

UNIVERSIDADE TECNOLOGIA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL

RICHARD ADERBAL GONÇALVES

ALGORITMOS CULTURAIS PARA O PROBLEMA DO
DESPACHO DE ENERGIA ELÉTRICA

TESE

CURITIBA

2010

RICHARD ADERBAL GONÇALVES

**ALGORITMOS CULTURAIS PARA O PROBLEMA DO
DESPACHO DE ENERGIA ELÉTRICA**

Tese de doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná como requisito parcial para a obtenção do título de Doutor em Ciências. Área de Concentração: Informática Industrial.

Orientadora: Profa. Dra. Myriam Regattieri Delgado
Co-orientador: Prof. Dr. Marco Cesar Goldberg

CURITIBA

2010

Ficha catalográfica elaborada pela Biblioteca da UTFPR – Campus Curitiba

G635a Gonçalves, Richard Aderbal
Algoritmos culturais para o problema do despacho de energia elétrica / Richard Aderbal Gonçalves. – 2010.
202 f. : il. ; 30 cm

Orientador: Myriam Regattieri Delgado
Co-orientador: Marco Cesar Goldbarg
Tese (Doutorado) – Universidade Tecnológica Federal do Paraná.
Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial. Área de Concentração: Informática Industrial, Curitiba, 2010
Bibliografia: f. 173-93

1. Sistemas de energia elétrica – Aspectos econômicos. 2. Sistemas imunológicos artificiais. 3. Algoritmos culturais. 4. Otimização multiobjetivo. 5. Informática industrial. I. Delgado, Myriam Regattieri De Biase da Silva, orient. II. Goldbarg, Marco Cesar, co-orient. III. Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial. Área de Concentração em Informática Industrial. IV. Título.

CDD 621.3

Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial

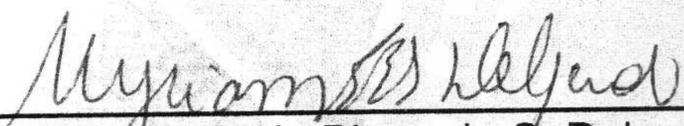
Título da Tese Nº 50:

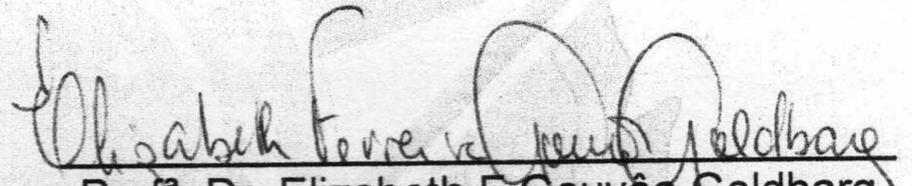
“Algoritmos Culturais para o Problema do Despacho de Energia Elétrica”

por

Richard Aderbal Gonçalves

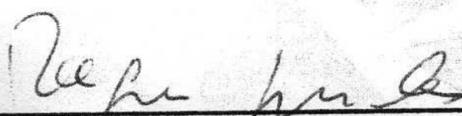
Esta tese foi apresentada, às 08h30min do dia 25 de fevereiro de 2010, como requisito parcial para a obtenção do título de **DOUTOR EM CIÊNCIAS** – Área de Concentração: **Informática Industrial**, pelo Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial – CPGEI – da Universidade Tecnológica Federal do Paraná – UTFPR. O trabalho foi aprovado pela Banca Examinadora, composta pelos professores:

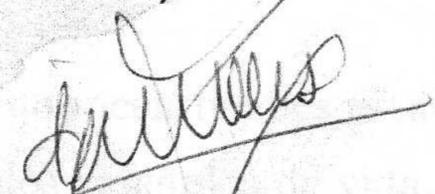

Prof^a. Dr. Myriam R. de Biase da S. Delgado
(Orientadora – UTFPR)


Prof^a. Dr. Elizabeth F. Gouvêa Goldberg
(UFRN)

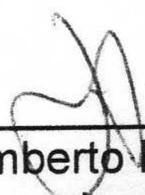

Prof. Dr. Fabio Hernandez
(UNICENTRO)


Prof. Dr. Leandro dos Santos Coelho
(PUC-PR)


Prof. Dr. Roger Gules
(UTFPR)


Prof. Dr. Gustavo Alberto Gimenez Lugo
(Suplente - UTFPR)

Visto da coordenação:


Prof. Dr. Humberto Remígio Gamba
(Coordenador do CPGEI)

Aos meus pais, pelo contínuo incentivo aos estudos e aperfeiçoamento e pelos exemplos de vida.

