro que todas as soluções que estão na primeira série, isto é, todos $i \in \mathbf{P}_1$, constituem a melhor série não-dominada na população. As segundas melhores soluções são aquelas que pertencem à segunda série, sendo todos os membros de \mathbf{P}_2 , e assim por diante. Obviamente, as piores soluções são aquelas pertencentes à série final, ou seja, todos os membros de \mathbf{P}_r , onde r é o número de diferentes séries não-dominadas na população. Examinando a fig. 4.7 fica claro que as soluções da primeira fronteira são melhores em termos de aproximação com a fronteira ótima de Pareto. Assim, faz sentido atribuir os maiores valores de ajuste para a melhor fronteira não-dominada e progressivamente obter os piores valores para as soluções pertencentes às fronteiras com maior índice i .

Isto é devido ao fato que as melhores soluções não-dominadas dentro de uma população estão perto de fronteira ótima de Pareto comparada com as outras soluções da população.

Seguindo, descreve-se o procedimento exato de atribuição de ajuste (fitness) que foi adotado por Srinivas e Deb (1994).

O procedimento de atribuição de ajuste começa com a primeira série não-dominada e sucessivamente avança para as outras séries. Para qualquer solução i da primeira (melhor) série não-dominada é atribuído um ajuste $F_i = N$ (tamanho da população). Este valor específico de N é usado com um propósito particular. Uma vez que todas as soluções pertencentes à primeira série não-dominada são igualmente importantes em termos de aproximação com a fronteira ótima de Pareto, deve-se definir o mesmo valor de ajuste para todas elas. Com respeito à fig. 4.7, as soluções 1 a 4 terão valor de ajuste igual a 10 (tamanho da população).

Atribuindo-se maiores ajustes às soluções pertencentes às melhores séries não-dominadas assegura-se uma pressão de seleção em direção à fronteira ótima de Pareto. Entretanto, a fim de se obter o segundo objetivo, uma diversidade entre as soluções na fronteira deve ser mantida. A menos que um mecanismo explícito de preservação da diversidade seja usado, os algoritmos genéticos evolutivos não asseguram esta diversidade. No NSGA, a diversidade é mantida degradando-se o ajuste baseando-se na quantidade de soluções vizinhas.

A fim de explicar a necessidade de preservação da diversidade, vamos nos referir às soluções da primeira fronteira da fig. 4.7. É claro que a fronteira não é representada adequadamente pelas quatro soluções (1 a 4). Três soluções (2 a 4) são agrupadas em uma porção da fronteira, sendo que somente uma solução representa a parte superior esquerda da curva. Se a solução 1 não satisfaz adequadamente o algoritmo genético, este irá tentar achar outra fronteira mais próxima da solução ótima de Pareto. A fim de evitar este problema, temos que ter certeza que menos regiões fechadas em uma fronteira são encontradas. Em um NSGA, o método da função de divisão (sharing function method) é usado para este propósito.

O método da função de divisão é usado em cada fronteira. Isto é, para cada solução i na fronteira P_1 , a norma euclidiana d_{ij} em relação à outra solução j na mesma fronteira é calculada como segue:

$$d_{ij} = \sqrt{\frac{|P_i| \sum_{k=1}^n \frac{x_k^{(i)} - x_k^{(j)}}{x_k^{\max} - x_k^{\min}}}{\theta}}^2 \quad (4.9)$$

É importante notar que a distância de divisão é calculada com as variáveis de decisão, ao invés da função objetivo como ocorre no algoritmo MOGA. Uma vez que estas distâncias são calculadas, são usadas para calcular o valor da função de divisão. A função de divisão assume o valor entre zero e a unidade, dependendo da distância d_{ij} . Qualquer solução j que apresente distância maior que S_{share} da i -ésima solução não contribui para o valor da função de divisão.

Depois que todos os valores das funções de divisão $|P_i|$ forem calculados, estes serão adicionados juntos para calcular a conta do nicho hc_i da i -ésima solução. A conta do nicho define o número de soluções na vizinhança da i -ésima solução incluindo-a. Se não existe outra solução dentro do raio S_{share} , a conta do nicho poderia ser igual a unidade. Por outro lado, se todas as soluções $|P_i|$ na fronteira forem muito próximas umas das outras comparadas com S_{share} , a conta do nicho de qualquer solução no grupo pode ser próxima a $|P_i|$. Por exemplo, se um S_{share} apropriado for escolhido, a conta do nicho da solução 1 na fig. 4.7 deverá ser um, sendo que a conta do nicho para as outras três soluções (2,3 e 4), todas na primeira fronteira, deverá ser próximo de três.

O propósito final é reduzir o ajuste da i -ésima solução pelo seu valor de nicho e obter o valor de ajuste dividido $F'_i = F_i / hc_i$. Este processo de degradação do ajuste de uma solução que está agrupada por muitas soluções ajuda a enfatizar as soluções residindo em regiões menos agrupadas. Este é precisamente o propósito do mecanismo de preservar a diversidade. Na figura 4.7, o valor de ajuste relativo da solução 1 é $F'_1=10$, sendo que para as outras três soluções teremos valor próximo de 10/3 ou 3,333.

Este processo completou o procedimento de definição de ajuste de todas as soluções na primeira fronteira. Com a finalidade de solucionar a segunda fronteira, tomamos o valor mínimo de ajuste dividido da primeira fronteira e então definimos um valor de ajuste levemente menor este valor mínimo. Isto assegura que o menor valor da primeira fronteira é maior que qualquer

valor da segunda fronteira. Repetimos o método da função de divisão entre as soluções da segunda fronteira e calculamos os respectivos valores dos ajustes relativos. Continuamos este processo até que seja definido um valor de ajuste a todas as soluções do problema.

O algoritmo NSGA utiliza o operador de roleta definido por Goldberg no método de ajuste e assim, copia a população proporcionalmente ao valor de ajuste relativo. Deste modo, cada solução na primeira fronteira tem uma melhor chance de sobreviver na população do que as soluções da segunda fronteira, e assim por diante. Por outro lado, a abordagem da função de divisão entre as soluções em cada fronteira certamente fará que soluções que estão isoladas terão mais cópias na população. Deste modo, as duas metas da otimização multi-objetivos serão alcançadas pelo procedimento de definição de ajuste. Como mencionado, os operadores de mutação e de cruzamento são aplicados na população inteira.

4.9.2 Procedimento de ajuste do NSGA

Passo 1 Escolher o parâmetro s_{share} e um pequeno número positivo ϵ e iniciar $F_{min} = N + \epsilon$. Iniciar o contador $j=1$.

Passo 2 Classificar a população P de acordo com a não-dominância:
 $(P_1, P_2, \dots, P_r) = \text{Sort}(P, \preceq)$

Passo 3 Para cada $q \in \{1, \dots, r\}$

Passo 3a Definir o ajuste $F^{(q)}_j = F_{min} - \epsilon$

Passo 3b Calcular a conta do nicho $h c_q$ entre as soluções de P_j somente.

Passo 3c Calcular o ajuste relativo $F'^{(q)}_j = F^{(q)}_j / h c_q$

Passo 4 $\mathbf{F}_{\min} = \min(\mathbf{F}^{(q)}_j : \mathbf{q} \hat{=} \mathbf{P}_j)$ e fazer $j = j + 1$

Passo 5 Se $j \leq r$ volte para *Passo 3*. Se não, o processo terminou.

Este procedimento de ajuste pode ser colocado em um algoritmo genético de simples-objetivo. O operador de seleção proporcional deve ser usado. Entretanto, qualquer operador de mutação ou cruzamento poder ser utilizado.

4.9.3 Complexidade Computacional

A complexidade computacional do procedimento de definição do ajuste acima citado é principalmente governada pelo procedimento de ordenação não-dominada e pela implementação da função de divisão. O primeiro requer uma complexidade $O(MN^2)$. A implementação da função de divisão requer que toda solução dentro de uma fronteira seja comparada com todas as outras soluções dentro da mesma fronteira, portanto executando uma grande quantidade de cálculos de distâncias, sendo que cada distância requer o cálculo de n diferenças entre os parâmetros. No pior caso (quando todas as soluções pertencerem a uma única fronteira) esta computação é $O(NM^2)$. Assim, a complexidade total será o maior valor entre $O(MN^2)$ e $O(NM^2)$.

4.9.4 Vantagens

A principal vantagem do algoritmo NSGA é a definição do ajuste de acordo com as séries não-dominadas. Uma vez que as melhores séries não-dominadas são enfatizadas sistematicamente, um NSGA progride em direção a uma região de fronteira Pareto-Ótima. Além disso, a execução de uma divisão no espaço de parâmetros permite que soluções diversas possam emergir quando usamos NSGA. Se desejado, a divisão pode ser executada também no espaço de busca.

4.9.5 Desvantagens

A técnica da função de divisão requer o ajuste do parâmetro σ_{SHARE} . Foi anteriormente observado por Srinivas e Deb (1994) que a performance de um NSGA é sensível a este parâmetro. Entretanto, se adequadamente escolhido, uma dispersão razoavelmente boa das soluções pode ser encontrada. Deb (2001) fornece sugestões para melhorar a performance.

4.10 Algoritmos Evolutivos Multi-Objetivos Elitistas

Nos parágrafos anteriores, apresentamos um número de algoritmos evolutivos multi-objetivos que não usam um operador preservador de elites. Como o nome sugere, um operador preservador de elites favorece as elites de uma população proporcionando uma oportunidade de ser diretamente transferida para a próxima geração. Em problemas de otimização simples-objetivo, o elitismo é introduzido em algoritmo genético por vários caminhos. Em um algoritmo evolutivo em regime constante, o elitismo pode ser introduzido de uma maneira simples.

Depois de dois descendentes serem criados usando operadores de cruzamento e mutação, ambos são comparados aos seus pais. Portanto, entre as quatro soluções pais/filhos as duas melhores serão selecionadas, permitindo assim, que pais elites compitam com seus descendentes por uma vaga na próxima geração. Elitismo também pode ser introduzido globalmente no sentido geracional. Uma vez que a população descendente é criada, pais e filhos podem ser combinados juntos. Portanto, os melhores N membros podem ser escolhidos para formar a próxima geração. Neste sentido também, os pais tem a chance de competir com os filhos e sobreviver para a próxima geração.

Não importando quanto elitismo é introduzido, existe a certeza que os melhores ajustes da população não se deteriorarão. Assim, uma boa solução encontrada nas primeiras gerações não será perdida a menos que uma solução melhor apareça e a ausência de elitismo não garante este aspecto. De fato, Rudolph (1996) provou que os algoritmos genéticos convergem para as soluções ótimas globais de algumas funções na presença de elitismo. Além do mais, a presença de elites aumenta a probabilidade de criar melhores descendentes.

Baseado no exposto acima, está claro que elitismo, de alguma forma, é importante em algoritmos evolutivos. Embora elitismo seja importante, ainda não está claro com que extensão o elitismo deve ser introduzido dentro de um algoritmo genético multi-objetivos, uma vez que elitismo pode ser implementado em diversos graus. Por exemplo, pode-se manter um conjunto de soluções ótimas dentro de uma população e atualizá-las se melhores soluções aparecem nas gerações subseqüentes, porém não utilizando elitismo dentro dos operadores genéticos. Esta operação não resultará em vantagem genética que poderia ser explorada das elites. Por outro lado, em outro extremo de implementação, todas as elites presentes em uma população poderiam ser passadas para a próxima geração, limitando assim, a quantidade de novas soluções que poderiam ser melhores que as soluções elites e deste modo, o processo de procura pode não progredir.

As implementações interessantes são aquelas onde um grau intermediário de elitismo é introduzido. Na maioria das implementações simples-objetivo, as melhores l soluções da população são usadas como elites. Entretanto, a escolha de um valor apropriado de l torna-se importante no sucesso do trabalho do algoritmo. O parâmetro l está diretamente relacionado à *pressão de seleção* associada com as elites. Uma vez que as soluções elites são diretamente transferidas para a próxima geração e também participam das operações genéticas, tendem a influenciar a população para se agrupar em torno delas mesmas. Se um grande l é usado, esta influência é maior e a população perde a diversidade. Entretanto, se um pequeno l é usado, uma vantagem adequada do uso de elitismo não será aproveitada. Valores de $l = 1$ até $l = 0,1N$ (10% do tamanho da população) são comuns e podem variar com as gerações ou permanecer constante durante todo o ciclo do algoritmo genético multi-objetivos.

Na otimização simples-objetivo, as elites são fáceis de serem identificadas, pois obviamente são as melhores soluções (por exemplo, com o maior valor da função objetivo) da população. Entretanto, o mesmo não ocorre no procedimento de otimização multi-objetivos, não sendo possível identificar as elites diretamente das funções objetivos sendo necessário um critério baseado em ranking e não-dominância. Assim, como já foi mencionado anteriormente, todas as melhores soluções não-dominadas tornam-se elites de mesma importância. Neste caso, introduzir $l = 1$ significa que todas as soluções pertencentes à primeira fronteira não dominada são

chamadas elites. Para $l = 2$ teremos também as soluções da segunda fronteira não-dominada incluídas como elites e assim por diante. Uma vez que o número de soluções não-dominadas dentro de uma fronteira não é controlado, torna-se difícil introduzir controle de elitismo em uma otimização multi-objetivos. Além do mais, o número de soluções não-dominadas na primeira fronteira geralmente é grande, particularmente quando temos um grande número de objetivos.

Assim, mesmo introduzindo elitismo com $l = 1$ teremos uma grande proporção de soluções elites vindo da geração anterior e isto irá causar perda de diversidade rapidamente. A presença de elitismo melhorará a performance de um algoritmo evolutivo multi-objetivos, mas cuidado deve ser tomado a fim de reduzir este efeito negativo.

Deb (2001) descreve 11 (onze) algoritmos evolucionários multi-objetivos elitistas. Abaixo segue uma lista dos sete principais

- **Rudolph's Elitist Multi-Objective Evolutionary Algorithm** – Rudolph (2001)
- **NSGA – II – Elitist Non-Dominated Sorting Genetic Algorithm** – Deb et al. (2000a, 2000b)
- **DPGA – Distance Based Pareto Genetic Algorithm** – Osyczka e Kundu (1995)
- **SPEA – Strength Pareto Evolutionary Algorithm** – Zitzler e Thiele (1998a)
- **TDGA – Thermodynamical Genetic Algorithm** – Kita et al. (1996)
- **PAES – Pareto Archived Evolution Strategy** – Knowles e Corne (2000)
- **MOMGA – Multi-Objective Messy Genetic Algorithm** – Veldhuizen (1999)

Neste trabalho faremos a aplicação de dois dos algoritmos acima propostos, sendo o NSGA-II proposto por Deb et al. (2000a, 2000b) e o algoritmo genético SPEA proposto por Zitzler e Thiele (1998a).

4.11 NSGA-II (Non-Dominated Sorting Genetic Algorithm II)

Deb e seus estudantes (2000a, 2000b) sugeriram um algoritmo genético de ordenação não-dominada elitista. Ao contrário do método que utiliza somente uma estratégia de preservação por

elite, **NSGA-II** também utiliza um mecanismo explícito de preservação da diversidade. Em muitos aspectos, este algoritmo não tem muita similaridade com o **NSGA** original, mas os autores mantiveram o nome **NSGA-II** para realçar sua origem.

4.11.1 Descrição do NSGA-II

No **NSGA-II**, primeiramente é criada uma população de descendentes Q_t usando a população de pais P_t . Entretanto, ao invés de achar somente a fronteira não-dominada de Q_t , as duas populações são combinadas para formar uma nova população R_t de tamanho $2N$. Então, uma ordenação não-dominada é usada para classificar a população R_t . Embora isto requiera mais esforço comparado a executar somente uma ordenação na população Q_t , permite uma verificação global de não-dominância entre as soluções dos pais e dos descendentes. Uma vez que a ordenação não-dominada está completa, a nova população é preenchida por soluções de diferentes fronteiras não-dominadas, uma de cada vez. O preenchimento começa com a melhor fronteira não-dominada e continua com as soluções da segunda melhor fronteira não-dominada, seguida pela terceira não-dominada e assim por diante. Uma vez que o tamanho total da população R_t é $2N$, nem todas as fronteiras podem ser acomodadas em N espaços disponíveis na nova população. Todas as fronteiras que não foram acomodadas são simplesmente excluídas. Quando a última fronteira permitida está sendo considerada, poderão existir mais soluções nesta última fronteira do que os espaços remanescentes na nova população. Este cenário é ilustrado na Figura 4.8. Ao invés de arbitrariamente descartar alguns membros da última fronteira, uma boa escolha é usar uma estratégia de nicho para escolher os membros da última fronteira, os quais residem na região menos aglomerada desta fronteira.

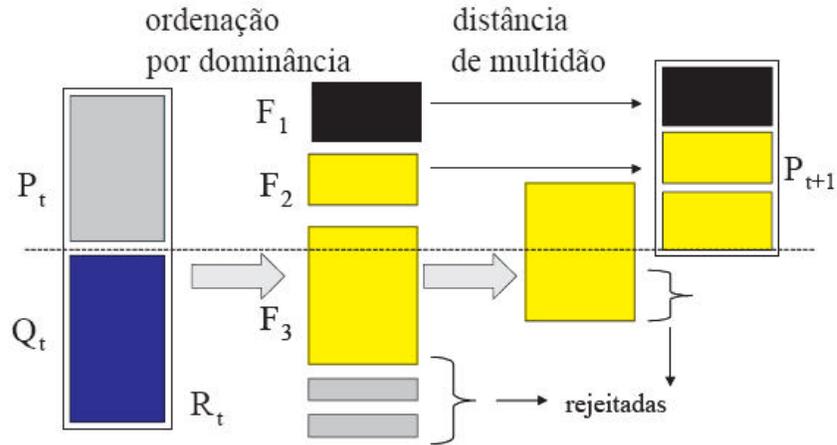


Figura 4.8 – Agrupamento no NSGA-II

Uma estratégia como esta, descrita acima, não afeta muito os procedimentos do algoritmo nos primeiros estágios da evolução. Isto ocorre porque, no início, existem muitas fronteiras na população combinada R_t . É provável que soluções de muitas fronteiras não-dominadas boas já estejam incluídas na nova população, antes delas serem adicionadas a N . Assim, não é necessário preocupar-se com as soluções incluídas na nova população. Entretanto, durante os últimos estágios da simulação, é provável que a maioria das melhores soluções estejam na melhor fronteira não-dominada. É também provável que na população combinada R_t de tamanho $2N$, o número de soluções na primeira solução não-dominada exceda o valor de N . O algoritmo então assegura que o processo de nicho irá escolher uma série diversa de soluções a partir desta fronteira. Quando a população inteira convergir para a Fronteira Ótima de Pareto, a continuação do algoritmo irá assegurar uma melhor dispersão entre estas soluções.

A seguir, descreve-se o algoritmo passo-a-passo. Inicialmente, uma população aleatória P_0 é criada. A população então é ordenada dentro de diferentes níveis de não-dominância. Para cada solução é definido um nível de ajuste que é igual ao seu nível de não-dominância (sendo 1 o melhor nível). Assim, necessitamos de uma minimização do ajuste. Seleção por torneio binário, operadores de recombinação e mutação são usados para criar uma população descendente Q_0 de tamanho N .

4.11.2 Procedimento de ajuste do **NSGA-II**

- Passo 1* Combinar as populações dos pais e dos descendentes criando a população \mathbf{R}_t de tamanho $2N$ ($\mathbf{R}_t = \mathbf{P}_t \cup \mathbf{Q}_t$). Executar a ordenação não-dominada em \mathbf{R}_t e identificar as diferentes fronteiras: $\mathbf{F}_i, i = 1, 2, \dots, \text{etc}$
- Passo 2* Criar a nova população $\mathbf{P}_{t+1} = \emptyset$. Fixar o contador $i = 1$.
Até que $|\mathbf{P}_{t+1}| + |\mathbf{F}_i| < N$, executar $\mathbf{P}_{t+1} = \mathbf{P}_{t+1} \cup \mathbf{F}_i$ e incrementar o contador $i = i + 1$
- Passo 3* Executar o procedimento de cálculo da distância de agrupamento (item 4.11.5) e incluir as $(N - |\mathbf{P}_{t+1}|)$ soluções mais dispersas usando a distância de agrupamento na ordenação entre \mathbf{F}_i até \mathbf{P}_{t+1} .
- Passo 4* Criar uma população descendente \mathbf{Q}_{t+1} a partir de \mathbf{P}_{t+1} usando procedimento de seleção por torneio agrupado (item 4.11.3) e aplicar os operadores de cruzamento e mutação.

No passo 3, a ordenação por agrupamento das soluções da fronteira i (a última fronteira que poderia não ser acomodada completamente) é executada usando-se uma métrica de distância de agrupamento (crowding distance metric, item 4.11.4). A população é re-alinhada em ordem decrescente de magnitude dos valores das distâncias de agrupamento. No passo 4, um operador de seleção por torneio por agrupamento que irá usar esta mesma distância será aplicado.

É importante mencionar aqui que a ordenação não-dominada indicada no passo 1 e o preenchimento da população \mathbf{P}_{t+1} podem ser executados em conjunto. Deste modo, toda vez que uma fronteira não-dominada é encontrada, seu tamanho pode ser usado para verificar se esta pode ser incluída na nova população. Se isto não for possível, não será mais necessária a ordenação. Isto irá ajudar a reduzir o tempo de execução do algoritmo.

4.11.3 Operador de Seleção por Torneio Agrupado

O operador de comparação agrupado (\llcorner) compara duas soluções e retorna a vencedora do torneio, assumindo que toda solução i tem dois atributos:

- 1) Um rank r_i não-dominado na população
- 2) Uma distância local de agrupamento d_i na população.

A distância de agrupamento d_i de uma solução i é a medida do espaço de procura em torno de i o qual não é ocupado por outra solução na população. Baseado nestes dois atributos, definimos o operador de seleção por torneio agrupado como segue:

Definição - Operador de Seleção por Torneio Agrupado: Uma solução i vence um torneio contra outra solução j se qualquer das seguintes condições são verdadeiras:

1. Se a solução i tem melhor classificação, isto é, $r_i < r_j$
2. Se ambas apresentam a mesma classificação, porém a solução i tem uma melhor distancia de agrupamento do que a solução j , isto é, $r_i = r_j$ e $d_i > d_j$.

A primeira condição garante que a solução escolhida esteja contida na melhor fronteira não-dominada. A segunda condição soluciona o problema de ambas as soluções estarem na mesma fronteira não-dominada e decide através da distância de agrupamento. Neste caso, a solução pertencente a uma região menos populosa (com maior distância d_i) vence a seleção. A distância de agrupamento d_i pode ser calculada de diversas maneiras. Dentre estas, a métrica da conta do nicho h_{c_i} e a métrica da conta de cabeças h_{c_i} são comuns. Ambas estas métricas requerem complexidade computacional $O(MN^2)$. Embora possam ser usadas, o fazem em sentido contrário. Uma solução tendo uma menor conta do nicho ou menor conta de cabeças significa uma solução menos densa e deve ser preferida. Assim, na segunda condição, deve-se escolher i se $r_i = r_j$ ou também $h_{c_i} < h_{c_j}$ ou, em termos de conta de cabeças, $h_{c_i} < h_{c_j}$. Entretanto, dentro do **NSGA-II**, utiliza-se uma métrica de distância de agrupamento que requer $O(MN \log N)$ computações.

4.11.4 Distância de Agrupamento

Para se ter uma estimativa da densidade de soluções em torno de uma solução particular \mathbf{i} na população, tomaremos a distância média de duas soluções em ambos os lados da solução \mathbf{i} considerando todos os objetivos. Esta quantidade \mathbf{d}_i serve como estimativa do perímetro do cubóide formado utilizando-se os vizinhos mais próximos como vértices (chamaremos de distância de agrupamento). Na figura 4.9, a distância de agrupamento da \mathbf{i} -ésima solução na sua fronteira (marcada com círculos sólidos) é o comprimento médio do lado do cubóide \mathbf{d}_i (retângulo superior, mostrado por uma linha tracejada). O seguinte algoritmo é usado para calcular a distância de agrupamento de cada ponto na fronteira \mathbf{F} .

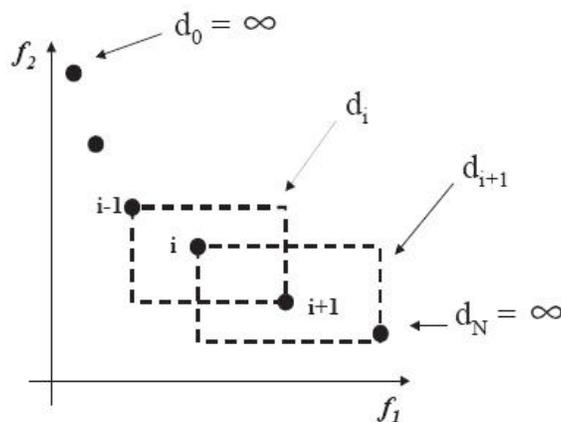


Figura 4.9 – Cálculo da distância de agrupamento

4.11.5 Procedimento de Cálculo da Distância de Agrupamento ($\mathbf{F}, <c$)

Passo C1 Chamar o número de soluções em \mathbf{F} como sendo $l = |\mathbf{F}|$. Para cada \mathbf{i} na série definir $\mathbf{d}_i = 0$.

Passo C2 Para cada função objetivo $\mathbf{m} = 1, 2, \dots, \mathbf{M}$, ordenar a série do pior para o melhor de \mathbf{f}_m ou, achar o vetor de índices ordenados $\mathbf{I}^m = \text{ordenar}(\mathbf{f}_m, <)$.

Passo C3 Para $m = 1, 2, \dots, M$, definir uma distância suficientemente elevada para as soluções no contorno, ou $\mathbf{d}_{I_1}^m = \mathbf{d}_I^m = \mathbf{8}$ e para todas as outras soluções $\mathbf{j} = 2$ até $(l - 1)$, definindo:

$$\mathbf{d}_{I_j}^m = \mathbf{d}_{I_j}^m + \frac{\mathbf{f}_m^{(I_{j+1}^m)} - \mathbf{f}_m^{(I_{j-1}^m)}}{\mathbf{f}_m^{\max} - \mathbf{f}_m^{\min}} \quad (4.10)$$

O índice I_j relaciona-se ao índice da solução do j -ésimo membro da lista ordenada. Assim, para qualquer objetivo, I_1 e I_l representam respectivamente o menor e o maior valor da função objetivo. O segundo termo a direita da equação é a diferença entre os valores das funções objetivos entre duas soluções vizinhas, em ambos os lados da solução I_j . Assim, esta métrica define metade do perímetro do cubóide inscrito com as soluções vizinhas mais próximas sendo estas alocadas nos vértices do cubóide (figura 4.9). É interessante notar que para qualquer solução i as mesmas soluções $(i + 1)$ e $(i - 1)$ não necessitam serem vizinhas em todos os objetivos, particularmente para $M = 3$. Os parâmetros \mathbf{f}_m^{\max} e \mathbf{f}_m^{\min} podem ser fixados como os valores máximo e mínimo da população para a m -ésima função objetivo.

A métrica acima requer M cálculos de ordenação no *passo C2*, cada um requerendo $O(N \log N)$ computações. O *passo C3* requer N computações. Assim, a complexidade da métrica acima é $O(MN \log N)$. Para um grande valor de N , esta complexidade será menor que $O(MN^2)$, que define a complexidade computacional requerida em outros métodos baseados em nichos.

4.11.6 Complexidade computacional

O *passo 1* do **NSGA-II** requer uma ordenação não-dominada da população de tamanho $2N$. Isto requer no máximo $O(MN^2)$ computações. Ordenação no *passo 3* requer a computação da distância de agrupamento para todos os membros da fronteira i . Entretanto, a seleção por torneio agrupado no *passo 4*, requer as computações das distâncias de agrupamento da população inteira P_{t+1} de tamanho N . Isto requer $O(MN \log N)$ computações. Assim a complexidade total do **NSGA-II** é no máximo $O(MN^2)$.

4.11.7 Vantagens

A diversidade entre as soluções não-dominadas é introduzida através do uso de procedimento de comparação de agrupamento, o qual é usado durante a seleção por torneio e redução de fase da população. Uma vez que as soluções competem com as suas distâncias de agrupamento (uma medida da densidade de soluções na vizinhança), nenhum parâmetro extra de nicho (tais como S_{SHARE} necessário em **MOGA**, **NSGA** ou **NPGA**) é requerido. Deb (1999) descreve que embora a distância de agrupamento seja calculada no espaço da função objetivo, esta pode ser implementada no espaço de parâmetros, caso desejado.

Na ausência do operador de comparação agrupado, este algoritmo também exibe uma prova de convergência da solução ótima de Pareto similar ao algoritmo proposto por Rudolph, mas o tamanho da população poderia crescer com o contador de geração. O mecanismo de elitismo não permite que uma solução de Pareto já descoberta seja eliminada.

4.11.8 Desvantagens

Quando a comparação de agrupamento é usada para restringir o tamanho da população, o algoritmo perde sua propriedade de convergência. Contanto que o tamanho da primeira série não-dominada não seja maior que o tamanho da população, este algoritmo preserva todos eles. Entretanto, para gerações posteriores, quando mais do que N membros pertencem a primeira série não-dominada em uma população combinada pais-filhos, alguma solução ótima de Pareto pode dar lugar a uma solução ainda não dominada. Embora estas soluções podem ser dominadas por outras soluções ótimas de Pareto em gerações seguintes, o algoritmo pode permanecer um certo tempo dentro do ciclo de gerar soluções aleatórias antes de finalmente convergir para uma série bem distribuída de soluções ótimas de Pareto.

Como mencionado anteriormente, a ordenação não-dominada necessita ser executada em uma população de tamanho $2N$, ao invés de uma população de tamanho N requerida na maioria dos algoritmos genéticos.

4.12 SPEA – Strength Pareto Evolutionary Algorithm

Zitzler and Thiele (1998a) propuseram um algoritmo evolutivo elitista o qual chamaram de Strength Pareto EA (SPEA). Este algoritmo introduz elitismo através de explicitamente manter um população externa $\bar{\mathbf{P}}$. Esta população armazena um número fixo de soluções não dominadas que são encontradas até o início de uma simulação. Em todas as gerações, as soluções não-dominadas recém encontradas são comparadas com as soluções existentes e as soluções não-dominadas resultantes são preservadas. O SPEA simplesmente preserva as elites; e também usa estas elites para participar de operações genéticas na população corrente com o intuito de influenciar a população a seguir na direção de regiões de boas soluções no espaço de procura.

4.12.1 Descrição do SPEA

Este algoritmo começa com a criação de uma população aleatória \mathbf{P}_0 de tamanho \mathbf{N} e uma população externa vazia $\bar{\mathbf{P}}$ com uma máxima capacidade $\bar{\mathbf{N}}$. Em alguma geração \mathbf{t} , as melhores soluções não-dominadas (pertencentes à primeira fronteira não-dominada) da população \mathbf{P}_t são copiadas para a população externa $\bar{\mathbf{P}}_t$. Portanto, as soluções dominadas pertencentes à população externa modificada são determinadas e excluídas da população externa. Deste modo, elites previamente encontradas e que agora são dominadas por uma nova elite de soluções são excluídas da população externa. O que permanece na população externa são as melhores soluções não-dominadas de uma população combinada contendo novas e velhas elites. Se este processo continua por muitas gerações, existe o perigo da população externa tornar-se sobrecarregada de soluções não-dominadas.

A fim de restringir o super crescimento da população o tamanho da população externa é limitado em ($\bar{\mathbf{N}}$). Isto é, quando o tamanho da população é menor que $\bar{\mathbf{N}}$, todas as elites são mantidas na população. Entretanto, quando o tamanho excede $\bar{\mathbf{N}}$, nem todas as elites podem ser acomodadas na população externa. Desta forma, os investigadores do SPEA consideram satisfazer o segundo objetivo da otimização multi-objetivos. Elites que são menos agrupadas na fronteira não-dominada são mantidas. Somente aquelas poucas soluções que são necessárias para

manter a população fixa no tamanho \bar{N} são retidas. Os investigadores sugeriram um método de agrupamento para alcançar este objetivo.

Uma vez que novas elites são preservadas para a próxima geração, o algoritmo retorna para a população corrente e usa operadores genéticos para achar a nova população. O primeiro passo é definir um ajuste para cada solução da população. Foi mencionado anteriormente que o SPEA também usa sua população externa \bar{P}_t em suas operações genéticas. Em adição à definição de ajuste para os indivíduos da população corrente, o ajuste também é definido para os indivíduos da população externa. De fato, o SPEA define um ajuste (chamado resistência ou strength) S_i para cada membro i da primeira população externa. A resistência S_i é proporcional ao número (n_i) de indivíduos na população corrente que são dominados pela solução externa i :

$$S_i = \frac{n_i}{N + 1} \quad (4.11)$$

Em outras palavras, a equação acima designa mais resistência (strength) para uma elite que domina mais soluções na população corrente. A divisão por $(N + 1)$ assegura que o valor máximo da resistência de qualquer membro da população externa nunca será maior ou igual a 1 . Portanto, uma solução não-dominada que domina outras poucas soluções terá um menor (ou melhor) ajuste.

Portanto, o ajuste de um indivíduo j da população corrente é calculado como o valor 1 mais a soma dos valores de resistência de todos os indivíduos da população externa que fracamente dominam j :

$$F_j = 1 + \sum_{i \in \bar{P}_t, i \prec j} S_i \quad (4.12)$$

Na equação 4.12, o termo 1 na adição permite que o ajuste de qualquer indivíduo j da população corrente P_t seja maior que o ajuste de qualquer indivíduo da população externa \bar{P}_t . Esta condição sugere que uma solução com menor ajuste é melhor. A figura 4.10 mostra o esquema de definição de ajuste para ambas as populações correntes (mostrado por quadrados) e

externa (mostrada por círculos) para um problema de minimização de dois objetivos. Os valores de ajuste também estão mostrados na figura. Estes valores mostram que os indivíduos da população externa têm menores valores de ajuste que os indivíduos da população corrente. Uma vez que existem seis indivíduos na população corrente, o denominador dos valores de ajuste será sete. Indivíduos da população corrente que estão dominados por muitos indivíduos da população externa terão maiores ajustes. Com estes ajustes, um procedimento de seleção por torneio binário é aplicado para combinar as populações ($\bar{\mathbf{P}}_t \dot{\cup} \mathbf{P}_t$) a fim de escolher soluções com menores valores de ajuste. Assim, provavelmente as elites externas serão enfatizadas durante o procedimento de seleção por torneio. Como usual, os operadores de cruzamento e mutação são aplicados na população combinada e uma nova população \mathbf{P}_{t+1} de tamanho N é criada.

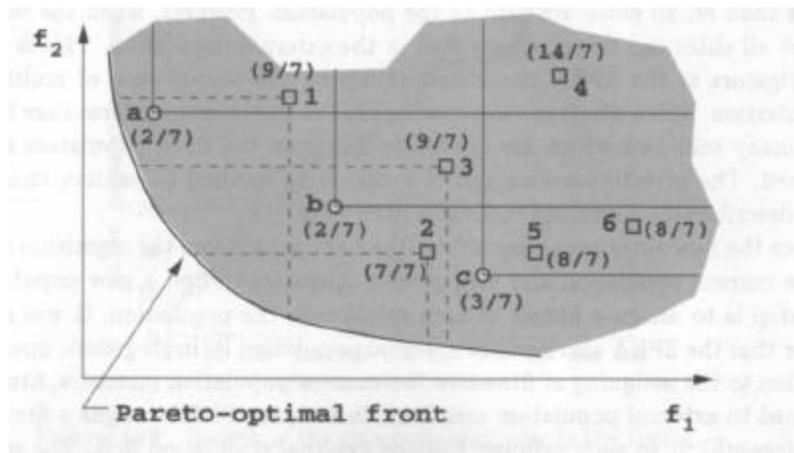


Figura 4.10 – Esquema de Ajuste do SPEA

4.12.2 Procedimento de ajuste do SPEA

Seguindo, descreve-se uma iteração do algoritmo passo-a-passo. Inicialmente, assume-se $\bar{\mathbf{P}}_t = \emptyset$

Passo 1 Achar a melhor série não-dominada $\mathbf{F}_1(\mathbf{P}_t)$. Copiar estas soluções para $\bar{\mathbf{P}}_t$, ou executar $\bar{\mathbf{P}}_t = \bar{\mathbf{P}}_t \dot{\cup} \mathbf{F}_1(\mathbf{P}_t)$

- Passo 2* Achar as melhores soluções não-dominadas $F_1(\overline{\mathbf{P}}_t)$ da população modificada $\overline{\mathbf{P}}_t$ e excluir todas as soluções dominadas, ou executar $\overline{\mathbf{P}}_t = F_1(\overline{\mathbf{P}}_t)$
- Passo 3* Se $|\overline{\mathbf{P}}_t| \tilde{n} \overline{\mathbf{N}}$, usar uma técnica de agrupamento para reduzir o tamanho de $\overline{\mathbf{N}}$. De outro modo, manter $\overline{\mathbf{P}}_t$ sem alterações. A população resultante é a população externa $\overline{\mathbf{P}}_{t+1}$ da próxima geração.
- Passo 4* Definir um ajuste para cada solução elite $i \hat{\mathbf{I}} \overline{\mathbf{P}}_{t+1}$ usando a equação 4.11. Assim, definir o ajuste para cada indivíduo $j \hat{\mathbf{I}} \mathbf{P}_t$ usando a equação 4.12.
- Passo 5* Aplicar uma seleção por torneio binário com estes valores de ajuste (no sentido de minimização) e aplicar os operadores de cruzamento e mutação a fim de criar a nova população \mathbf{P}_{t+1} de tamanho \mathbf{N} a partir da população combinada $(\overline{\mathbf{P}}_{t+1} \hat{\cup} \mathbf{P}_t)$ de tamanho $(\overline{\mathbf{N}} + \mathbf{N})$

Os passos 3 e 5 resultam em novas populações externas e correntes, as quais são processadas na próxima geração. Este algoritmo continua até que um critério de parada seja satisfeito.

4.12.3 Algoritmo de Agrupamento

Vamos agora descrever o algoritmo de agrupamento que reduz o tamanho da população externa $\overline{\mathbf{P}}_t$ de tamanho $\overline{\mathbf{N}}'$ para $\overline{\mathbf{N}}$ (onde $\overline{\mathbf{N}}' > \overline{\mathbf{N}}$).

Primeiramente, cada solução em $\overline{\mathbf{P}}_t$ é considerada residente em um grupo separado. Assim, inicialmente existem $\overline{\mathbf{N}}'$ grupos. Assim, as distâncias entre grupos são calculadas entre todos os

pares de grupos. Em geral, a distância d_{12} entre dois grupos C_1 e C_2 é definida como a média Euclidiana da distância de todos os pares de soluções ($i \in C_1$ e $j \in C_2$), ou matematicamente:

$$d_{12} = \frac{1}{|C_1||C_2|} \sum_{i \in C_1, j \in C_2} d(i, j) \quad (4.13)$$

A distância $d(i, j)$ pode ser calculada no espaço de decisão ou no espaço objetivo. Os autores do SPEA propuseram usar o espaço objetivo. Uma vez que todas as distâncias de grupos são computadas, os dois grupos que tiveram as mínimas distâncias de grupo são combinados para formarem um grupo maior. Assim, as distâncias de grupos são recalculadas para todos os pares de grupos e os dois grupos mais próximos são unidos. Este processo de união de grupos continua até que o número de grupos na população externa seja reduzido a \bar{N} . Portanto, em cada grupo, a solução com a distância média mínima em relação a outro grupo é mantida e todas as outras são excluídas.

4.12.4 Procedimento Agrupamento

Passo C1 Inicialmente, cada solução pertence a um grupo distinto $C_i = \{i\}$, formando $C = \{C_1, C_2, \dots, C_{\bar{N}}\}$

Passo C2 Se $|C| \leq \bar{N}$, ir para o passo C5. De outro modo ir para passo C3

Passo C3 Para cada par de grupo (existem $\frac{|C|}{2}$ deles), calcular as distâncias de grupo pela equação 4.13. Achar o par (i_1, i_2) que corresponde ao valor de distância mínima.

Passo C4 Unir os dois grupos C_{i1} e C_{i2} . Isto reduz o tamanho de C reduzindo um grupo. Voltar ao passo C2

Passo C5 Escolher somente uma solução de cada grupo e excluir as outras soluções dos grupos. A solução que tem a distância média mínima em relação às demais soluções no grupo pode ser escolhida como a solução que representa o grupo.

Uma vez alcançado o passo C5, todas menos uma solução representativa são mantidas e a série resultante tem no máximo \bar{N} soluções.

4.12.5 Complexidade Computacional

O algoritmo acima descrito representa um algoritmo de fácil implementação, porém demanda um grande capacidade computacional $O(MN^3)$. Entretanto, o algoritmo de agrupamento pode ser executado com uma codificação adequada e as computações podem ser reduzidas. Uma vez que as soluções não são removidas até o passo C5, todas as distâncias computacionais podem ser, na prática, executadas uma vez no início. Isto requer $O(MN^2)$ computações. Portanto, os cálculos das distâncias médias dos grupos, o cálculo das distâncias mínimas de todos os pares de grupos (com um procedimento de registro complexo) e a união dos grupos podem ser feitos através de um código especial em um tempo linear entre os passos C2 e C4. Uma vez que são chamados $(\bar{N}' - \bar{N})$ vezes, a complexidade total é $O(N'^2)$. A remoção final de soluções extras com computação da distância intra-grupo é \bar{N}' . Assim, se implementado com cuidado, a complexidade do algoritmo de agrupamento pode ser reduzido para $O(MN'^2)$.

4.12.6 Vantagens

É fácil perceber que uma vez que uma solução é encontrada na fronteira ótima de Pareto, esta é imediatamente armazenada na população externa. O único caminho pelo qual esta solução pode ser eliminada é quando outra solução Pareto-Ótima que gerará uma melhor dispersão seja

descoberta. O algoritmo de agrupamento assegura que uma melhor dispersão é alcançada entre as soluções não-dominadas encontradas.

Este algoritmo de agrupamento é não-paramétrico, portanto tornando-se atrativo. O procedimento de definição do ajuste no SPEA é mais ou menos similar ao procedimento do MOGA e é relativamente simples de calcular. Para fazer o procedimento de agrupamento encontrar o máximo de soluções diversas cada solução extrema pode ser forçada a permanecer em um grupo independente.

4.12.7 Desvantagens

O SPEA introduz um parâmetro extra \bar{N} , o tamanho da população externa. Um balanceamento entre o tamanho N da população regular e o tamanho da população externa \bar{N} é importante para o sucesso do SPEA. Se uma grande população externa é usada (em comparação com a população regular), a pressão de seleção para as elites será grande e o SPEA pode não convergir para uma fronteira Ótima de Pareto. Por um outro lado, se uma população externa pequena for usada, o efeito do elitismo pode ser desprezível. Além do mais, muitas soluções na população não serão dominadas por qualquer membro da população externa e os ajustes gerados serão idênticos. Os pesquisadores do SPEA recomendam a utilização de uma relação 1:4 entre a população externa e a população corrente.

O SPEA divide um problema comum com o MOGA. Uma vez que a ordenação não-dominada da população inteira não é usada para se definir o ajuste, este ajuste pode não favorecer igualmente todas as soluções não-dominadas de mesmo ranking. Esta influência na determinação do ajuste de soluções da mesma fronteira é dependente da população e da densidade de soluções no espaço de procura.

No procedimento de ajuste do SPEA, uma solução externa que domina mais soluções terá um pior ajuste. Esta classificação é justificada quando todas as soluções não-dominadas são concentradas perto da solução dominante. Uma vez que na maioria dos casos isto não é verdade o efeito de agrupamento poderia vir somente do algoritmo de agrupamento. Por outro lado, esta

classificação de ajuste pode dar uma pressão de seleção equivocada para soluções não dominadas.

4.13 Comentários Finais

Neste capítulo foram apresentados os Algoritmos Genéticos como técnicas de otimização multi-objetivos. Foram detalhados os principais modelos de algoritmos genéticos para estes problemas. Dentro de cada algoritmo existem algumas métricas para avaliar o desempenho e vantagens e desvantagens foram mostradas. Todos estes conceitos, vistos nos capítulos 3 e 4, serão usados para formular e aplicar estas técnicas em um problema de Otimização de Componentes Mecânicos, as quais serão apresentadas nos próximos dois capítulos.

Capítulo 5

Simulações Teóricas e seus Resultados

Neste capítulo pretende-se verificar a aplicação simplificada através dos métodos de planejamento experimental aplicado na forma de DOE (delineamento de experimentos), metodologia de superfície de resposta e otimização multi-objetivos através de algoritmos genéticos que foram propostos nos capítulos 3 e 4, para um sistema dinâmico rotor-mancal.

As simulações teóricas foram executadas para uma configuração de um sistema mecânico composto de um rotor fixado a um eixo flexível e apoiado em dois mancais hidrodinâmicos. O programa ROTORTEST[®] desenvolvido no LAMAR, laboratório de máquinas rotativas da UNICAMP, foi usado para simular este sistema. O ROTORTEST[®] é um programa de gerenciamento, análise e simulação de rotores horizontais, capaz de montar os modelos matemáticos, e obter as soluções de resposta em frequência dos rotores durante o projeto e funcionamento. O código do programa foi implementado inicialmente em linguagem Fortran, tendo interface com usuário através da linguagem VB NET[®]. Isso resulta em um programa de interface interativa com o usuário (*user friendly*).

O estudo procura obter os parâmetros estimados para o modelo matemático, obtendo possíveis ajustes para a modelagem e projeto de componentes de máquinas rotativas tais como

mancais hidrodinâmicos, rotores de turbinas a gás ou vapor, rotores de bombas, acoplamentos elásticos, volantes de inércia, etc.

Usualmente, na modelagem de máquinas rotativas, rotores de bombas, compressores e turbinas são modelados como discos rígidos. Além disso, por se tratar de um disco de espessura considerável, considerou-se o eixo com maior diâmetro externo ao longo desta extensão, inserindo, desta forma, sua influência na flexibilidade do eixo. O restante da massa do rotor foi concentrado em dois nós sucessivos do modelo, conforme Figura 5.2. Desta maneira, o eixo foi modelado por elementos de viga, com as forças agindo no plano normal ao eixo, e localizadas nos centros de massa dos discos rotativos.

Entre as preocupações em se estudar e projetar máquinas rotativas, estão o monitoramento e a determinação de picos excessivos de amplitudes para determinadas frequências, além das frequências próprias de ressonância que podem causar falhas em componentes mecânicos. A análise das vibrações em máquinas rotativas pode ocorrer de duas formas, uma pelo estudo das vibrações em estruturas e a outra na dinâmica de rotores. A análise do comportamento dinâmico de máquinas rotativas horizontais (turbogeradores) envolve certa complexidade, sendo os sistemas geralmente constituídos por rotores, eixos, mancais, acoplamentos, carcaça, fundação e outros equipamentos auxiliares.

A Figura 5.1 mostra um modelo simplificado de rotor, contendo um eixo flexível e um disco rígido.

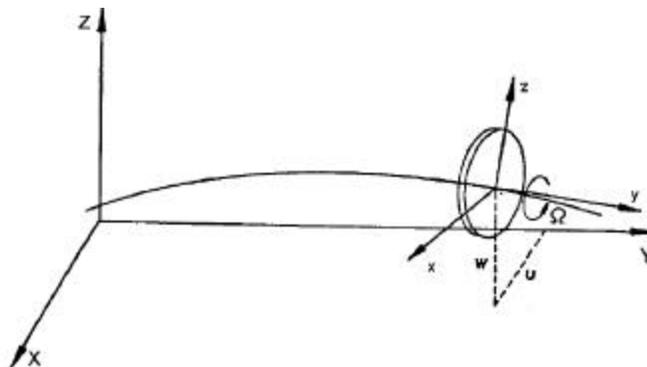


Figura 5.1 – Simplificação de modelo de um rotor.

As equações de movimento dos componentes do sistema mecânico podem ser deduzidas aplicando-se a equação de Lagrange definida pela equação 5.1, nas expressões das energias cinéticas (E_c), e potenciais (E_p) dos componentes.

$$\frac{d}{dt} \left(\frac{\partial E_c}{\partial \dot{q}_i} \right) - \frac{\partial E_c}{\partial q_i} + \frac{\partial U}{\partial q_i} = F_{q_i} \quad (5.1)$$

Sendo q_i é a i -ésima coordenada generalizada; F_{q_i} é a força generalizada atuando na direção da i -ésima coordenada generalizada.

A equação de movimento do rotor pode ser descrita pela equação:

$$[M]\ddot{q}(t) + [C]\dot{q}(t) + [K]q(t) = \{f_e\} - \{p\} \quad (5.2)$$

onde, $[M]$ é a matriz de massa do eixo e dos elementos a ele associados;

$[C]$ é a matriz de amortecimento do eixo e dos mancais, incluindo a matriz giroscópica;

$[K]$ é a matriz de rigidez do eixo;

$\{f_e\}$ é o vetor de forças externas;

$\{p\}$ é a força peso do disco que compõe o rotor;

$\{\ddot{q}\}$ é o vetor de aceleração em relação ao centro do eixo;

$\{\dot{q}\}$ é o vetor de velocidade em relação ao centro do eixo;

$\{q\}$ é o vetor de deslocamento em relação ao centro do eixo.

As forças externas $\{f_e\}$ podem ser posicionadas sobre o disco rígido, como forças de desbalanceamento do rotor devido à presença de excentricidade da massa. As forças que surgem no sistema devido ao filme de óleo são introduzidas na matriz de amortecimento $[C]$. A matriz de amortecimento $[C]$ é proporcional à matriz de rigidez $[K]$ e inclui um amortecimento estrutural de 2%, introduzido através de coeficientes equivalentes de rigidez e amortecimento, dependentes da posição e da velocidade do rotor, em relação às coordenadas do centro do mancal.

A figura 5.2 apresenta um exemplo de conjunto eixo-mancal no ROTORTEST®.

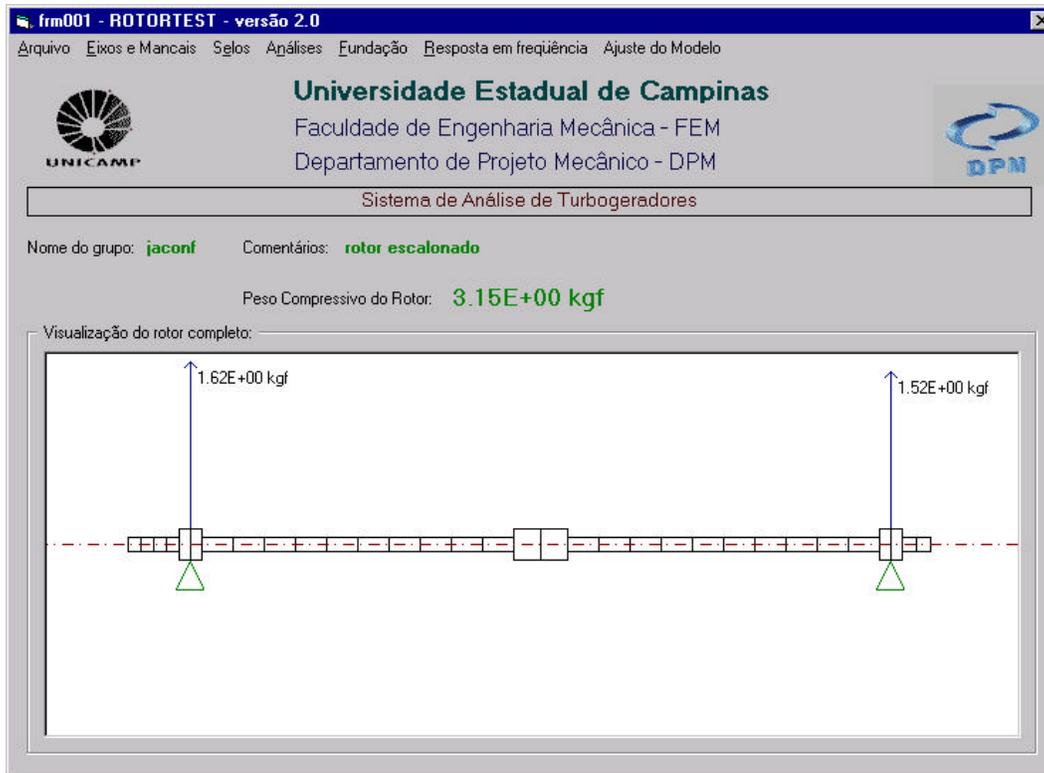


Figura 5.2 – Conjunto eixo-mancal-massa do programa ROTORTEST®.

No ROTORTEST® é possível a entrada de dados relativos ao sistema. O tipo de mancal é cilíndrico, as folgas radiais são 180 e 250 μm , um lóbulo por mancal, as temperaturas do lubrificante são 30°C e 40°C, a largura dos mancais é 20 mm e o diâmetro de 31 mm. Os comprimentos dos eixos são 460 mm e 600 mm com diâmetros de 12 mm. O disco rígido tem massa de 2,35 kg, espessura de 43 mm e diâmetro externo de 94,82 mm. Neste passo, o programa calcula os coeficientes diretos e acoplados de rigidez e amortecimento do filme de óleo. Então, tais termos são inseridos nas matrizes de rigidez e amortecimento montadas para o eixo. Com os dados fornecidos e os calculados, é possível realizar o cálculo da resposta do rotor, a partir de uma velocidade de rotação inicial até uma final, com um determinado passo de velocidade, sendo que todos estes parâmetros são definidos pelo usuário do programa. É possível também a realização de uma avaliação de estabilidade do sistema em todas estas velocidades. Também é

definido o vetor de carregamento atuante no sistema. Com o sistema global montado, a solução dinâmica do sistema pode ser obtida.

Para o modelo dos mancais hidrodinâmicos, a equação de Reynolds, desenvolvida por Osborne Reynolds é a base da teoria de lubrificação para mancais hidrodinâmicos. Várias soluções podem ser obtidas aplicadas a casos especiais de interesse, onde serão formuladas com as hipóteses necessárias a cada caso. Quando aplicada no estudo de filmes de óleo em mancais hidrodinâmicos, a solução desta equação fornece a distribuição de pressão no filme de óleo, a qual é necessária para o cálculo da maioria dos problemas básicos. A criação de um sistema equivalente entre o filme de óleo e um conjunto de molas e amortecedores produz uma série de equações lineares que facilitam a solução dos problemas de mancais hidrodinâmicos.

5.1 Delineamento de Experimentos (DOE) e Superfície de Resposta (MSR)

Foi executado um planejamento de experimentos simples para a obtenção dos resultados de amplitude e frequência para os valores inferiores e superiores dos seguintes parâmetros: distância entre mancais (comprimento do eixo), folga radial e temperatura do óleo.

Ensaio Ordem do DOE	Distância entre Mancais – x_1	Folga Radial x_2	Temperatura do óleo – x_3
1	–	–	–
2	+	–	–
3	–	+	–
4	+	+	–
5	–	+	+
6	+	–	+
7	–	+	+
8	+	+	+

Tabela 5.1 – DOE – Ordem de ensaios

Os valores dos parâmetros são mostrados na tabela 5.2.

Parâmetro	Variáveis	Limite Inferior (-)	Limite Superior (+)
Distancia entre Mancais	x_1	460 mm	600 mm
Folga Radial	x_2	180 μm	250 μm
Temperatura do Óleo	x_3	30°C	40°C

Tabela 5.2 – Parâmetros para simulação teórica

A ordem do DOE define a seqüência que as variáveis devem ser introduzidas na análise de superfície de resposta. A tabela 5.3 e as figuras 5.3a até 5.3h mostram os resultados de acordo com as simulações executadas com auxílio do programa ROTORTEST[®], onde é possível visualizar as curvas estimadas de amplitude de vibração no domínio da freqüência. O programa ROTORTEST[®] gera curvas diferentes para vertical e horizontal, baseadas no diagrama de pressões hidrodinâmicas dos mancais.

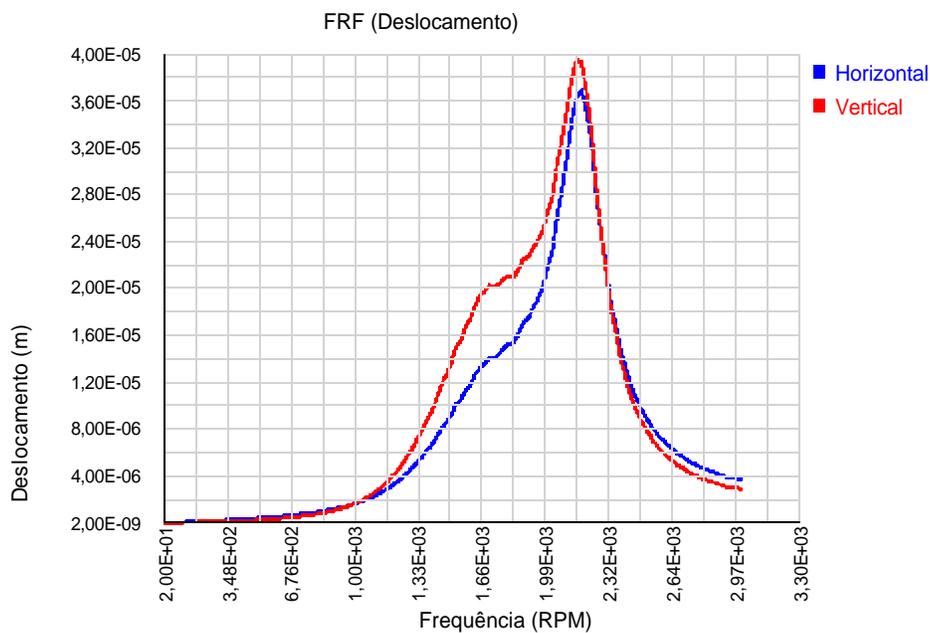


Fig. 5.3a – Simulação 1 ($x_1 = 460\text{mm}$, $x_2 = 180\mu\text{m}$, $x_3 = 30^\circ\text{C}$)

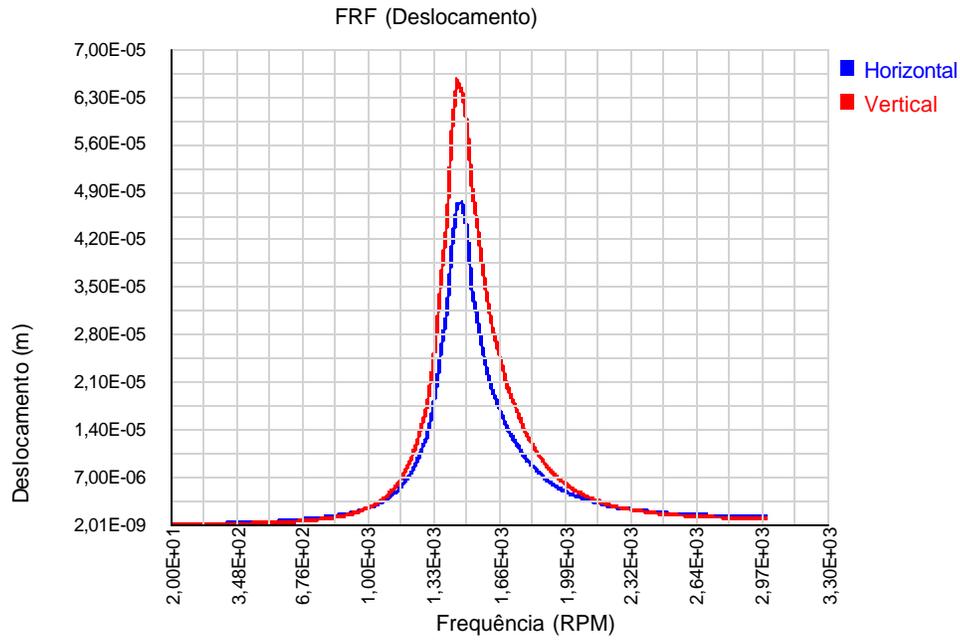


Fig. 5.3b – Simulação 2 ($x_1 = 600\text{mm}$, $x_2 = 180\mu\text{m}$, $x_3 = 30^\circ\text{C}$)

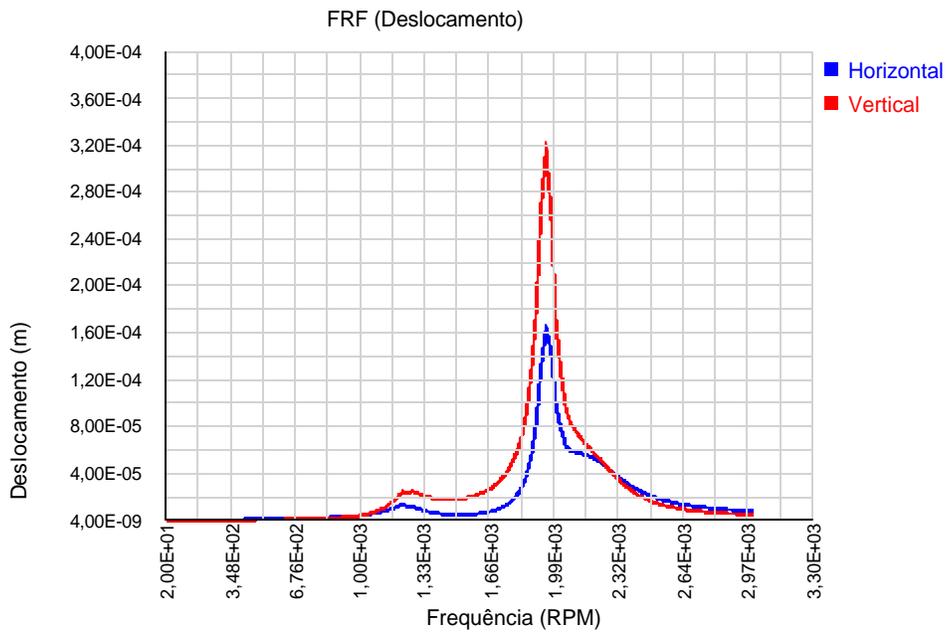


Fig. 5.3c – Simulação 3 ($x_1 = 460\text{mm}$, $x_2 = 250\mu\text{m}$, $x_3 = 30^\circ\text{C}$)

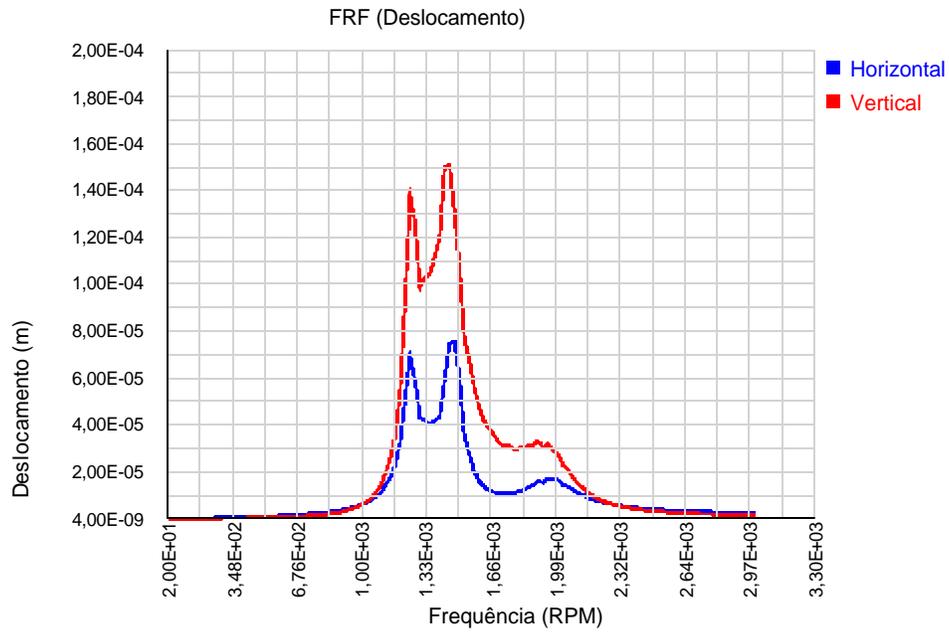


Fig. 5.3d – Simulação 4 ($x_1 = 600\text{mm}$, $x_2 = 250\mu\text{m}$, $x_3 = 30^\circ\text{C}$)

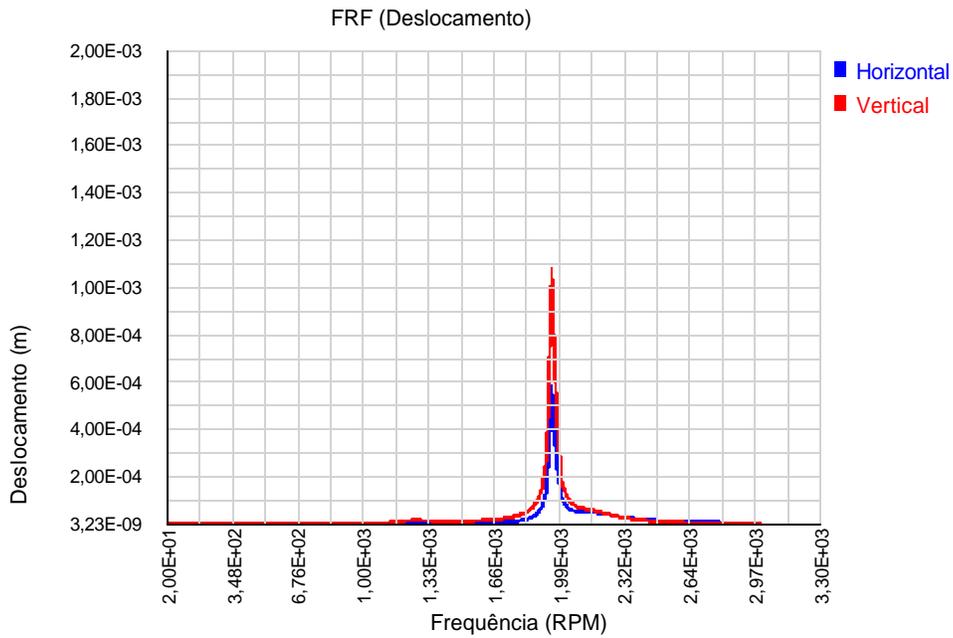


Fig. 5.3e – Simulação 5 ($x_1 = 460\text{mm}$, $x_2 = 250\mu\text{m}$, $x_3 = 40^\circ\text{C}$)

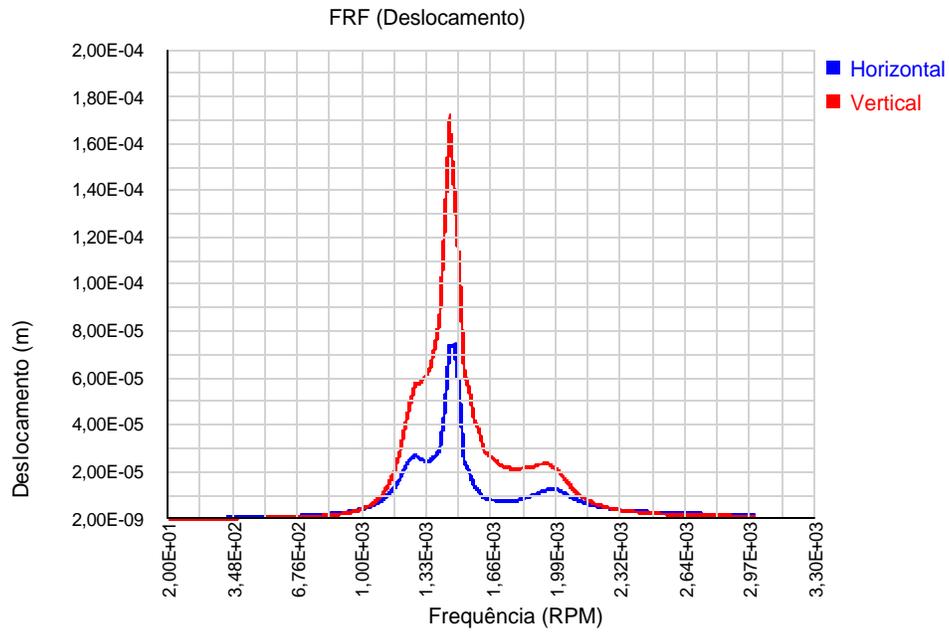


Fig. 5.3f – Simulação 6 ($x_1 = 600\text{mm}$, $x_2 = 180\mu\text{m}$, $x_3 = 40^\circ\text{C}$)

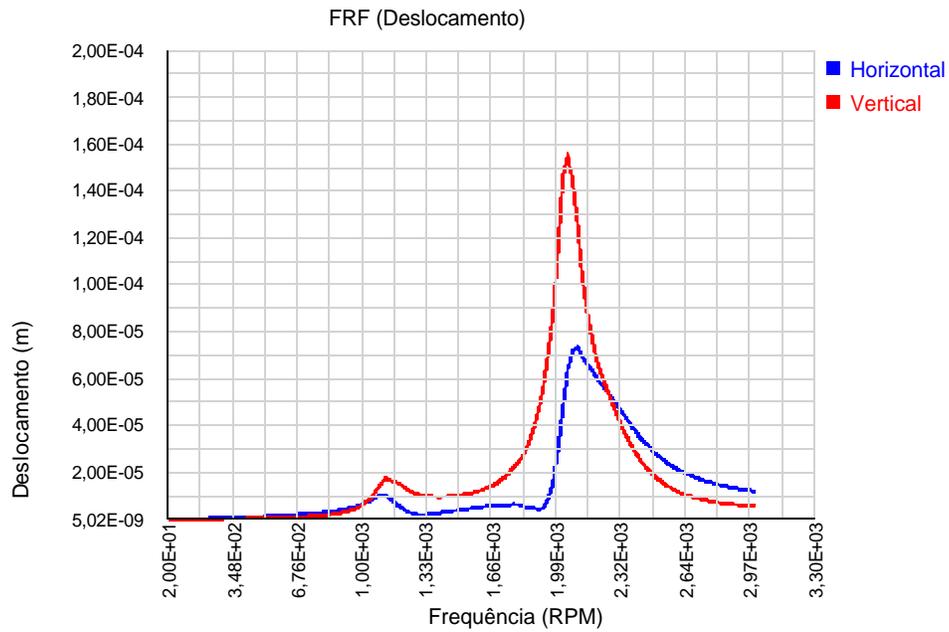


Fig. 5.3g – Simulação 7 ($x_1 = 460\text{mm}$, $x_2 = 250\mu\text{m}$, $x_3 = 40^\circ\text{C}$)

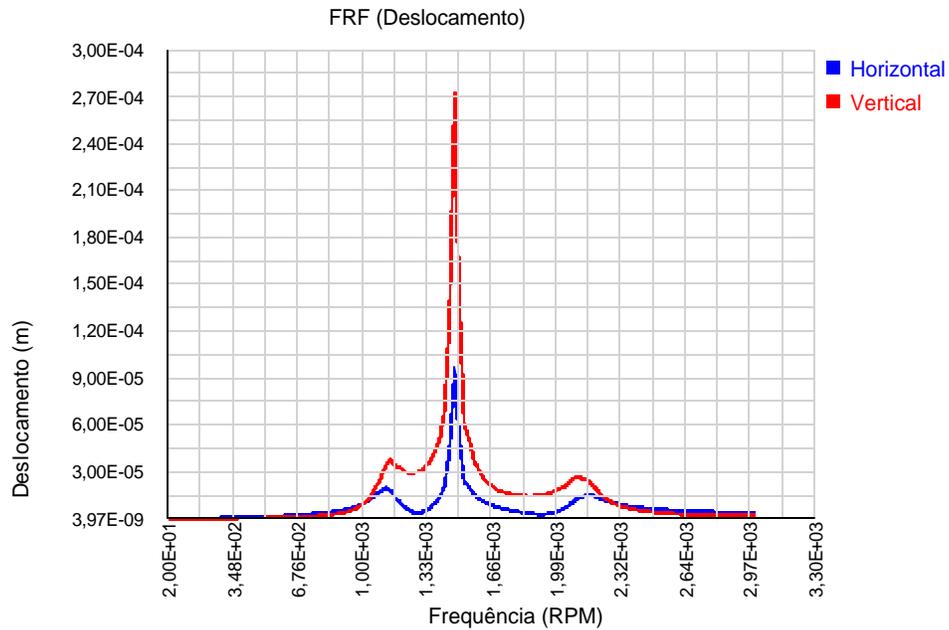


Fig. 5.3h – Simulação 8 ($x_1 = 600\text{mm}$, $x_2 = 250\mu\text{m}$, $x_3 = 40^\circ\text{C}$)

Para cada ordem de DOE foi obtida uma resposta do sistema (simulações 1 a 8). As amplitudes nas direções horizontal e vertical foram mostradas no domínio da frequência e os picos máximos estão na tabela 5.3.

Simulações	DOE	Rotação Crítica [rpm]	Amplitude Vertical [10^{-5} m]	Amplitude Horizontal [10^{-5} m]
1	--	2175	3,99	3,73
2	+--	1450	6,55	4,73
3	--+	1950	22,0	10,6
4	++-	1450	15,1	7,48
5	++	1950	83,6	40,7
6	+++	1450	19,2	8,48
7	+++	2050	13,4	5,39
8	+++	1475	26,7	9,17

Tabela 5.3 – Valores das Simulações Teóricas.

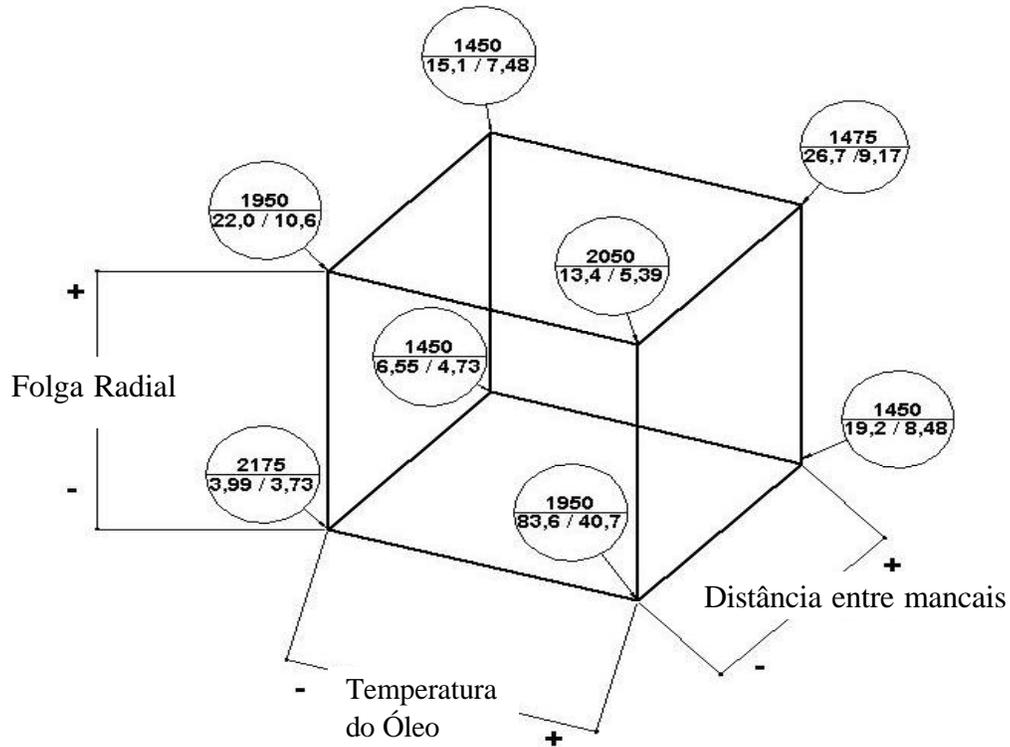


Fig. 5.4 – Resultados do DOE mostrado no cubo dimensional.

As amplitudes foram linearmente combinadas gerando apenas uma resposta de amplitude média no domínio da frequência, sendo que esta simplificação não afeta os resultados esperados deste trabalho. Pode-se mostrar os efeitos de cada simulação na forma de um cubo contendo os valores das simulações nos vértices. Em cada círculo mostrado estão os valores da rotação crítica e das amplitudes. A figura 5.4 mostra o cubo dimensional.

5.2 Codificação das variáveis

Os parâmetros são transformados em variáveis codificadas que serão utilizadas na metodologia de superfície de resposta:

5.2.1 Distância entre mancais L

Calculamos a distância média:

$$\bar{L} = \frac{460 + 600}{2} = 530 \quad (5.3)$$

E a variação:

$$DL = \frac{600 - 460}{2} = 70 \quad (5.4)$$

e definimos a variável x_1 :

$$x_1 = \frac{L - \bar{L}}{DL} = \frac{L - 530}{70} \quad \text{onde } x_1 \in [-1 \dots 1] \quad (5.5)$$

5.2.2 Folga Radial F

Da mesma maneira definimos a variável x_2 :

$$x_2 = \frac{F - \bar{F}}{DF} = \frac{F - 215}{35} \quad \text{onde } x_2 \in [-1 \dots 1] \quad (5.6)$$

5.2.3 Temperatura do Óleo T

Da mesma maneira definimos a variável x_3 :

$$x_3 = \frac{T - \bar{T}}{DT} = \frac{T - 35}{5} \quad \text{onde } x_3 \in [-1 \dots 1] \quad (5.7)$$

5.3 Funções Polinomiais

Vamos ajustar as variáveis x_1 , x_2 e x_3 por um modelo bi-linear:

$$f = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_{12}x_1x_2 + a_{13}x_1x_3 + a_{23}x_2x_3 + a_{123}x_1x_2x_3 \quad (5.8)$$

Utilizando o método dos mínimos quadrados, determinamos os valores dos parâmetros (α_s). Estes cálculos são efetuados diretamente no programa ÓTIMA®, figura 5.5, desenvolvido por Silveira (2004).

Dados de Entrada

Nº de parâmetros (U.I.): 3
 Nº de níveis: 2
 Nº de réplicas: 2
 Nº de respostas: 1

Opções de visualização

Visualizar valores: reais
 Ordenar valores pela coluna: Nº do experimento

Níveis dos Parâmetros

Nome do Parâmetro	Baixo	Médio	Alto
distância	460		600
folga	180		250
temperatura	30		40

nº exp.	l	distância	folga	temperatura	R 1	R 2
1	1	460	180	30	3,99	3,73
2	1	600	180	30	6,55	4,73
3	1	460	250	30	22,0	10,6
4	1	600	250	30	15,1	7,48
5	1	460	180	40	83,6	40,7
6	1	600	180	40	19,2	8,48
7	1	460	250	40	13,4	5,39
8	1	600	250	40	26,7	9,17

Importante: separador de casas decimais (. ou .) configurado no sistema (painel de controle).

Fig. 5.5 – Entrada de dados no Programa ÓTIMA®.

O programa ÓTIMA® determina os coeficientes das funções polinomiais a serem posteriormente otimizadas. Estes coeficientes normalmente são sujeitos a análises estatísticas que não serão aplicadas neste trabalho.

Tomando-se os coeficientes lineares foram geradas duas funções polinomiais que serão utilizados como funções objetivos.

Para a amplitude (10^{-5} m) teremos a seguinte função polinomial:

$$f_1 = 17,55 - 5,38x_1 - 3,82x_2 + 8,281x_3 + 6,26x_1x_2 + \\ - 4,57x_1x_3 - 8,34x_2x_3 + 7,96x_1x_2x_3 \quad (5.9)$$

E para a frequência natural (rpm) teremos a seguinte função polinomial

$$f_2 = 1744 - 287,5x_1 - 12,5x_2 - 12,5x_3 + 18,75x_1x_2 + \\ + 18,75x_1x_3 + 43,58x_2x_3 - 37,5x_1x_2x_3 \quad (5.10)$$

Estas funções polinomiais representam as seguintes superfícies de resposta, mostradas nas figuras 5.6 a 5.9.

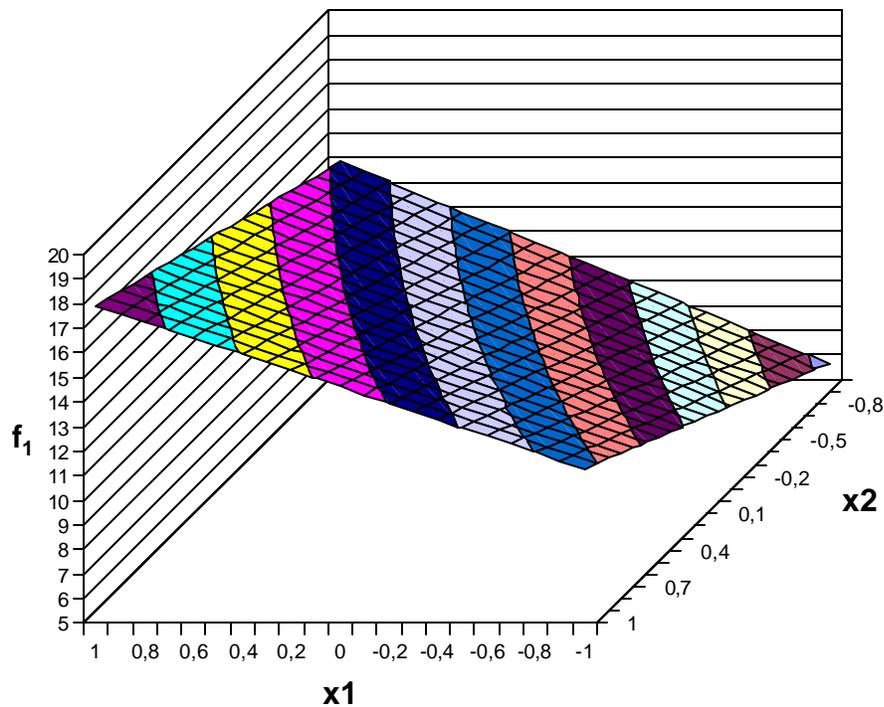


Fig. 5.6 – Superfície de Resposta f_1 em relação à x_1 e x_2

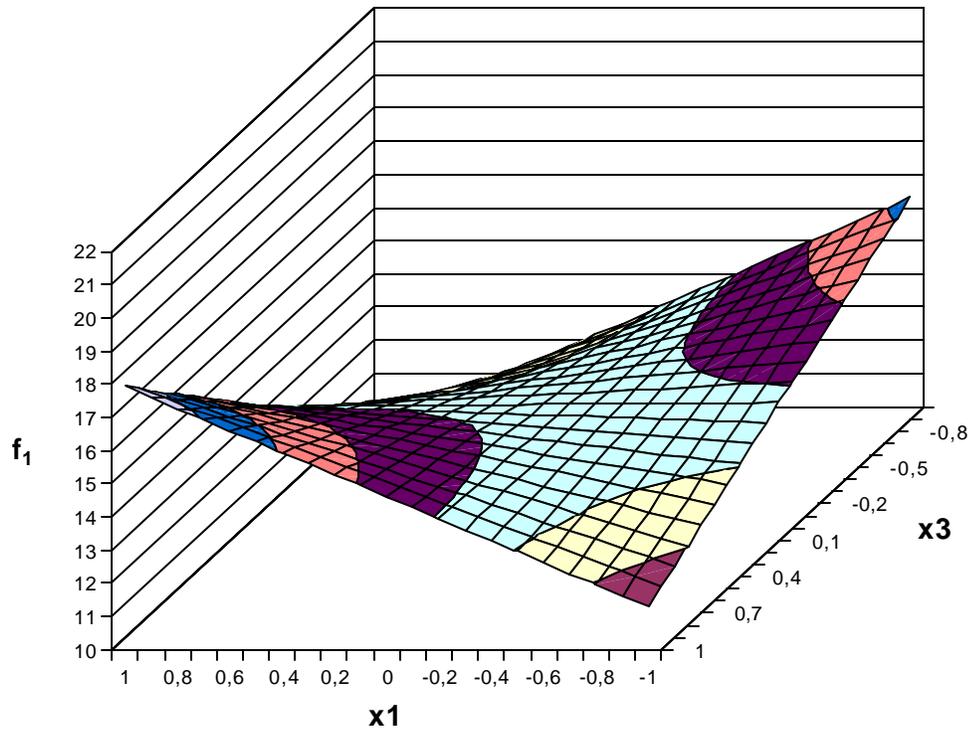


Fig. 5.7 – Superfície de Resposta f_1 em relação à x_1 e x_3

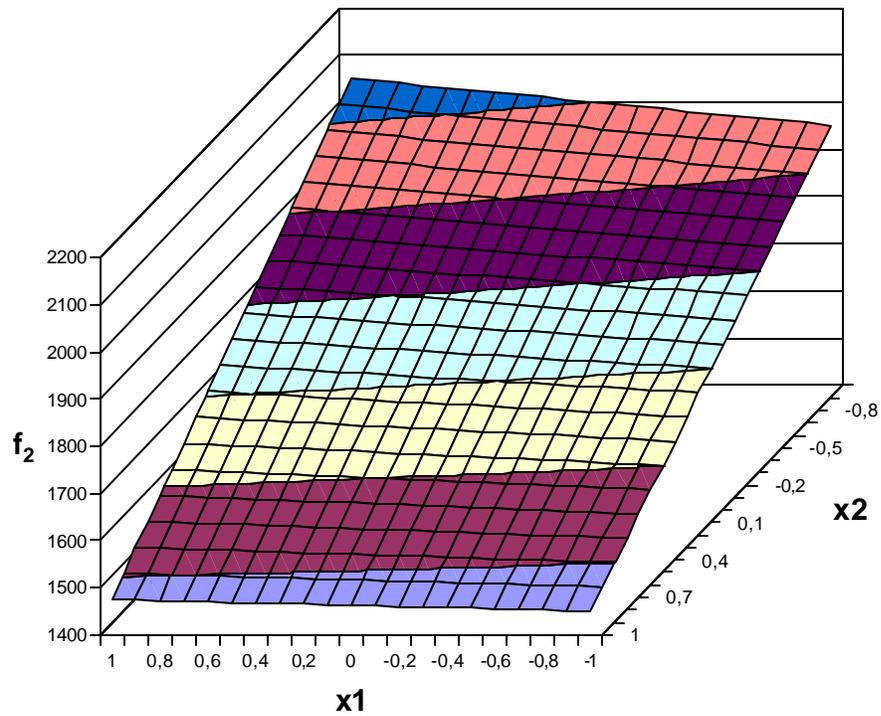


Fig. 5.8 – Superfície de Resposta f_2 em relação à x_1 e x_2

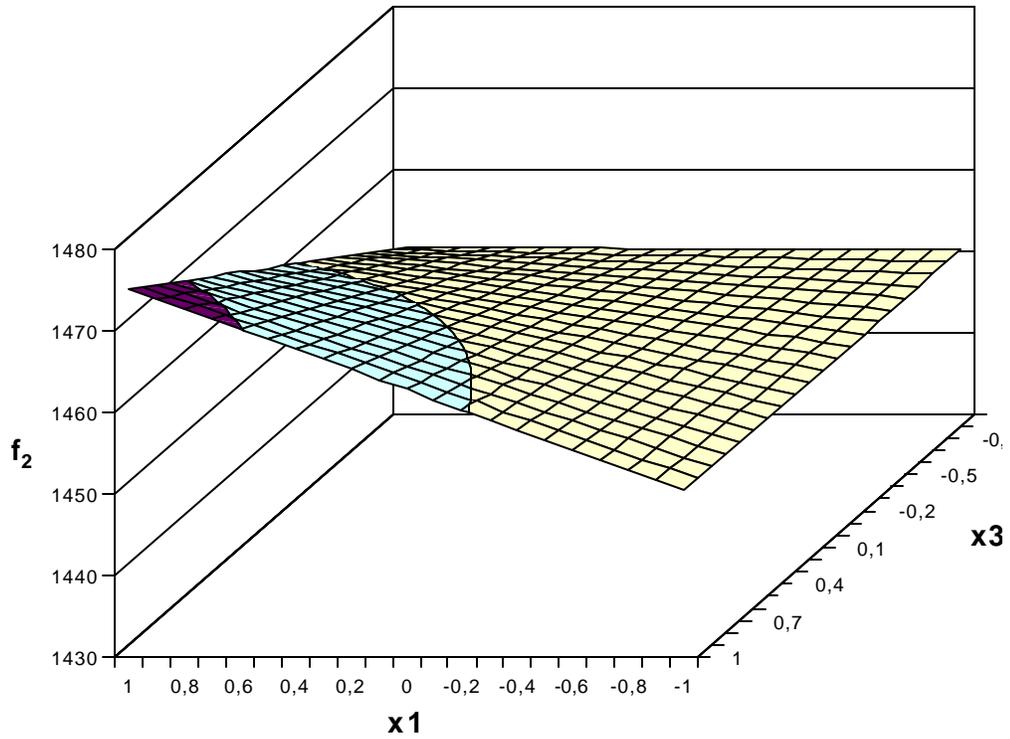


Fig. 5.9 – Superfície de Resposta f_2 em relação à x_1 e x_3

5.4 Aplicação do Algoritmo Genético Multi-Objetivos

Para a otimização multi-objetivos foram implementados quatro algoritmos genéticos, MOGA, NSGA, NSGA-II e SPEA. A figura 5.10 mostra o formulário geral de entrada dos dados principais.

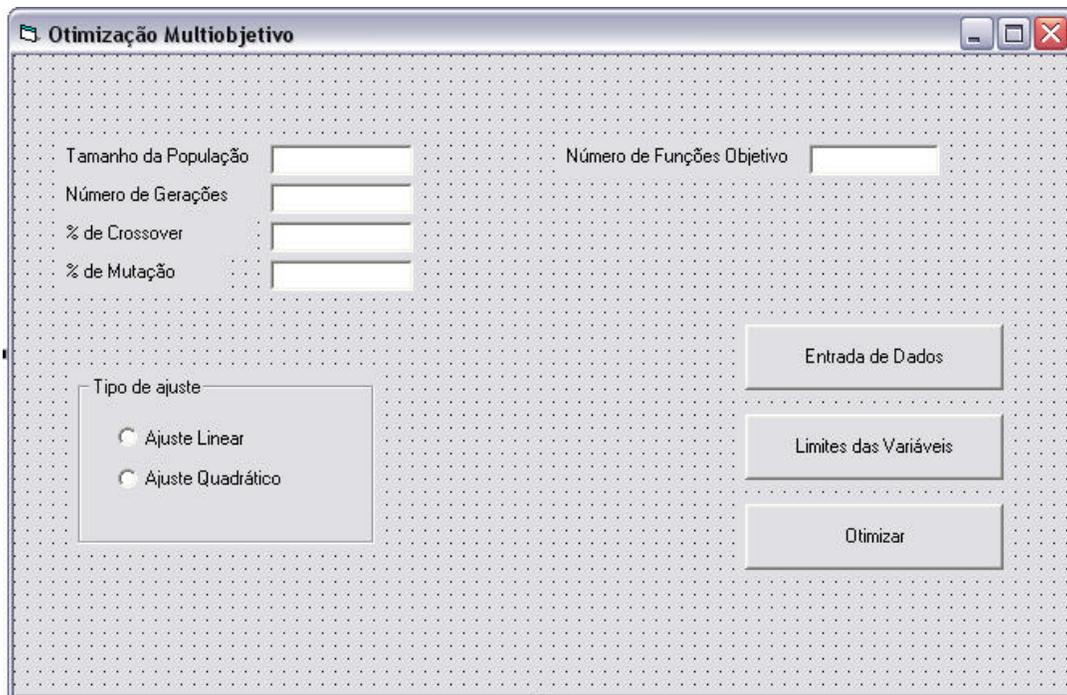


Fig 5.10 – Entrada de dados para a otimização multi-objetivos

Os campos “Tamanho da População”, “Número de Gerações”, “% de Crossover”, “% de Mutação” e “Número de Funções Objetivo” devem ser definidos onde:

- *Tamanho de população* refere-se ao número de indivíduos na análise. A quantidade de indivíduos representa a quantidade de soluções que serão encontradas. Deve-se escolher uma quantidade que atenda à convergência e apresente bons resultados;
- *Número de Gerações* é a quantidade de ciclos do programa principal. Este valor geralmente é elevado;
- *% de Crossover* é a probabilidade de cruzamento;
- *% de Mutação* é a probabilidade de mutação;
- *Número de funções objetivo* é a quantidade de funções a serem avaliadas.

5.4.1 Processo de Otimização Multi-Objetivos

$$\begin{array}{ll} \text{Minimizar} & f1(x1,x2,x3) \\ & f2(x1,x2,x3) \end{array} \quad (5.11)$$

$$\begin{array}{ll} \text{Sujeito a} & -1 \leq x1 \leq 1 \\ & -1 \leq x2 \leq 1 \\ & -1 \leq x3 \leq 1 \end{array}$$

Nesta otimização os seguintes parâmetros foram usados:

Tamanho da População: 500 indivíduos ou soluções

Número de Gerações: 1500

Probabilidade de Cruzamento: 2%

Probabilidade de Mutação: 0,2%

Número de Bits em cada variável: 8

As equações 5.9 e 5.10 formam o espaço de busca mostrado na figura 5.11. Os pontos mostrados representam a geração inicial gerada aleatoriamente. As soluções mostradas são soluções conflitantes que podem ser otimizados pelos algoritmos genéticos multi-objetivos.

Os resultados mostrados foram gerados pelo Algoritmo Genético MOGA, uma vez que os resultados dos algoritmos são similares e o MOGA é o mais rápido. O MOGA foi aplicado às funções objetivos $f1$ e $f2$, determinadas pelas equações 5.9 e 5.10 resultando na solução mostrada na figura 5.12.

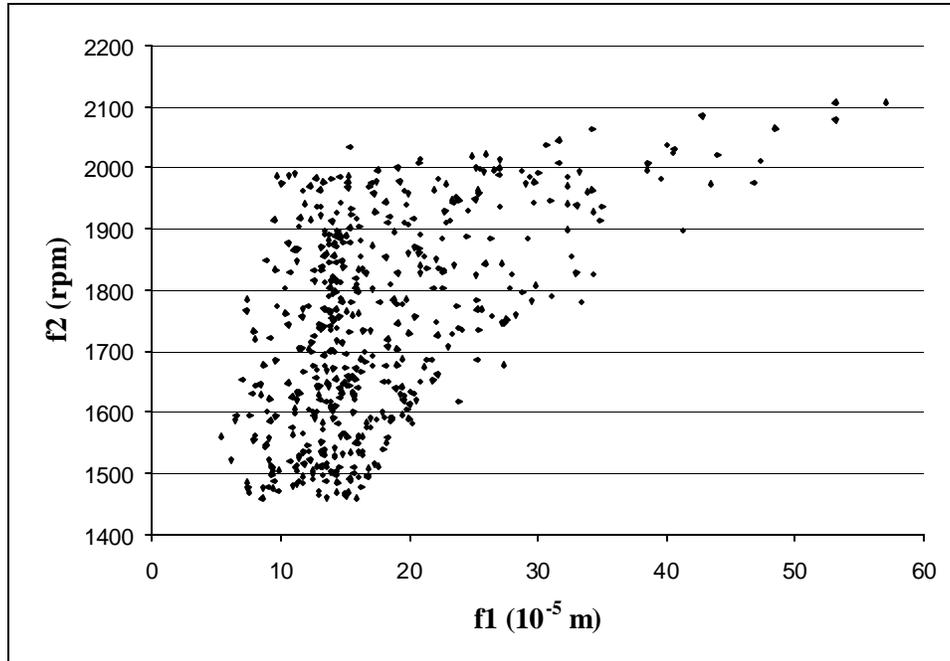


Fig. 5.11 – Geração Inicial

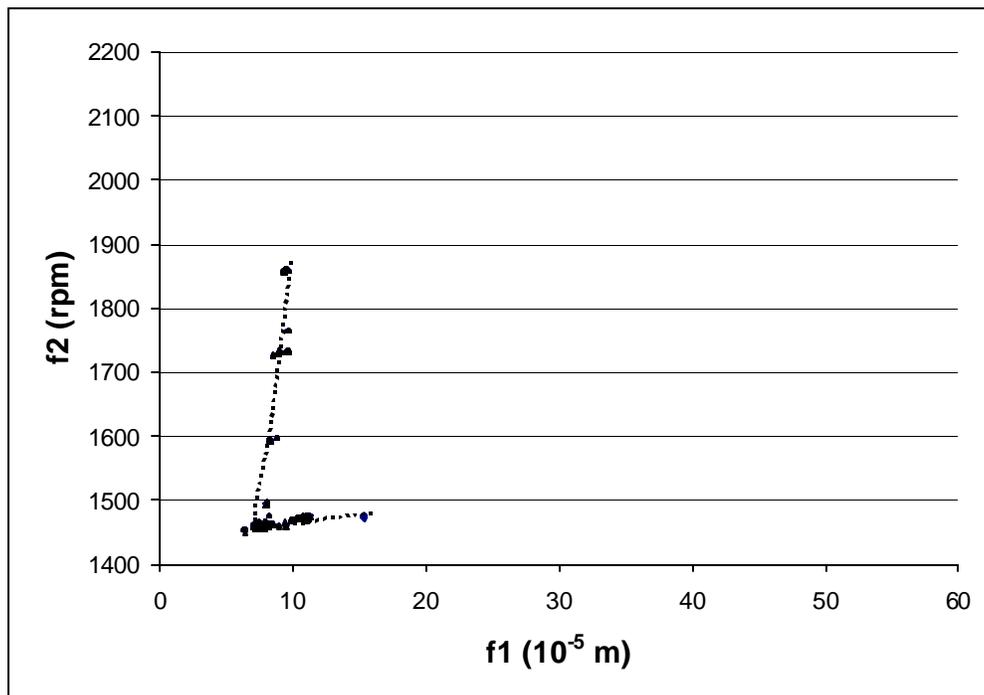


Fig. 5.12 – A linha tracejada representa a fronteira de Pareto.

5.5 Resultados

A Fronteira Pareto-Ótima está representada pela linha tracejada conforme mostrado na figura 5.12. Observa-se que as soluções otimizadas encontradas pelo algoritmo MOGA situam-se perto da Fronteira Pareto-Ótima. O algoritmo de otimização MOGA gerou 21 soluções ótimas, indicando que muitas estão sobrepostas, uma vez que temos uma população de 500 soluções.

A determinação da Fronteira Pareto-Ótima, mostrada pela linha tracejada na figura 5.12, também foi facilitada pelo prévio conhecimento do problema. A dispersão de valores, apesar de estar mostrada ao longo da linha tracejada, não indica realmente que se trata de linha contínua ou segmentos de reta.

Sabendo que para cada solução temos uma combinação de parâmetros x_1 , x_2 e x_3 , podemos determinar seus valores para cada solução ótima. As relações entre as soluções conflitantes estão mostradas nas figuras 5.13 a 5.16.

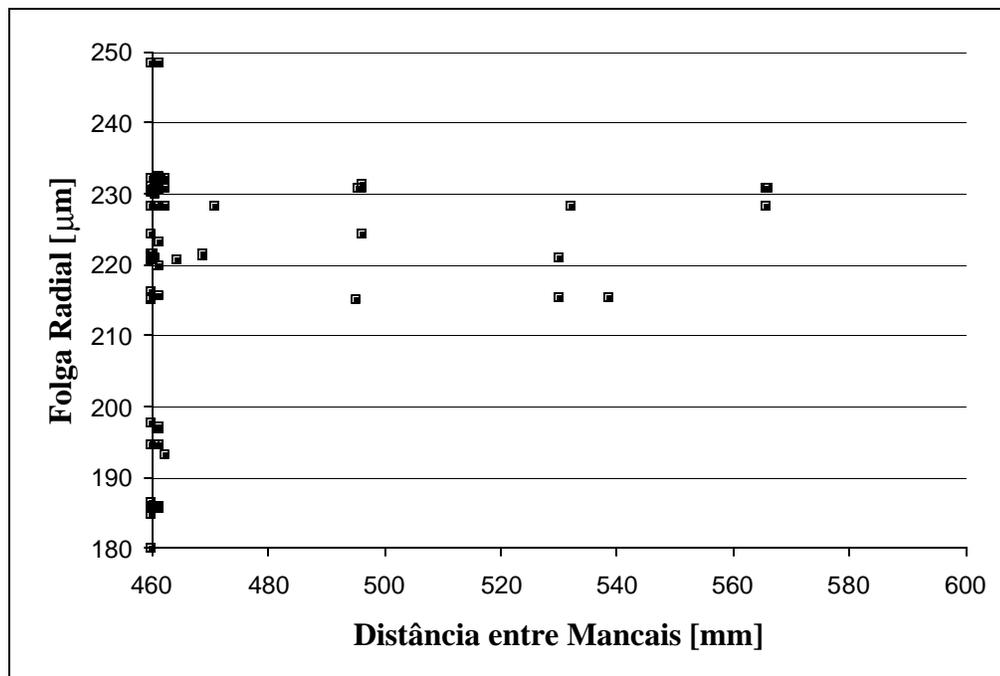


Fig. 5.13 – Relação entre Folga Radial e Distância entre Mancais.

A Figura 5.13 mostra os valores otimizados da Folga Radial em relação à Distância entre Mancais. Percebe-se que a maioria das folgas radiais está posicionada na distância de 460 mm e existe uma faixa entre 215 a 235 μm de folga radial onde é possível utilizar outras distâncias entre mancais.

A figura 5.14 mostra os valores otimizados da Temperatura do Óleo em relação à Distância entre Mancais. Está claro, neste caso, que a temperatura do óleo dever ser mantida perto de 30°C.

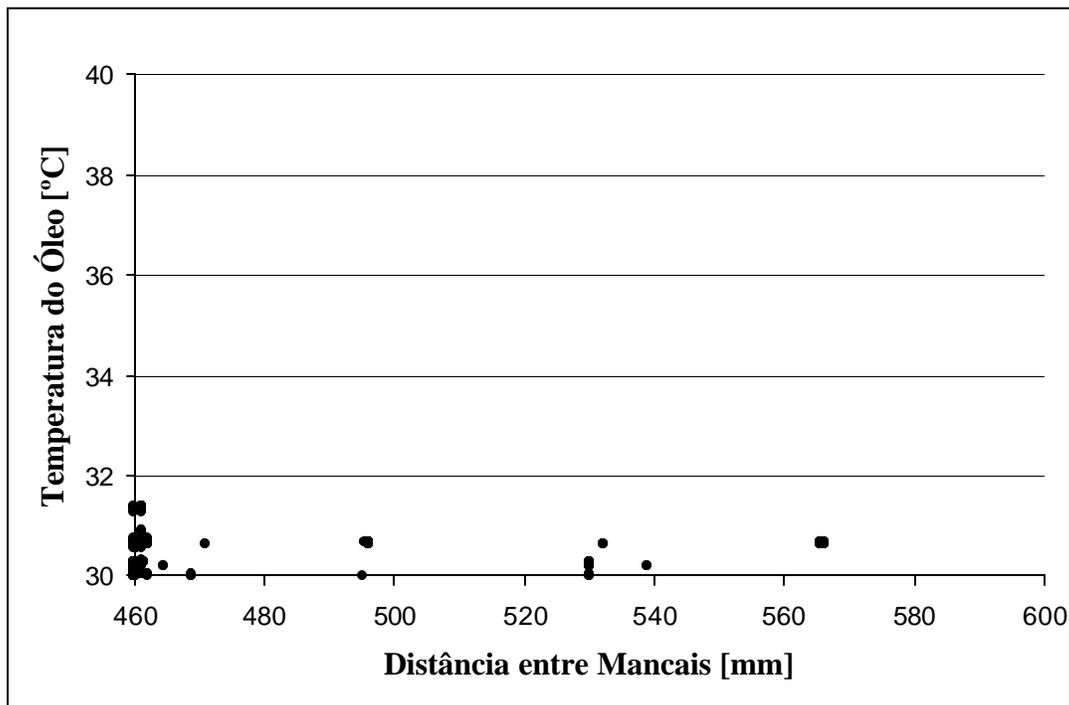


Fig. 5.14 – Relação entre Temperatura do Óleo e Distância entre Mancais.

A figura 5.15 mostra os valores otimizados da Temperatura do Óleo em relação à Folga Radial. Também fica claro que a temperatura do óleo dever ser mantida perto de 30°C.

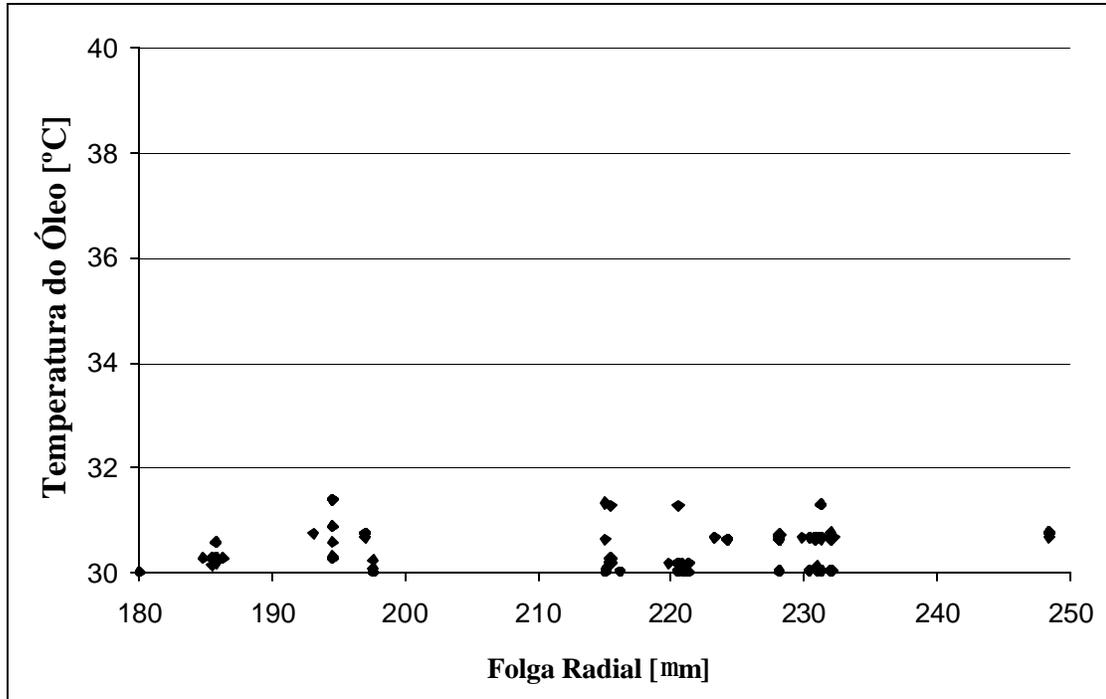


Fig. 5.15 – Relação entre Temperatura do Óleo e Folga Radial

A série completa de valores ótimos para as soluções conflitantes está mostrada na Figura 5.16. Cada cor representa uma combinação de valores otimizados respectivos às soluções ótimas encontradas pelo MOGA. Pode-se visualizar que para todas as soluções ótimas encontradas, a temperatura do óleo está perto de 30°C. A figura 5.16 mostra, por exemplo, que uma possível solução ótima seria a combinação: 460 mm de distância entre mancais, 250 μm de folga radial e 31 °C de temperatura do óleo. Outra possível solução ótima seria a combinação: 460 mm de distância entre mancais, 190 μm de folga radial e 30 °C de temperatura do óleo. Entre as soluções mostradas na figura 5.16, não existe a melhor solução, todas são consideradas ótimas e para se achar a melhor solução entre elas, outro critério de escolha deve ser utilizado, diferente dos parâmetros utilizados na otimização.

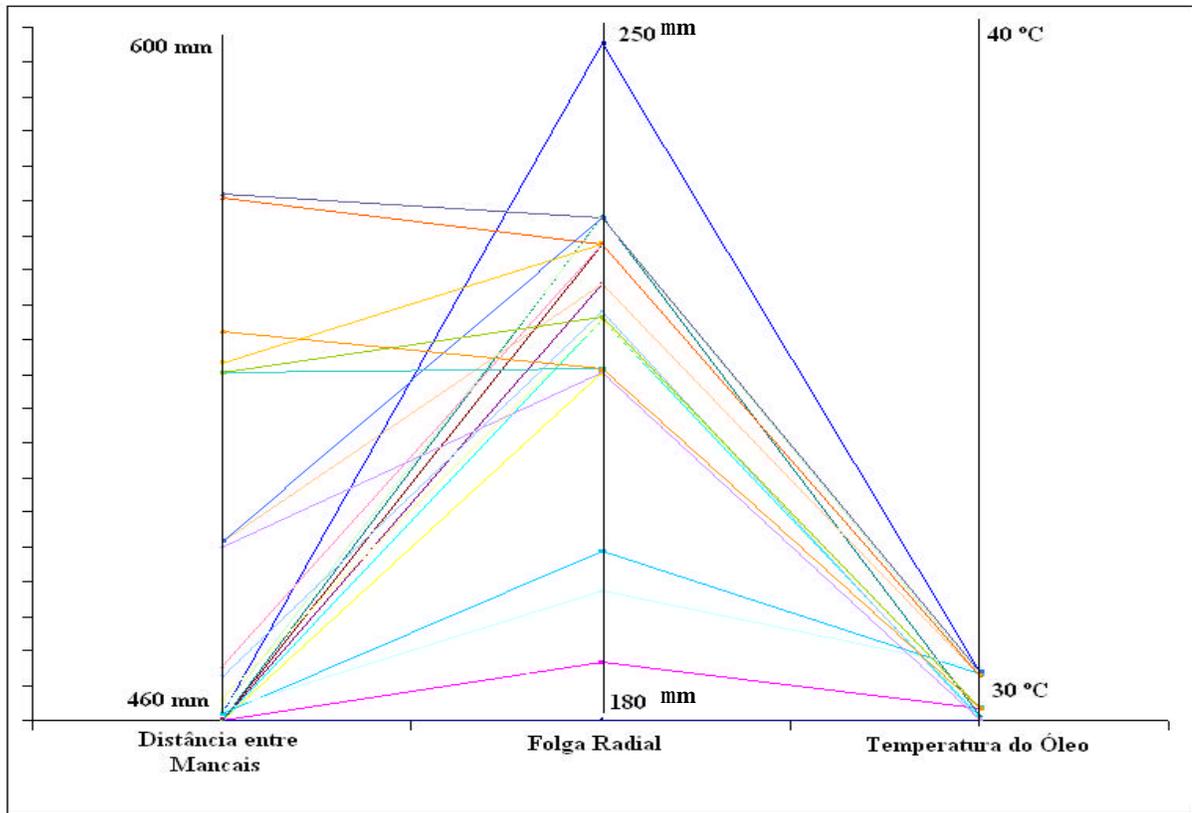


Fig. 5.16 – Resultado da Otimização Multi-Objetivos

Capítulo 6

Aplicações Experimentais e seus Resultados

Neste capítulo pretende-se estender o estudo realizado no capítulo 5, aplicando-se as técnicas em um sistema rotor-mancal experimental montado no Laboratório de Máquinas Rotativas. A figura 6.1 representa um esquema da bancada experimental.

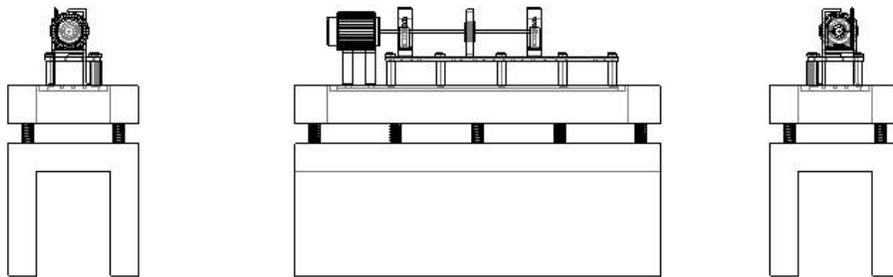


Figura 6.1 – Representação da bancada experimental.

As respostas foram obtidas diretamente de medições realizadas nos mancais, considerando-se as amplitudes nas direções X e Y, para ambos os mancais. As temperaturas nos mancais também foram medidas. A figura 6.2 mostra o esquema de instrumentação e sensores que monitoram aos mancais hidrodinâmicos da bancada.

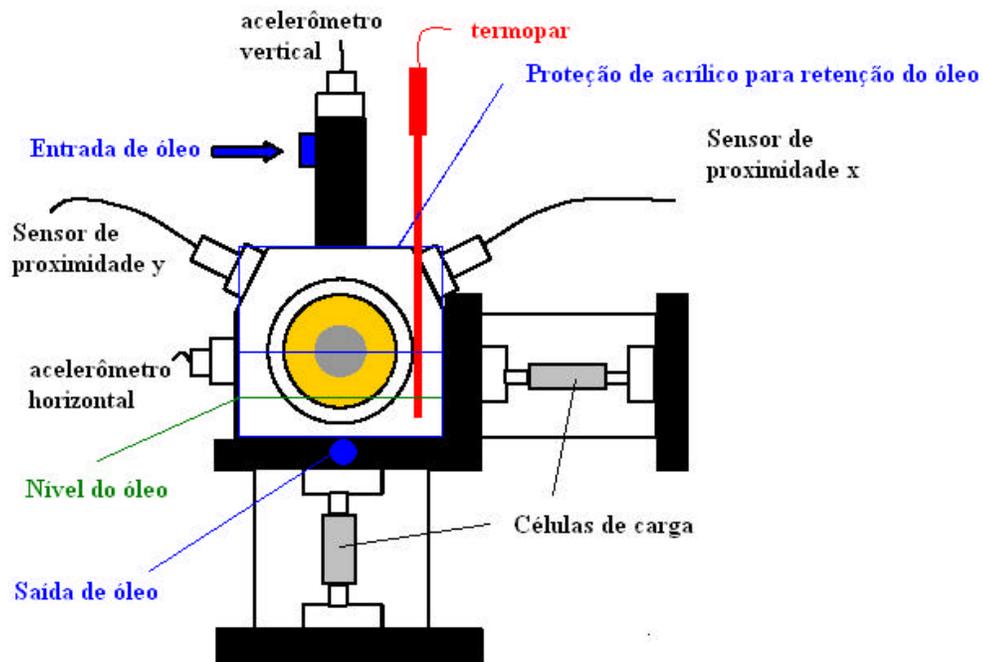


Figura 6.2 – Esquema de sensores e instrumentos nos mancais.

Os parâmetros de projeto, ou fatores de planejamento de experimentos, serão os mesmos utilizados nas simulações teóricas, sendo a distância entre mancais, a folga radial e a temperatura do óleo. Os limites inferiores e superiores foram adequados de acordo com a bancada experimental. Para a distância entre mancais foram selecionados os valores 350 mm e 600 mm. Para a folga radial foram mantidos os mesmos valores de 180 μm e 250 μm . E para a temperatura do óleo optou-se pela temperatura ambiente e um acréscimo de 10°C para a temperatura superior, assim, os limites estabelecidos para a temperatura ficaram em 26°C e 37°C. O óleo foi aquecido no tanque e a temperatura monitorada nos mancais. Os parâmetros podem ser identificados na tabela 6.1.

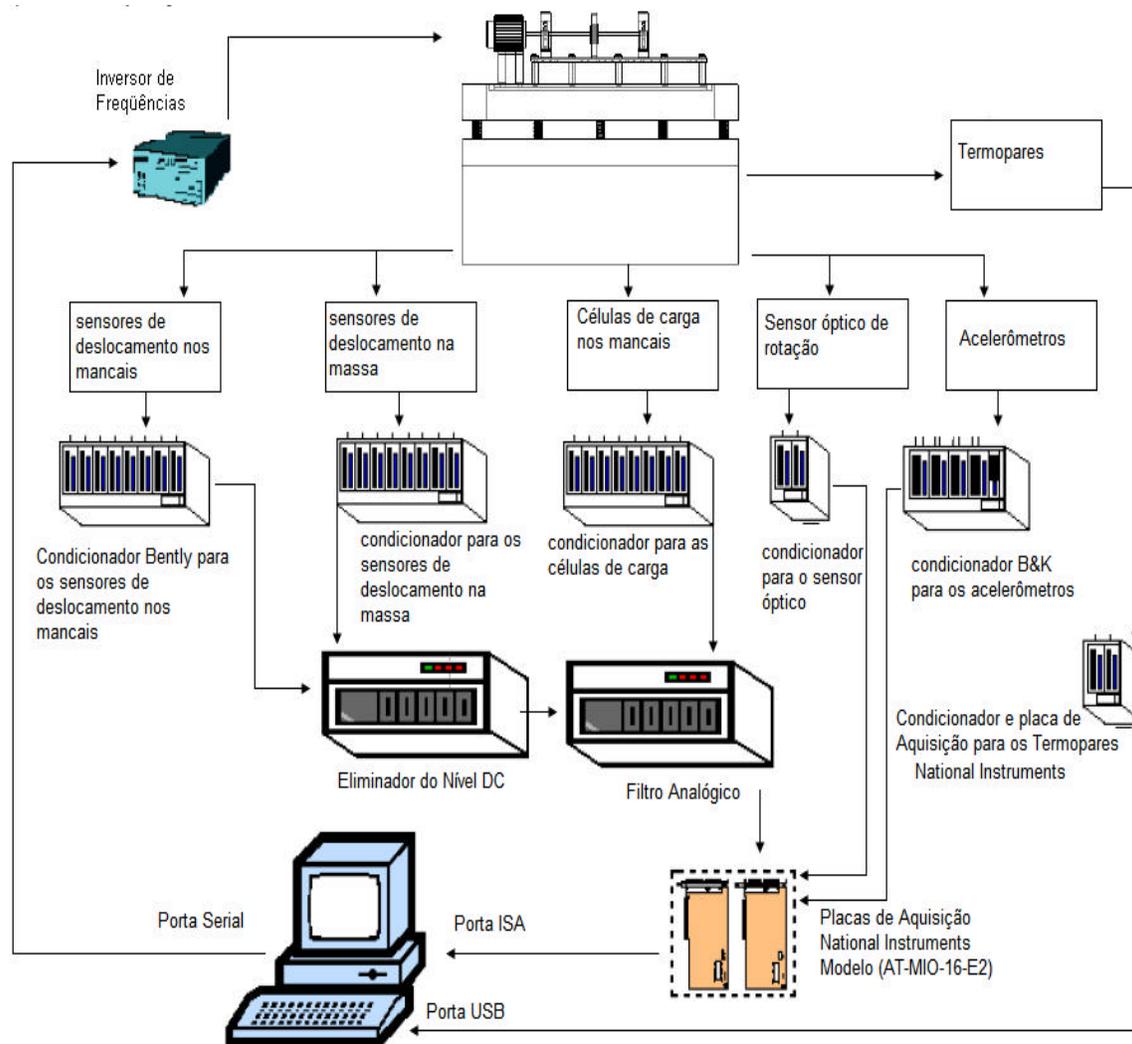


Figura 6.3 – Esquema da instrumentação utilizada

6.1 Delineamento de Experimentos (DOE) e Superfície de Resposta (MSR)

O DOE seguiu o mesmo procedimento utilizado no capítulo 5, considerando três parâmetros com dois limites, inferior e superior, porém foram realizados dezesseis experimentos, uma vez que para cada um dos oito experimentos propostos pelo DOE foi aplicada análise modal no rotor com excitação pelo shaker em duas direções, vertical e horizontal, ambas transversais ao eixo. A figura 6.4 mostra o sistema montado com o shaker na posição vertical.

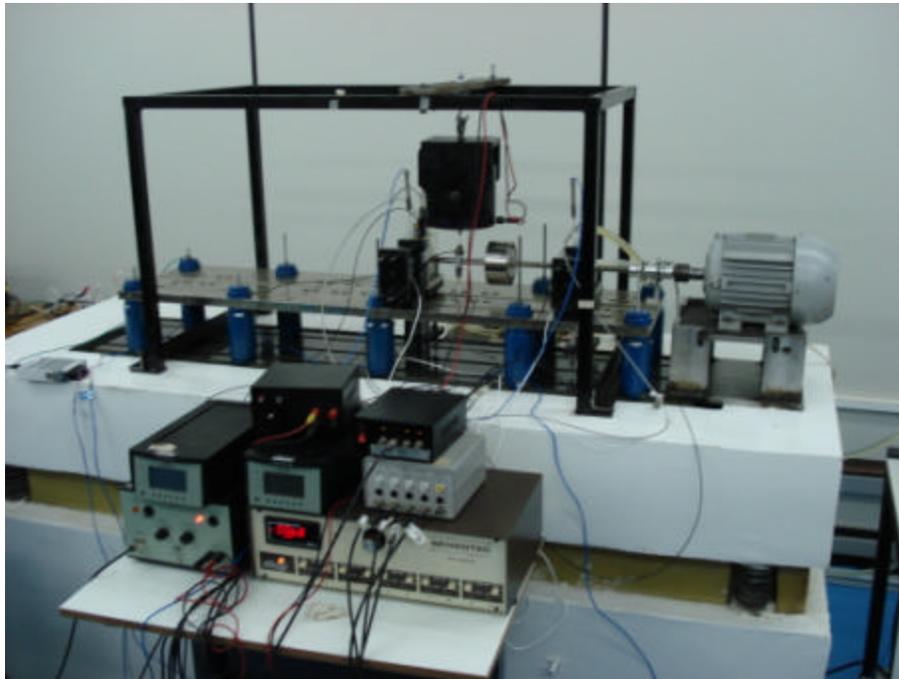


Figura 6.4 – Bancada Experimental – LAMAR - UNICAMP

Parâmetro	Variáveis	Limite Inferior (-)	Limite Superior (+)
Distancia entre Mancais	x_1	350 mm	600 mm
Folga Radial	x_2	180 μm	250 μm
Temperatura do Óleo	x_3	26°C	37°C

Tabela 6.1 – Parâmetros para análise experimental

6.1.1 Medições – Amplitudes no domínio da frequência

Os resultados das medições são mostrados nas figuras 6.5a até 6.5p. As figuras 6.5 mostram as curvas de amplitude no domínio da frequência. Uma vez que podemos utilizar medidas relativas em nossa análise, a amplitude é obtida em função da força de excitação (FRF) em [mm/N] e a primeira frequência natural em ciclos por segundo [Hz].

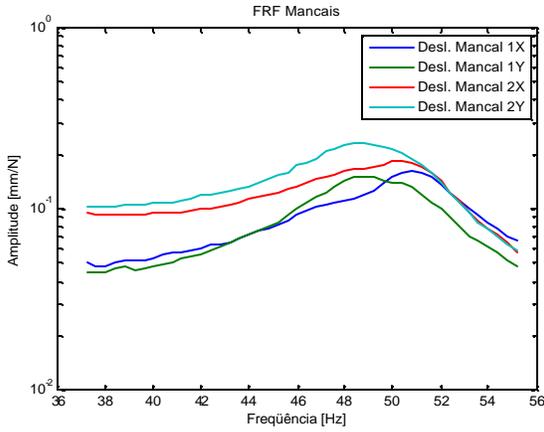


Figura 6.5a – Excitação Horizontal Amplitudes –

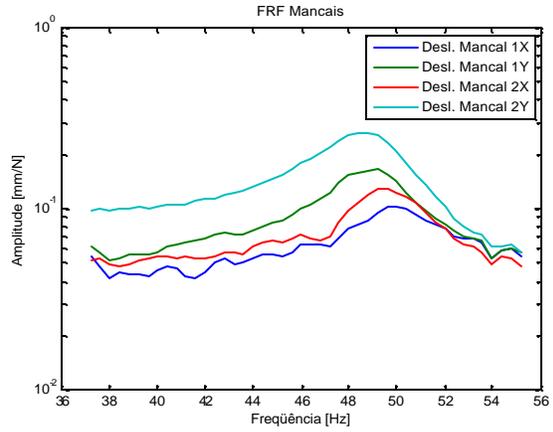


Figura 6.5b – Excitação Vertical Amplitudes – Experimento 1

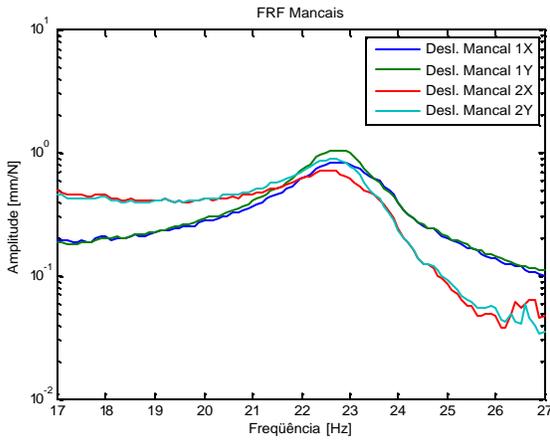


Figura 6.5c – Excitação Horizontal Amplitudes – Experimento 2

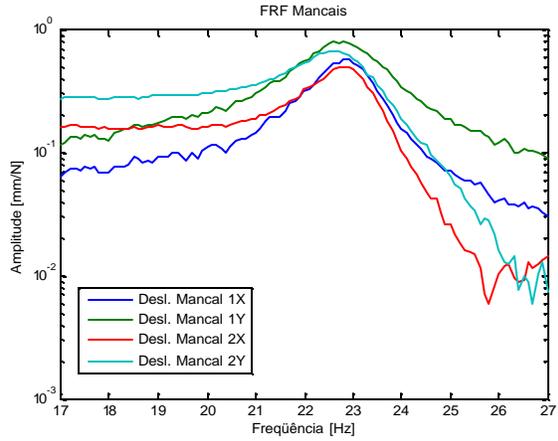


Figura 6.5d – Excitação Vertical Amplitudes – Experimento 2

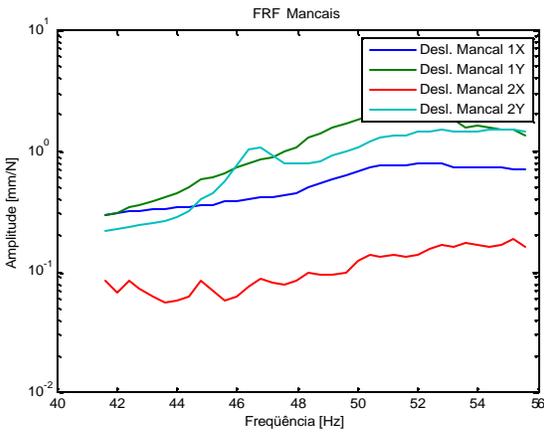


Figura 6.5e – Excitação Horizontal Amplitudes – Experimento 3

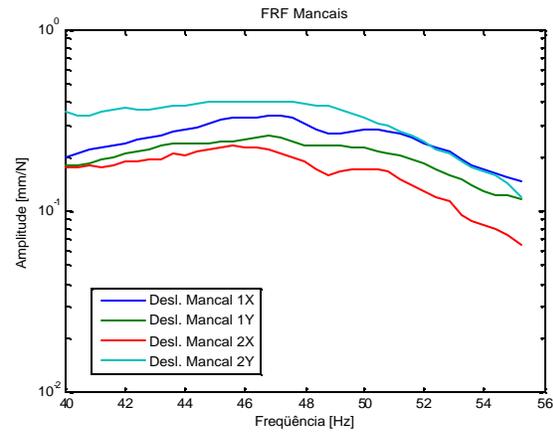


Figura 6.5f – Excitação Vertical Amplitudes – Experimento 3

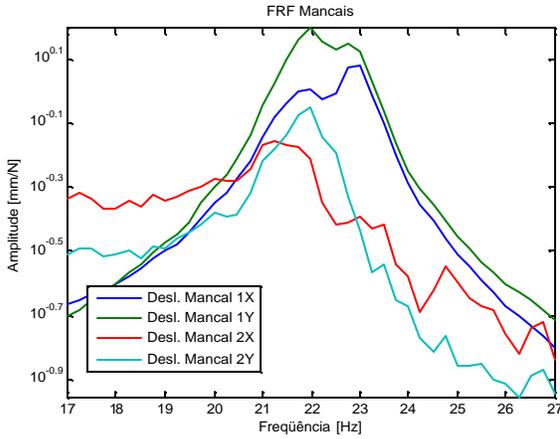


Figura 6.5g – Excitação Horizontal
Amplitudes – Experimento 4

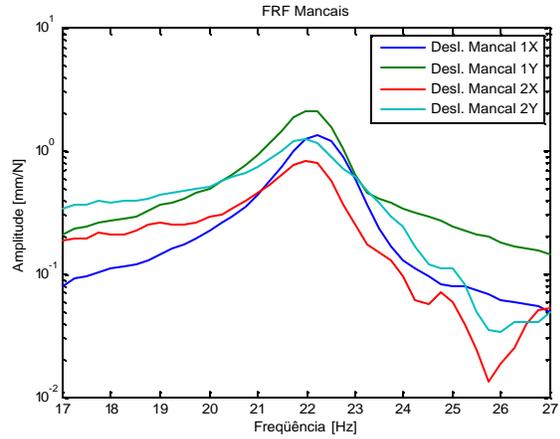


Figura 6.5h – Excitação Vertical
Amplitudes – Experimento 4

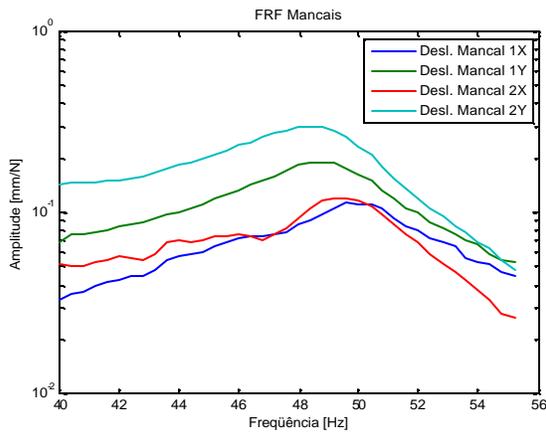


Figura 6.5i – Excitação Horizontal
Amplitudes – Experimento 5

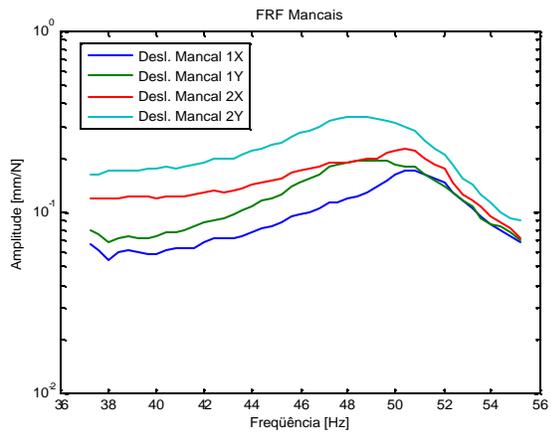


Figura 6.5j – Excitação Vertical
Amplitudes – Experimento 5

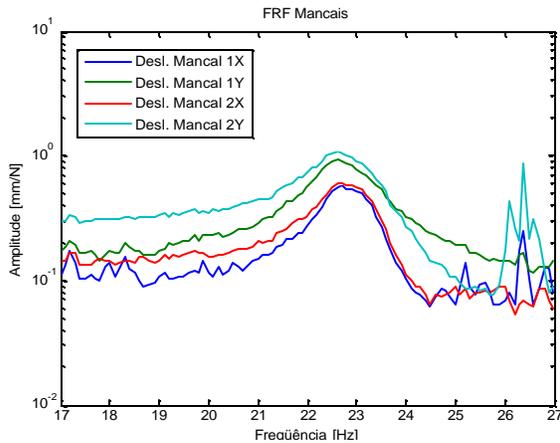


Figura 6.5l – Excitação Horizontal
Amplitudes – Experimento 6

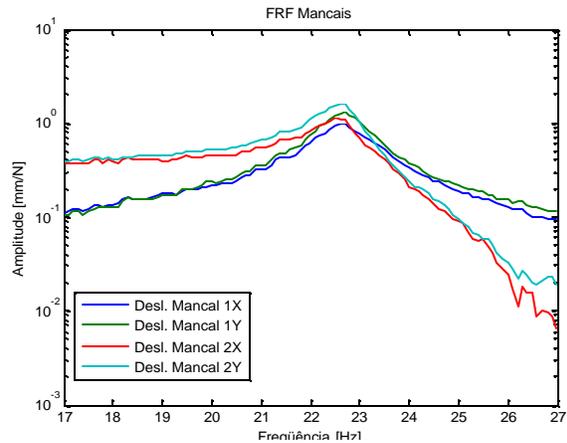


Figura 6.5m – Excitação Vertical
Amplitudes – Experimento 6

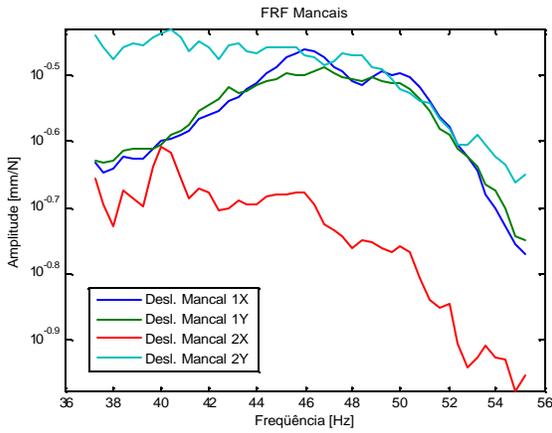


Figura 6.5n – Excitação Horizontal
Amplitudes – Experimento 7

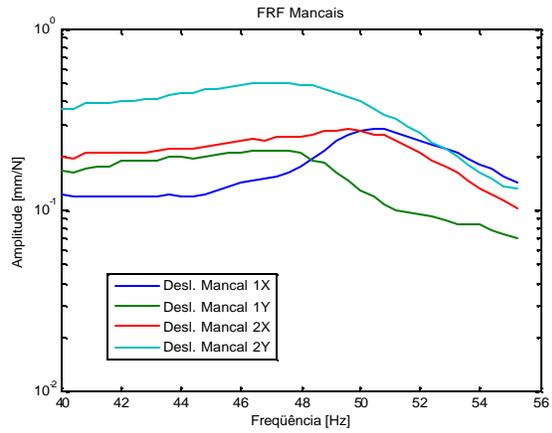


Figura 6.5o – Excitação Vertical
Amplitudes – Experimento 7

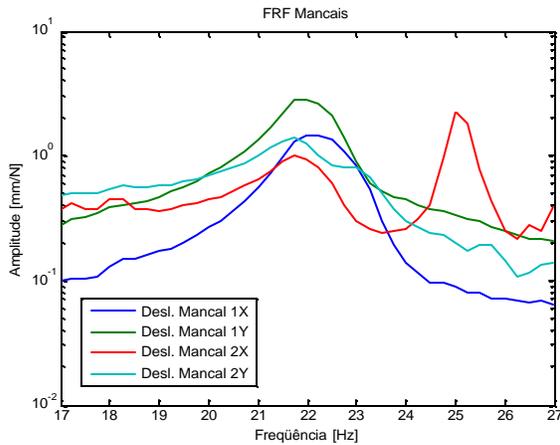


Figura 6.5p – Excitação Horizontal
Amplitudes – Experimento 8

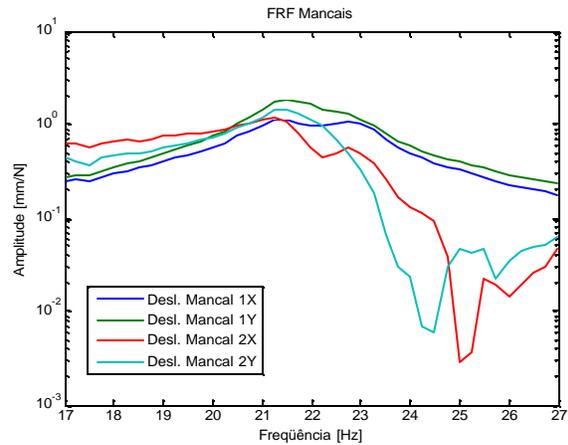


Figura 6.5q – Excitação Vertical
Amplitudes – Experimento 8

A tabela 6.2 mostra os resultados experimentais obtidos. Para cada análise foram tomadas as freqüências naturais e as amplitudes X e Y nos mancais 1 e 2. Deste modo, para cada ordem do DOE foram consideradas duas respostas de freqüência e quatro respostas em cada mancal, combinadas linearmente nas análises de superfície de resposta. Foram geradas com estes dados três funções polinomiais, sendo uma para freqüência natural, uma para amplitude no mancal 1 e outra para amplitude no mancal 2.

Ensaio	DOE	Freq. Natural [Hz]	Amplitude Mancal 1X [mm/N]	Amplitude Mancal 1Y [mm/N]	Amplitude Mancal 2X [mm/N]	Amplitude Mancal 2Y [mm/N]
1V	--	50,80	5,66E-06	7,09E-06	9,75E-06	1,70E-05
1H	--	49,20	2,95E-06	7,64E-06	5,11E-06	2,05E-05
2V	+-	22,90	2,59E-04	3,78E-04	1,82E-04	2,68E-04
2H	+-	23,20	1,30E-04	2,39E-04	1,04E-04	1,58E-04
3V	-+-	52,40	2,35E-06	6,90E-04	2,25E-07	1,69E-05
3H	-+-	47,20	6,20E-05	3,82E-05	2,83E-05	8,52E-05
4V	++-	23,00	5,03E-04	1,60E-03	1,10E-04	5,96E-04
4H	++-	22,25	8,59E-04	1,84E-03	2,24E-04	4,50E-04
5V	++	48,80	3,44E-06	1,08E-05	3,72E-06	2,73E-05
5H	++	50,40	6,96E-06	1,28E-05	1,44E-05	3,83E-05
6V	+++	23,30	1,19E-04	3,26E-04	1,43E-04	4,15E-04
6H	+++	22,70	3,19E-04	6,26E-04	3,26E-04	8,76E-04
7V	+++	50,00	7,52E-05	6,84E-05	4,58E-05	1,22E-04
7H	+++	49,60	1,85E-05	1,08E-05	2,76E-05	1,01E-04
8V	+++	22,50	1,45E-03	3,80E-03	4,06E-04	6,99E-04
8H	+++	21,75	9,87E-04	2,09E-03	4,06E-04	1,51E-03

Tabela 6.2 – Resultados Experimentais

A figura 6.6 mostra a entrada de valores para a análise da frequência natural. Estas análises foram executadas com o programa ÓTIMA®, descrito no capítulo 5. O planejamento experimental e análise para a amplitude são executados da mesma maneira.

Planejamentos Iniciais - esc

Planejamento Fatorial Completo | Planejamento Fatorial Fracionário | Arranjo Ortogonal (Taguchi)

Dados de Entrada

Nº de parâmetros (U.I.): 3
 Nº de níveis: 2
 Nº de réplicas: 2
 Nº de respostas: 1

Níveis dos Parâmetros

Nome do Parâmetro	Baixo	Médio	Alto
Distância	350		600
Folga	180		250
Temperatura	26		37

Visualizar valores: reais
 Ordenar valores pela coluna: Nº do experimento

nº exp.	I	Distância	Folga	Temperatura	R 1	R 2
1	1	350	180	26	50,8	49,2
2	1	600	180	26	22,9	23,2
3	1	350	250	26	52,4	47,2
4	1	600	250	26	23	22,25
5	1	350	180	37	48,8	50,4
6	1	600	180	37	23,3	22,7
7	1	350	250	37	50	49,6
8	1	600	250	37	22,5	21,75

Imprimir | Salvar dados | Continuar

Importante: separador de casas decimais (, ou .) configurado no sistema (painel de controle).

Fig. 6.6 – Entrada de dados no Programa ÓTIMA®.

O programa ÓTIMA® determina os coeficientes das funções polinomiais a serem posteriormente otimizadas. Estes coeficientes normalmente são sujeitos a análises estatísticas que não serão aplicadas neste trabalho.

6.2 Funções Polinomiais

Tomando-se os coeficiente gerados pelo ÓTIMA® foram geradas três funções polinomiais que serão utilizados como funções objetivos.

Para a frequência natural tem-se a seguinte função polinomial:

$$f1 = 36,25 - 13,55x1 - 0,163x2 - 0,119x3 - 0,163x1x2 - \\ - 0,019x1x3 - 0,006x2x3 - 0,106x1x2x3 \quad (6.1)$$

E para a amplitude no mancal 1 tem-se a seguinte função polinomial

$$f2 = 51,71 + 45,32x1 + 36,38x2 + 10,31x3 + 30,7x1x2 + \\ + 14,12x1x3 + 7,85x2x3 + 11,79x1x2x3 \quad (6.2)$$

E para a amplitude no mancal 2 tem-se a seguinte função polinomial

$$f3 = 23,24 + 19,72x1 + 6,94x2 + 9,02x3 + 5,12x1x2 + \\ + 7,89x1x3 + 2,27x2x3 + 1,43x1x2x3 \quad (6.3)$$

6.2.1 Superfícies de Resposta

Estas funções polinomiais 6.1 a 6.3, geram superfícies de resposta, as quais são mostradas figuras 6.7 a 6.12 para os parâmetros principais. Pode-se observar que as superfícies são conflitantes quando se consideram a frequência natural e as amplitudes, com referência aos parâmetros $x1$ (distância entre mancais), $x2$ (folga radial) e $x3$ (temperatura do óleo).

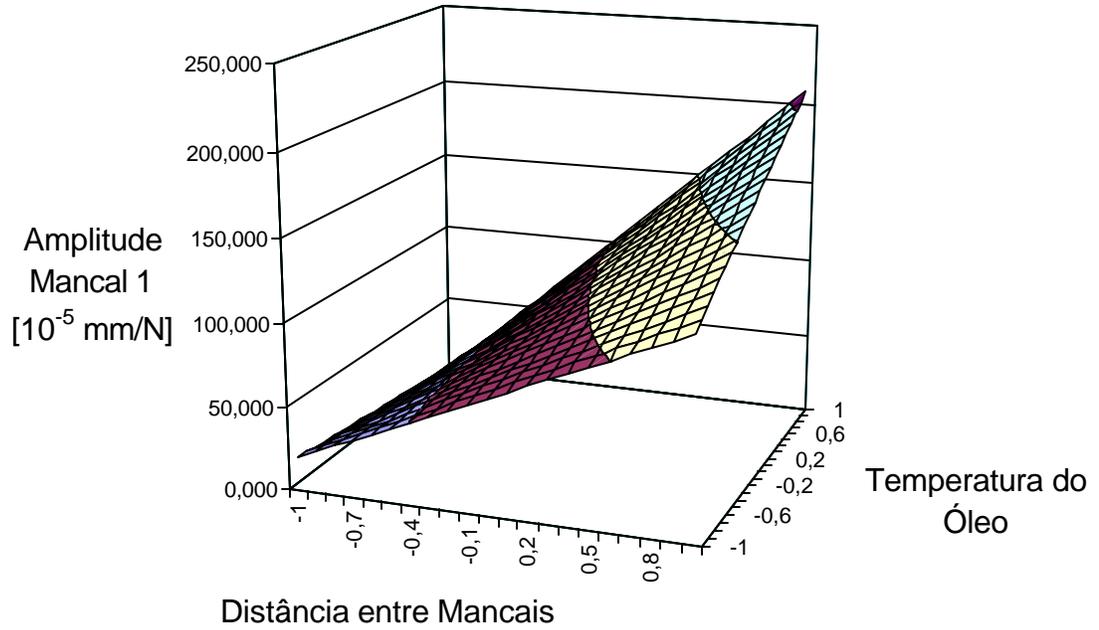


Fig. 6.7 – Superfície de Resposta para f_2 (x_1 = distância entre mancais e x_3 = temperatura do óleo, x_2 = constante = $180 \mu\text{m}$)

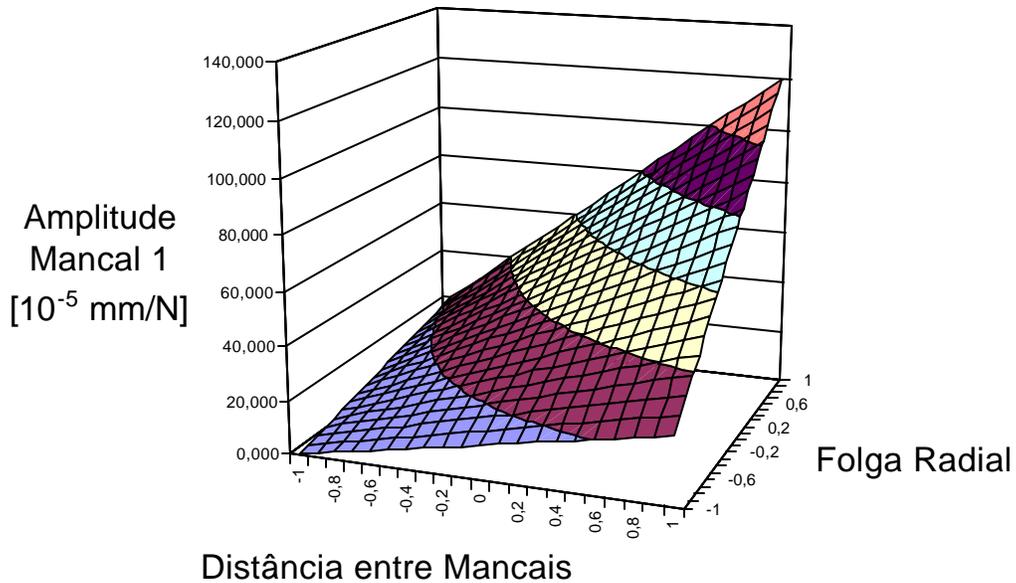


Fig. 6.8 – Superfície de Resposta para f_2 (x_1 = distância entre mancais e x_2 = folga radial, x_3 = constante = 26°C)

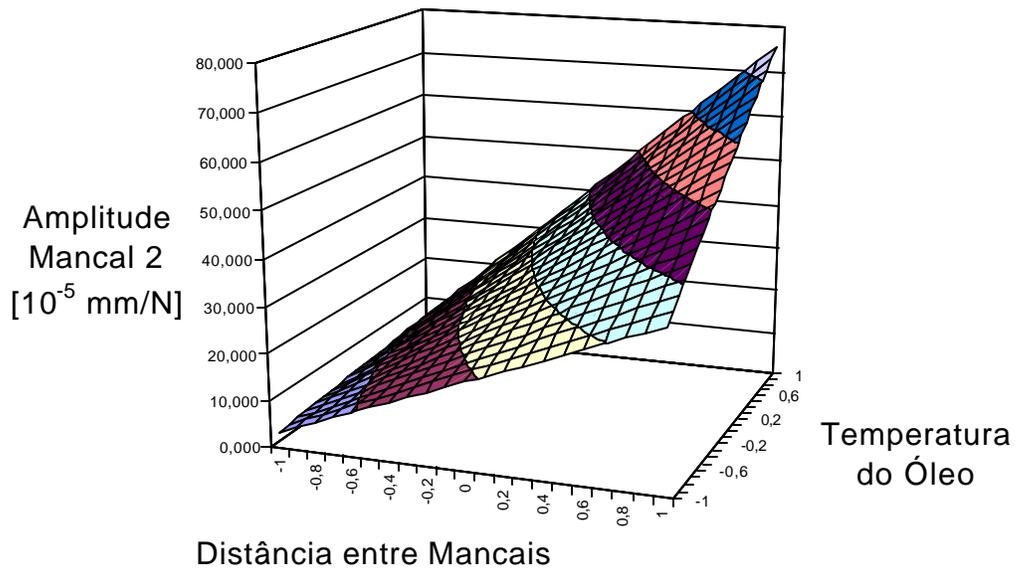


Fig. 6.9 – Superfície de Resposta para f_3 (x_1 = distância entre mancais e x_3 = temperatura do óleo, x_2 = constante = $180 \mu\text{m}$)

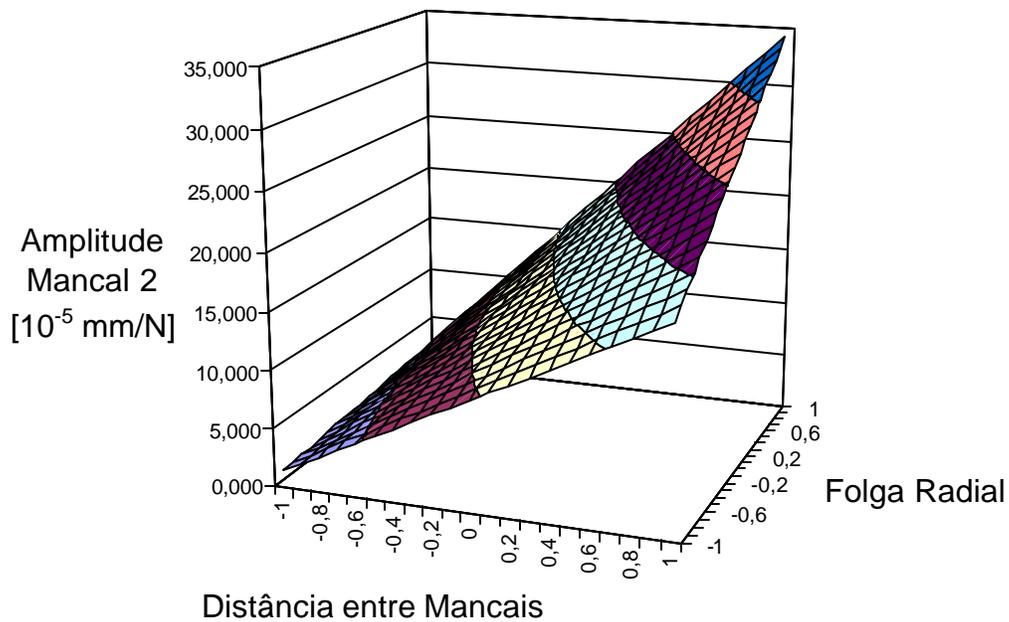


Fig. 6.10 – Superfície de Resposta para f_3 (x_1 = distância entre mancais e x_2 = folga radial, x_3 = constante = 26°C)

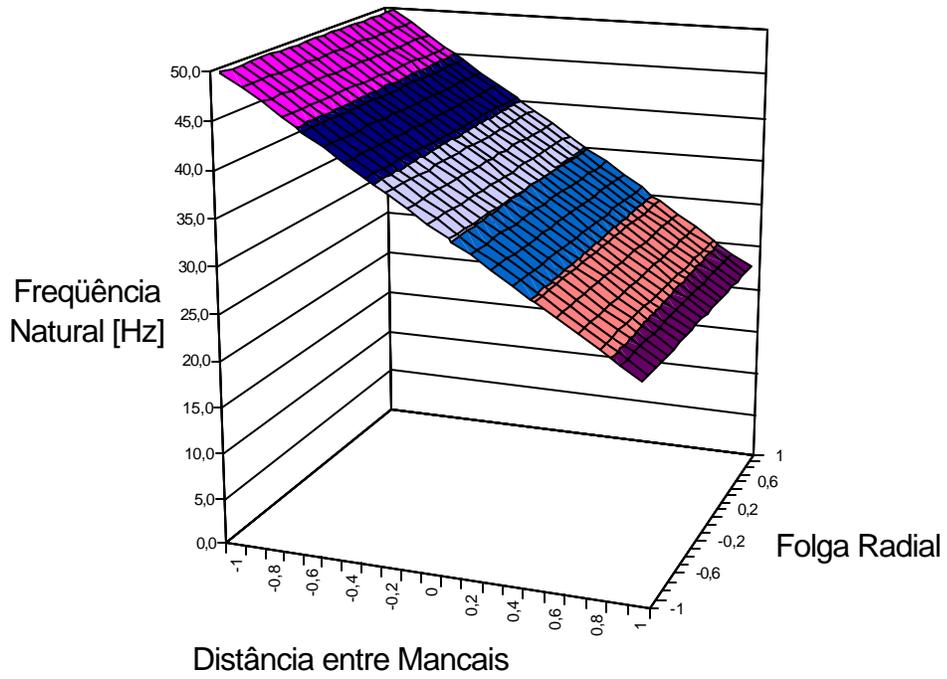


Fig. 6.11 – Superfície de Resposta para f_1 (x_1 = distância entre mancais e x_2 = folga radial, x_3 = constante = 26°C)

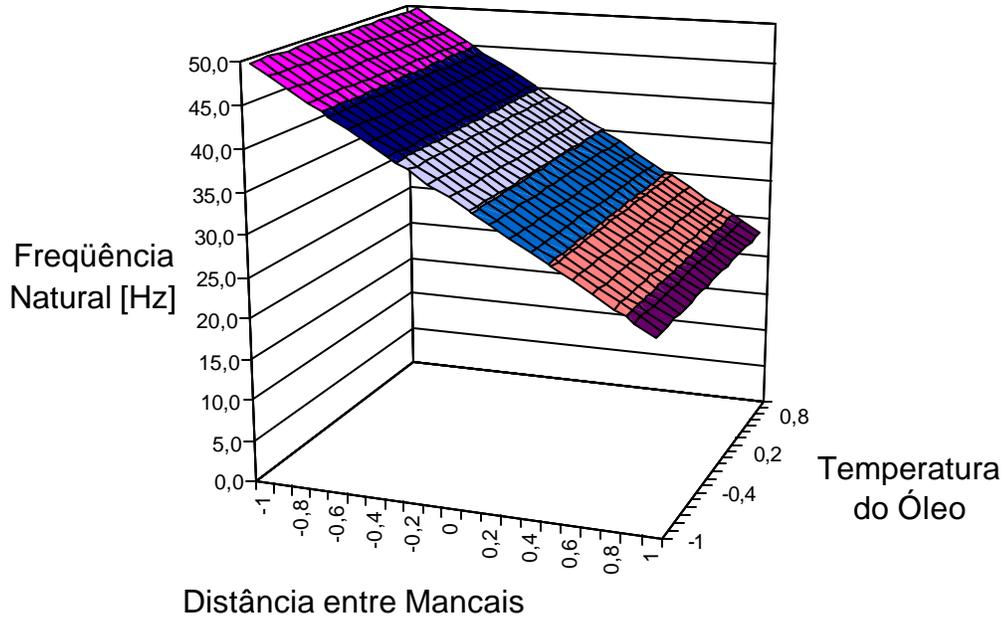


Fig. 6.12 – Superfície de Resposta para f_1 (x_1 = distância entre mancais e x_3 = temperatura do óleo, x_2 = constante = $180\ \mu\text{m}$)

6.3 Aplicação do Algoritmo Genético Multi-Objetivos

Para a otimização multi-objetivos foram implementados quatro algoritmos genéticos, MOGA, NSGA, NSGA-II e SPEA. O procedimento de entrada dos dados é o mesmo utilizado no capítulo 5.

6.3.1 Processo de Otimização Multi-Objetivos

Minimizar	$f_1(x_1, x_2, x_3)$
	$f_2(x_1, x_2, x_3)$
	$f_3(x_1, x_2, x_3)$
Sujeito a	$-1 \leq x_1 \leq 1$
	$-1 \leq x_2 \leq 1$
	$-1 \leq x_3 \leq 1$

Nesta otimização os seguintes parâmetros foram usados:

Tamanho da População: 500 indivíduos ou soluções

Número de Gerações: 1500

Probabilidade de Cruzamento: 5 %

Probabilidade de Mutação: 2 %

Número de Bits em cada variável: 8

As equações 6.1 a 6.3 formam o espaço de busca mostrado nas figuras 6.13 e 6.14. Os pontos mostrados representam a geração inicial gerada aleatoriamente. As soluções mostradas são soluções conflitantes que podem ser otimizados pelos algoritmos genéticos multi-objetivos. A figura 6.13 mostra a relação entre f_1 (frequência natural) e f_2 (amplitude no mancal 1) e a figura 6.14 mostra a relação entre f_1 e f_3 (amplitude no mancal 2).

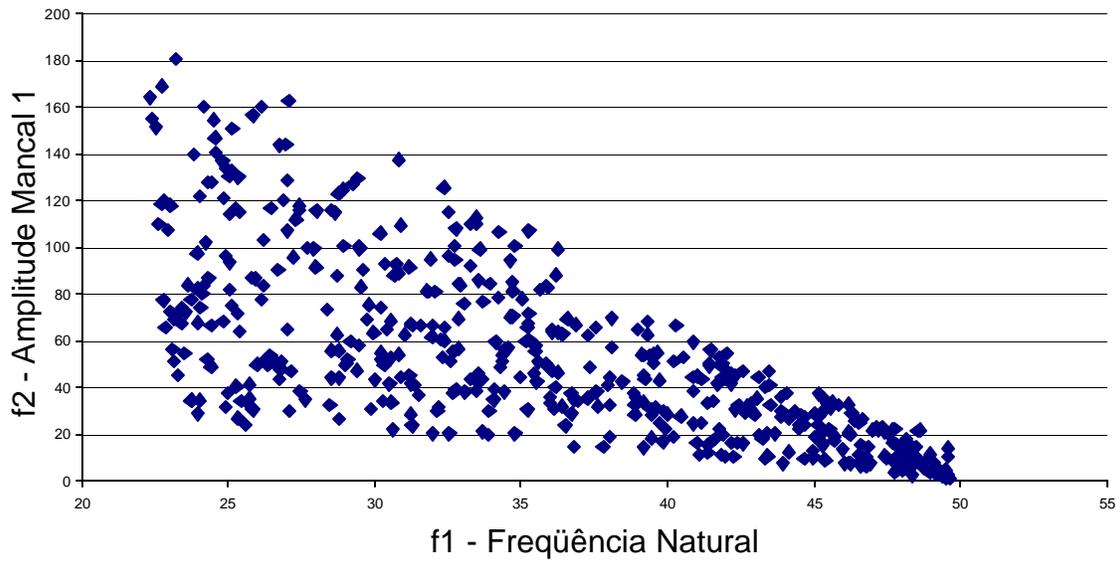


Fig. 6.13 – Geração Inicial (f1- frequência x f2 – amplitude mancal 1)

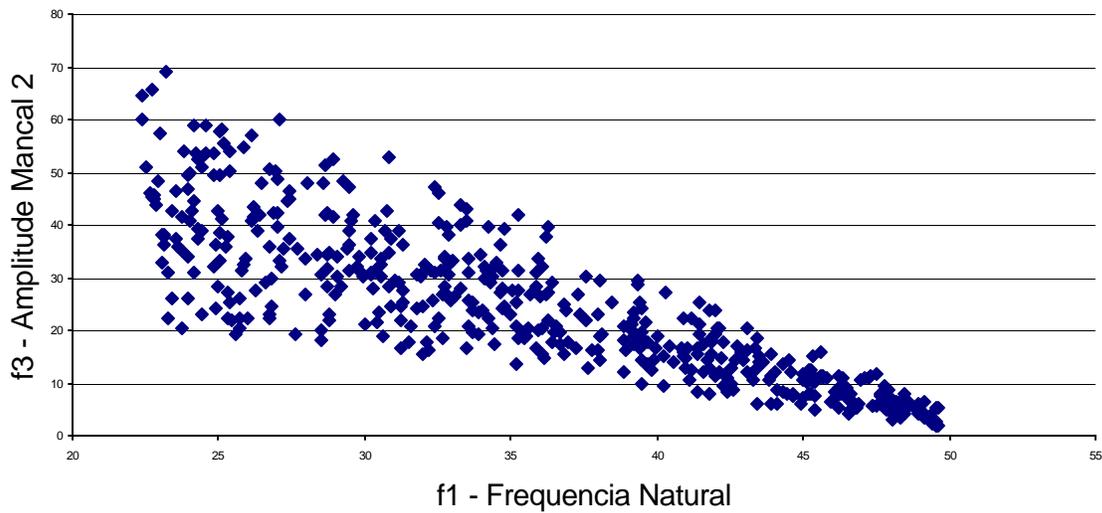


Fig. 6.14 – Geração Inicial (f1- frequência x f3 – amplitude mancal 2)

O algoritmo de otimização multi-objetivos MOGA foi aplicado às funções objetivos f1 a f3, determinadas pelas equações 6.1 a 6.3 resultando nas soluções mostradas nas figuras 6.15 e 6.16.

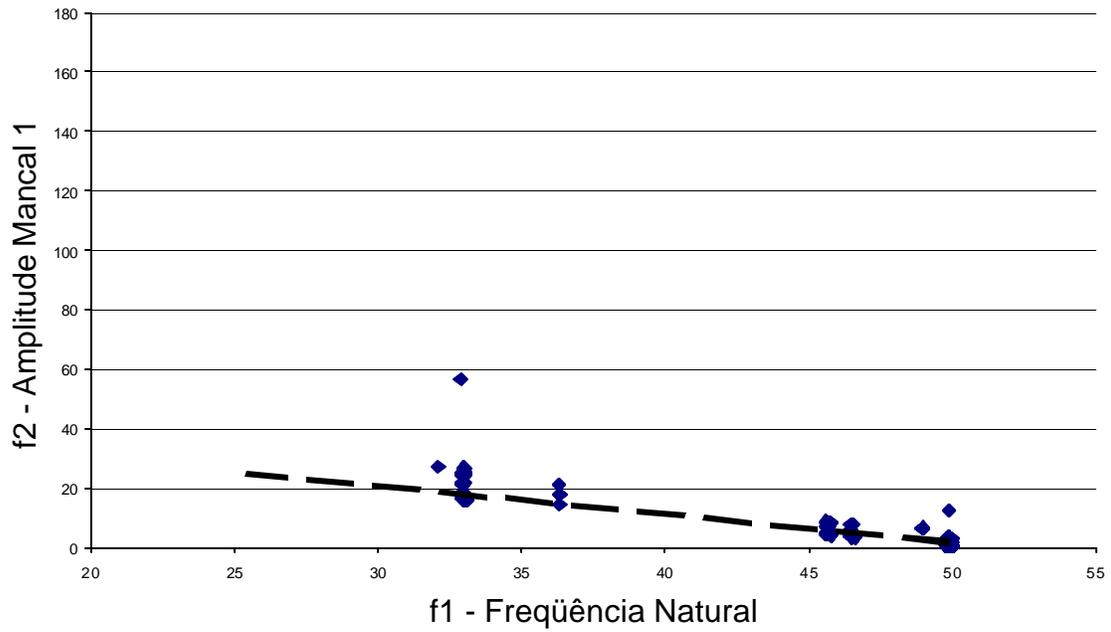


Fig. 6.15 – Soluções e Fronteira Pareto-Ótima (f1- frequência x f2 – amplitude mancel 1)

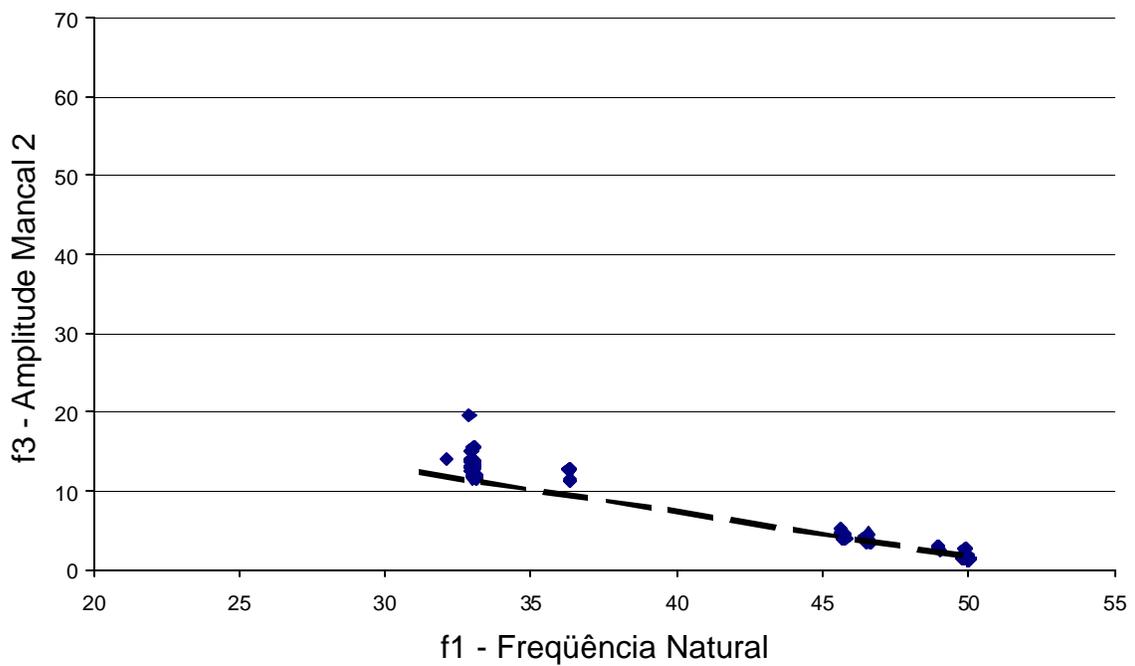


Fig. 6.16 – Soluções e Fronteira Pareto-Ótima (f1- frequência x f3 – amplitude mancel 2)

6.4 Resultados

As Fronteiras Pareto-Ótimas estão representadas por linhas tracejadas conforme mostrado nas figuras 6.15 e 6.16. Observa-se que as soluções otimizadas encontradas pelo algoritmo MOGA situam-se perto das Fronteiras Pareto-Ótimas. Ocorreram também sobreposições de soluções, reduzindo o número de repostas ótimas.

A determinação das Fronteiras Pareto-Ótimas, mostradas pelas linhas tracejadas nas figuras 6.15 e 6.16, também foi facilitada pelo prévio conhecimento do problema. A dispersão de valores, apesar de estar mostrada ao longo das linhas tracejadas, não indica realmente que se trata de linhas contínuas ou segmentos de reta.

Sabendo que para cada solução temos uma combinação de parâmetros x_1 , x_2 e x_3 , podemos determinar seus valores para cada solução ótima. As relações entre as soluções conflitantes estão mostradas nas figuras 6.17 a 6.20.

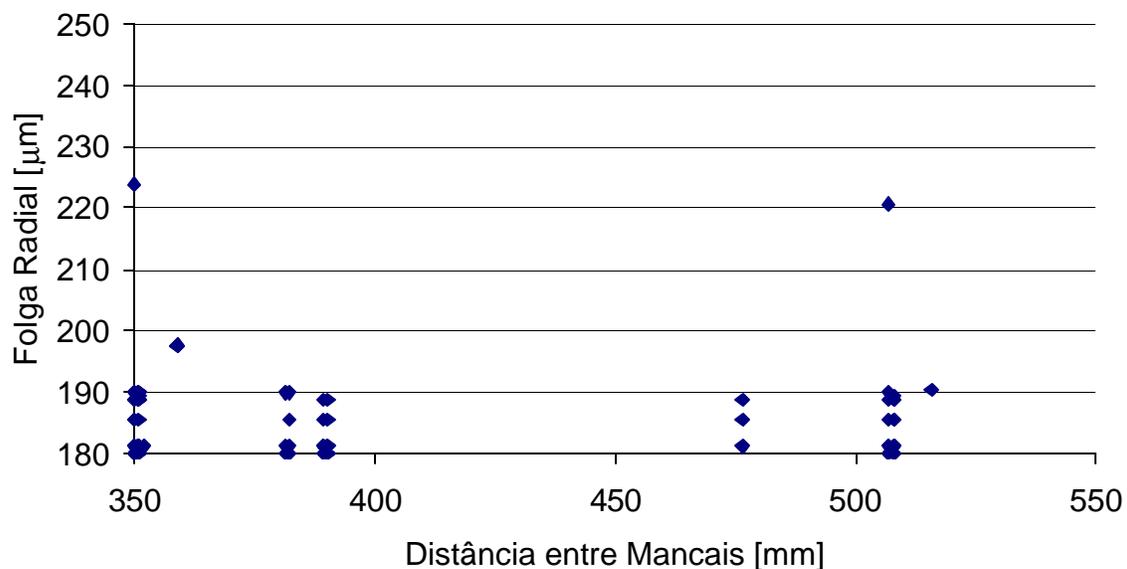


Fig. 6.17 – Relação entre Folga Radial e Distância entre Mancais

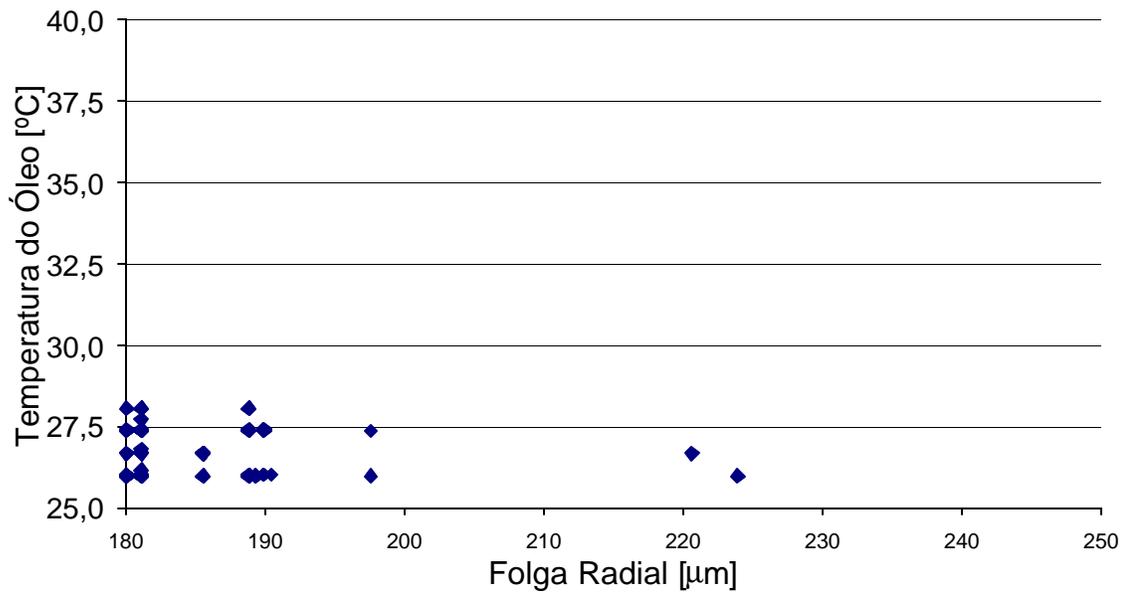


Fig. 6.19 – Relação entre Temperatura do Óleo e Folga Radial

A série completa de valores ótimos para as soluções conflitantes está mostrada na Figura 6.20. Novamente, como mostrado no capítulo 5, cada cor representa uma combinação de valores otimizados respectivos às soluções ótimas encontradas pelo MOGA. Pode-se visualizar que para todas as soluções ótimas encontradas, a temperatura do óleo aproxima-se de 26°C e na maioria das soluções, da folga radial de 180 μm . Entre as soluções ótimas encontradas, como já foi mencionado, não existe uma solução melhor que a outra com relação aos parâmetros utilizados e para se obter uma solução final deve-se utilizar outros parâmetros de análise.

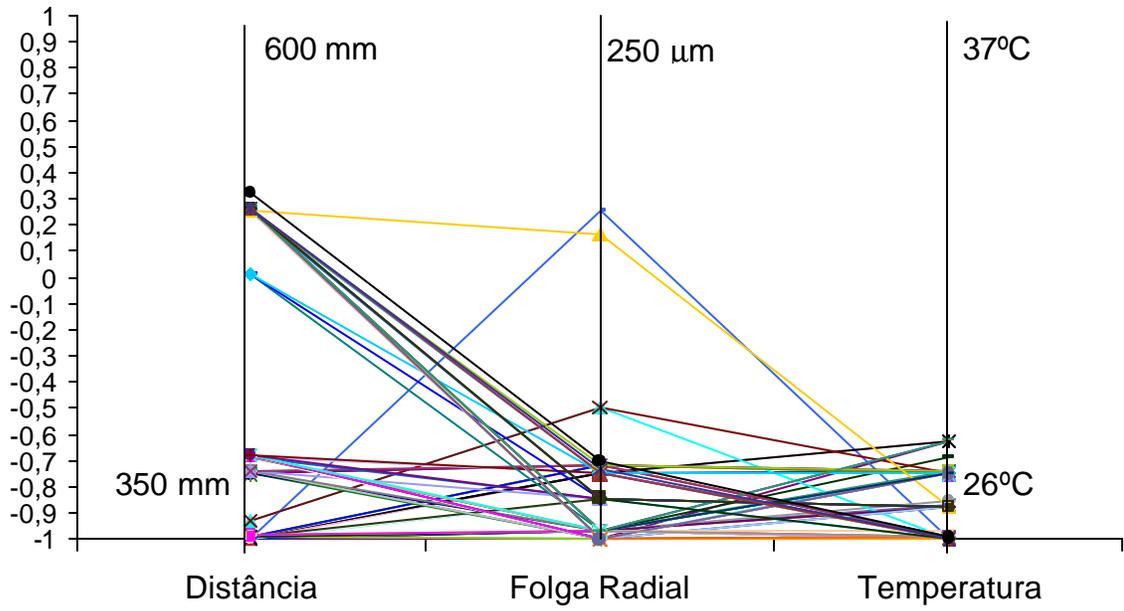


Fig. 6.20 – Resultados da Otimização Multi-Objetivos

Capítulo 7

Conclusões e Propostas Futuras

7.1 Conclusões

Este trabalho teve como objetivos principais o estudo e a demonstração de que métodos de otimização multi-objetivos através de algoritmos genéticos, bem como a utilização de técnicas estatísticas de planejamento de experimentos e metodologia de superfície de resposta podem ser utilizados de forma satisfatória na análise de sistemas mecânicos.

A utilização de métodos de planejamentos experimentais, mostrada neste trabalho como DOE, delineamento de experimentos, tem a capacidade de sistematizar a quantidade de experimentos a serem executados em um determinado problema experimental. Esta fase de estudo torna-se fundamental na otimização de custos durante um projeto, reduzindo tempos de ensaios, custos de preparação, recursos técnicos e pessoais. Os ensaios e simulações realizadas obedeceram ao planejamento fatorial completo 2^3 , sendo três parâmetros trabalhados a dois níveis, compondo oito soluções que se inter-relacionam. Os valores encontrados mostraram-se suficientes para posterior aplicação da metodologia de superfície de resposta.

A MSR, metodologia de superfície de resposta, foi aplicada aos valores encontrados no delineamento de experimentos. Considerando a finalidade de se gerar funções polinomiais que representassem o sistema mecânico, pode-se dizer que a MSR cumpriu seu papel dentro dos limites bi-lineares utilizados. Contudo, a metodologia de superfície de resposta pode ser caracterizada como um processo de regressão linear e interpolação, no qual os valores experimentais são utilizados na regressão. Sua utilização vem principalmente das áreas de pesquisas de campo e geralmente é acompanhada de uma análise estatística do tipo teste de hipóteses ou ANOVA, análise de variância.

Os capítulos 3 e 4 foram dedicados, em quase suas totalidades, à análise de otimizações multi-objetivos, enfatizando os algoritmos evolutivos, mais especificamente os algoritmos genéticos multi-objetivos. Foram desenvolvidos quatro programas computacionais, em caráter experimental, que possibilitaram a aplicação e análise dos algoritmos genéticos multi-objetivos. Concluiu-se que todos os quatro algoritmos propostos, MOGA, NSGA, SPEA e NSGA-II podem ser aplicados e que as diferenças, neste caso estudado, são pequenas.

Os algoritmos genéticos elitistas (SPEA e NSGA-II) não apresentaram uma melhor dispersão das soluções conforme citado na literatura mostrada. O algoritmo MOGA, talvez pela simplicidade das funções polinomiais utilizadas, apresentou-se como o mais rápido mantendo a diversidade de soluções perto da Fronteira de Pareto. Portanto, não foi possível determinar qual dos quatro algoritmos é melhor, porém foi possível concluir que os algoritmos genéticos utilizados cumprem seu papel de encontrar soluções ótimas conflitantes, mantendo a diversidade, utilizando o conceito de dominância.

Os capítulos 5 e 6 mostraram que é possível a utilização de algoritmos genéticos multi-objetivos na análise dinâmica de sistemas mecânicos. As funções polinomiais podem ser substituídas por equações determinísticas, por exemplo, equações de energia ou variacionais, possibilitando a otimização de sistemas complexos.

Mostrou-se também que é possível através de uma metodologia estatística, tal como superfície de resposta, a utilização de dados obtidos diretamente de análises experimentais, sem o

conhecimento prévio das equações diferenciais normalmente utilizadas. No entanto, o conhecimento prévio do comportamento do sistema facilita a análise.

Analisando as soluções finais otimizadas, percebemos que tanto para a simulação teórica quanto para a análise experimental em bancada, a temperatura do óleo deve permanecer perto do limite inferior. Também se verificou que as folgas radiais situam-se em uma região intermediária na simulação teórica, enquanto nas análises experimentais, estas folgas radiais devem estar próximas ao limite inferior. A distância entre mancais é o parâmetro mais significativo para ambas as análises em relação à frequência natural.

7.2 Propostas Futuras

Os algoritmos de otimização multi-objetivos evolutivos representam uma vasta área de pesquisa a ser explorada. Como propostas de trabalhos futuros podemos destacar:

- a substituição das equações polinomiais por equações variacionais;
- a aplicação em sistemas complexos, com montagens não simétricas e estruturas flexíveis;
- a aplicação na própria concepção da montagem experimental, dependendo dos objetivos pretendidos, por exemplo, pontos ótimos de posicionamento de sensores de monitoramento de sinais, ou ainda, transdutores de força para excitação externa de rotores e estruturas não simétricas.

Bibliografia

Araújo, A.L., Herskovits, J., Soares, C.M.M., Pedersen, P. *Parameter Estimation in Active Plate Structures using Gradient Optimization and Neural Networks*. In Proceedings of Inverse Problems, Design and Optimization Symposium, Rio de Janeiro, Brazil, 2004.

Assis, E.G., Steffen Jr, V. *Inverse problems in Rotor Dynamics Using Optimization Techniques*. In Proceedings of the Sixth International Conference on Rotor Dynamics, IFToMM 2002, pp 815-821, 2002.

Baker, J. E. *Adaptive selection methods for genetic algorithms*. In Proceedings of the International Conference on Genetic Algorithms, pp 101-111, 1985.

Baker, J. E. *Reducing bias and inefficiency un the selection algorithm*. In Proceedings of the Second International Conference on Genetic Algorithms, pp 14-21, 1987.

Box, G., Hunter, S., Hunter, W. *Statistics for Experiments*. John Wiley and Sons, 605p, 1998.

Castro, H.F., Cavalca, K. L. *Hybrid Meta-Heuristic Method applied to parameter estimation of a non-linear rotor-bearing system*. In Proceedings of the 7th International Conference on Rotor Dynamics, IFToMM, 2006.

- Castro, H.F., Cavalca, K. L., Nordmann, R. *Rotor-Bearing system instabilities considering a non-linear hydrodynamic model*. In Proceedings of the 7th International Conference on Rotor Dynamics, IFToMM, 2006.
- Cavalca, K. L., Dedini, F. G. *Experimental analysis of a tilting pad journal bearing influence in a vertical rotating system*. In Proceedings of the IFToMM 1998, Darmstadt, Germany, pp 571-582, 1998.
- Cavalca, K. L., Bachschmid, N., Cheli, F. *The relevance of the dynamic behaviour of the supporting structure in calculating critical speeds of multistage centrifugal pumps with interstage seals*. In Proceedings of the ROTORDYNAMICS'92, International Conference on Rotating Machine Dynamics, Venezia, Italia, pp 158-165, 1992.
- Cavalca, K. L., Dedini, F.G., Cheli, F. *Supporting structure effects on rotating machinery vibrations*. In Proceedings of the I MechE, Vibrations in Rotating Machinery – Institution of Mechanical Engineers, Bath, Inglaterra, pp 543-548, 1992.
- Cavalca, K. L., Cavalcante, P.F., Okabe, E.P. *An investigation in the influence of the supporting structure on the dynamics of the rotor system*. In Mechanical System and Signal Processing, Cambridge – UK, v. 19, issue 1, pp 157-174, 2005.
- Cavalcante, P.F., Cavalca, K. L. *Method to analyze the interaction between rotor foundation systems*. In Proceedings of 16th International Modal Analysis Conference, 16th IMAC, pp 775-781, 1998
- Celi, R. *Recent applications of design optimization to rotorcraft – A survey*. In Journal of Aircraft, 1999, vol. 6, pp 176-189.
- Chen, W., Varadarajan S. *Integration of Design of Experiments and Artificial Neural Networks for Achieving Affordable Concurrent Design*. In AIAA. Paper 1997-1230, Collection of Technical Papers, Pt. 2 (A97-24112 05-39)

- Chipperfield, A., Fleming, P. *Gas Turbine Engine Controller Design Using Multiobjective Genetic Algorithms*. In *Genetic Algorithms in Engineering Systems: Innovations and Applications*, 1995. Conference Publication n° 414, IEE.
- Chuang, A.S., Wu, F.F. *An Extensible Genetic Algorithm Framework for Problem Solving in a Common Environment*. In *IEEE Transactions in Power System*, 2000.
- Chung, H.S., Alonso, J.J. *Multiobjective Optimization Using Approximation Model-Based Genetic Algorithms*. In *Proceedings of 10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*. Paper 2004-4325, New York,, Sep 2004.
- Deb, K. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, Chichester, 515p, 2001.
- Deb, K. *Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems*. In *Evolutionary Computation*, Vol. 7(3), pp 205-230, 1999.
- Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. *A fast and elitist multiobjective sorting genetic algorithm: NSGA-II*. Technical Report 200001, Indian Institute of Technology, Kanpur: Kanpur Genetic Algorithm Laboratory (KanGAL), 2000a.
- Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. *A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multiobjective Optimization: NSGA-II*. In *Proceedings of the Parallel Problem Solving from Nature VI (PPSN-VI)*, pp 849-858, 2000b.
- Dias, G.P., Herskovits, J., Rochinha, F.A. *Simultaneous Shape Optimization and Nonlinear Analysis of Elastic Solids*. In *Computational Mechanics*, CIMNE, Barcelona, Spain, 1998.
- Escobar, R.L., Cavalca, K. L. *Dynamic Analysis Prediction using Multi-Objective Genetic Algorithm in rotor-bearing-coupling system*. In *SAE2006 Congress*, Brazil. November, 2006.

- Ewins, D. J. *Modal Testing: Theory and Practice*. RSP John Wiley, 269p, Letchworth 1984.
- Fisher, R.A. *Statistical Methods for Research Workers*. Oliver and Boyd, 1935.
- Fisher, R.A. *The Design of Experiments*. Oliver and Boyd, 1935.
- Fleming, P.F., Purshouse, R.C. *Genetic Algorithms in Control Systems Engineering*. In Technical Report No. 789. Depart. of Automatic Control and Systems Engineering, University of Sheffield, UK, May, 2001.
- Fonseca, C.M., Fleming, P.J. *Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization*. In Proceedings of the Fifth International Conference on Genetic Algorithms, Sao Mateo, CA, July 1993, pp. 416-423
- Fusato, D., Celi, R. *Multidisciplinary Design Optimization for Aeromechanics and Handling Qualities*. Presented at 28th European Rotorcraft Forum, Bristol, UK, 2002.
- Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley Longman, 412p, 1989.
- Hajela, P., Lin, C.Y. *Genetic search strategies in multi-criterion optimal design*. In Structural Optimization 4 (2), pp 99-107, 1992.
- Hann, G., Hoel, R. *Key Challenges for Statisticians in Business and Industry*. In Technometrics 40 (3), pp 210-236, 1998.
- Holland, J.H. *Adaptation In Natural and Artificial Systems*. Ann Arbor, The MIT Press, 1975.
- Horn, J., Nafploits, N., Goldberg, D. *A niched Pareto genetic algorithm for multi-objective optimization*. In Proceedings of the First IEEE Conference on Evolutionary Computation, pp 82-87, 1994.

- Kida, Y., Kawamura, H., Tani, A., Takizawa A. *Multi-Objective Optimization of Spatial Truss Structures by Genetic Algorithm*. In FORMA, Vol. 15, No 2, pp 133-139, 2000.
- Kita, H., Yabumoto, Y., Mori, N., Nishikawa, Y. *Multi-objective optimisation by means of thermodynamical genetic algorithm*. In Proceedings of Parallel Problem Solving from Nature IV, PPSN IV, pp 504-512, 1996.
- Knowles, J., Corne, D. W. *Approximating the non-dominating front using the Pareto archived evolution strategy*. In Evolutionary Computation Journal 8 (2), pp 149-172, 2000.
- Koch, P.N., Mavris, D., Mistree, F. *Partitioned, Multilevel Response Surfaces for Modeling Complex Systems*. In AIAA Journal 2000. Paper 0001-1452, vol.38 no.5, pp 875-881.
- Kundu, S. *A Note on Optimality vs. Stability--A Genetic Algorithm based Approach*. In Proceedings of the Third World Congress of Structural and Multidisciplinary Optimization (WCSMO), Buffalo, New York, May 1999.
- Kursawe, F. *A variant of evolution strategies for vector optimization*. In Parallel Problem Solving from Nature I, PPSN I, pp 193-197, 1990.
- Lallane, M., Ferraris, G. *Rotordynamics: Prediction in Engineering*. John Wiley and Sons, New York, 198p, 1990.
- Laumanns, M., Rudolph, G., Schwefel H.P. *A spatial predator-prey approach to multi-objective optimisation: A preliminary study*. In Proceedings of the Parallel Problem Solving from Nature V, PPSN V, pp 241-249, 1998.
- Lee, M.A, Esbensen, H. *Fuzzy/Multiobjective genetic systems for intelligent systems design tools and components*. In W. Pedrycz (Ed.), Fuzzy Evolutionary Computation, pp 57-80, 1997.

- Maulik, U., Bandyopadhyay, S. *Genetic Algorithm Based Clustering Technique*. In Pattern Recognition 33 (2000), pp 1455-1465.
- Mitchell, M. *Introduction to Genetic Algorithms*. Bradford, MIT Press, 210p, 1996.
- Montgomery, D. C. *Design and analysis of experiments*. John Wiley & Sons. 538p. 1994.
- Montgomery, D. C., Runger, G.C. *Applied Statistics and Probability for Engineers*. John Wiley & Sons. 560p. 1999.
- Murata, T., Ishibuchi, H. *MOGA: Multi-objective genetic algorithms*. In Proceedings of the Second IEEE International Conference on Evolutionary Computation, pp 289-294, 1995.
- Myers, R. H., Montgomery, D. C., *Response Surface Methodology: Process and Product Optimization using Designed Experiments*. John Wiley and Sons, New York, 699p, 1995.
- Okabe, E.P., Cavalca, K. L. *Rotordynamic analysis of systems with a non-linear model of tilting pad bearings*. In Proceedings of the 7th International Conference on Rotor Dynamics, IFToMM, 2006.
- Osyczka, A., Kundu, S. *A new method to solve generalized multicriteria optimisation problems using the simple genetic algorithm*. Structural Optimization 10 (2), pp 94-99, 1995.
- Rajalingham, C., Prabhu, B. S. *The influence of variation of viscosity with temperature on steady state characteristics of journal bearings-simplified analysis*. Tribology International, vol 20, n°5, pp 261-266, 1987.
- Rudolph, G. *Convergence of evolutionary algorithms in general search space*. In Proceedings of Third IEEE Conference on Evolutionary Computation, pp 50-54, 1996.

Rudolph, G. *Evolutionary search under partially ordered fitness sets*. In Proceedings of the International Symposium on Information Science Innovations in Engineering of Natural and Artificial Intelligent Systems, ISI 2001, pp 818-822, 2001.

Sellar, R.S., Batill, S.M., Renaud, J.E. *Response Surface Based, Concurrent Subspace Optimization for Multidisciplinary System Design*. In Proceedings of 34th Aerospace Sciences Meeting and Exhibit, Paper AIAA-1996-714, Reno, NV, Jan 15-18, 1996.

Schaffer, J.D. *Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms*. PhD. Thesis, Nashville, TN: Vanderbilt University, 1984.

Shukla, P.K., Deb, K., Tiwari, S. *Comparing Classical Generating Methods with an Evolutionary Multi-Objective Optimization Method*. In Evolutionary Multi-Criterion Optimization, Third International Conference EMO 2005, pp 311-325, Springer Lecture Notes in computer Science vol 3410, Mexico, 2005.

Silveira, Z.C. *Análise Estatística e Otimização de Parâmetros de Projeto em Componentes de Sistemas Mecânicos*. Tese de Doutorado, Faculdade de Engenharia Mecânica, UNICAMP, 2003.

Silveira, Z.C., Cavalca, K.L. *Statistical analysis and optimization of design parameters in components of rotating machinery*. In Proceedings of the 8th International Conference on Vibrations in Rotating Machinery, ImechE 2004, Wales, Swansea, 2004.

Srinivas, N., Deb, K. *Multiobjective function Optimization using Nondominated Sorting Genetic Algorithms*. In Evolutionary Computation Journal, Vol. 2(3), pp 221-248, 1994.

Steinberg,, D.M., Hunter, W.G. *Experimental design: review and comment*. In Technometrics. v.26, n.2, p. 71-97. May 1984.

- Tapia, A.T. *Modelagem dos acoplamentos mecânicos nos sistemas horizontais rotor-acoplamento-mancal*. In Tese de Doutorado, Faculdade de Engenharia Mecânica, UNICAMP, Campinas, 250p, 2003.
- Tapia, A. T., Cavalca, K. L., Dedini, F. G. *Flexible Coupling Modelling Effect In Mechanical System*. XVI Engineering Brazilian Congress, COBEM2001, Uberlandia-Brazil, pp 44-53, 2001.
- Tapia, A. T., Escobar, R. L., Cavalca, K. L. *Dynamic Analysis of Flexible Couplings in rotor-bearing-coupling systems*. In SAE2003 Congress, Brazil, November, 2003.
- Veldhuizen, D. V. *Multiobjective Evolutionary Algorithms: Classifications, Analysis and New Innovations*. PhD. Thesis, Dayton, OH: Air Force Institute of Technology. Technical Report AFIT/DS/ENG/99-01, 1999.
- Zitzler, E., Thiele, L. *An evolutionary algorithm for multiobjective optimization: The strength Pareto approach*. Technical report 43, Zürich, Switzerland: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), 1998a
- Zitzler, E., Thiele, L. *Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Case Study*. In Parallel Problem Solving from Nature – PPSN V, Amsterdam, The Netherlands, pp 292-30, 1998b
- Zitzler, E., Deb, K., Thiele, L. *Comparison of Multiobjective Evolutionary Algorithms: Empirical Results*. In Evolutionary Computation Journal 8 (2), pp 125-148, 2000.