À Carol, pelo carinho, dedicação e afeto.

AGRADECIMENTOS

Primeiramente gostaria de agradecer a minha orientadora, Profa. Dra. Myriam Regattieri De Biase da Silva Delgado, pelo carinho, dedicação e paciência demonstrados ao longo do período de realização deste trabalho. Não poderia ter tido uma orientadora melhor!

Gostaria de agradecer também ao meu co-orientador, Prof. Dr. Marco César Goldberg, e sua esposa, Profa. Dra. Elizabeth Ferreira Gouvêa Goldberg, pelas valiosas contribuições para a melhoria deste trabalho, pelas experiências compartilhadas e pela hospitalidade durante o período que eu e Carol passamos em Natal.

Agradeço à Carol, amor da minha vida, pelo companheirismo, dedicação e trocas de idéias.

Aos meus pais, pelo carinho, apoio, dedicação e pelas oportunidades propiciadas.

Aos colegas do DECOMP que possibilitaram meu afastamento para doutoramento.

Aos colegas de LASD, em particular ao Luís Mendes e ao Neto, pela configuração e manutenção do *cluster* de computadores utilizado durante a realização de parte dos experimentos.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo auxílio financeiro para o doutorado-sanduíche em Natal.

Aos membros da banca por terem gentilmente aceito o convite para a participação da avaliação deste trabalho.

A todos aqueles que, direta ou indiretamente, participaram da realização deste trabalho.

Culture is the arts elevated to a set of beliefs.
(WOLFE, Thomas).

Cultura é a arte elevada a um conjunto de
crenças. (WOLFE, Thomas).

Resumo

GONÇALVES, Richard Aderbal. **Algoritmos Culturais para o Problema do Despacho de Energia Elétrica.** 2010. 202f. Tese (Doutorado em Informática Industrial) - Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2010.

Nesta tese, os Sistemas Imunológicos Artificiais são aplicados a diferentes instâncias do despacho econômico e econômico/ambiental de energia elétrica. Os sistemas imunes considerados são baseados no princípio da seleção clonal e usam uma representação real com operador de *aging* puro e operadores de hipermutação utilizando distribuições de probabilidade Gaussianas e de Cauchy. Algoritmos Culturais utilizando fontes de conhecimento normativo, situacional, histórico e topográfico são incorporados para melhorar a capacidade de otimização global dos sistemas imunes. Todas as abordagens propostas possuem vários pontos de auto-adaptação e a maioria utiliza um operador de busca local baseado na técnica quase-simplex. Uma sequência caótica também é considerada como uma potencial fonte de melhoria na variação cultural do algoritmo. Procedimentos de reparação constituem outra contribuição do trabalho e são aplicados para evitar lidar com soluções ineficazes em todos os problemas abordados. Na primeira parte dos experimentos, quatro instâncias do problema do despacho econômico de energia são consideradas. Em todos os casos, foi utilizada uma função não suave de custo de combustível levando em consideração os efeitos de ponto de válvula. Uma das instâncias também considera as perdas na transmissão de energia. Nos experimentos de comparação entre as abordagens propostas, as versões imuno-culturais superam a versão puramente imune. O método cultural proposto que apresenta melhor resultado é escolhido para ser comparado a outras técnicas modernas de otimização reportadas na literatura recente. Em todos os casos mono-objetivo considerados, a abordagem proposta é capaz de encontrar o menor custo de combustível. A segunda parte dos experimentos trata do problema do despacho econômico/ambiental. Esta é uma versão do problema do despacho econômico de energia onde a emissão de poluentes é adicionada como um novo objetivo, tornando este um problema de otimização multiobjetivo não-linear com restrições. Algoritmos imuno-culturais baseados em fatores de escalarização e dominância de Pareto são propostos para este caso. Várias instâncias do problema são utilizadas nos experimentos, algumas das quais consideram perdas na transmissão de energia. Os algoritmos propostos são favoravelmente comparados com um algoritmo do estado-da-arte para otimização multiobjetivo. O melhor algoritmo proposto também é comparado com métodos reportados na literatura recente. As comparações mostram o bom desempenho da melhor abordagem proposta e confirmam seu potencial para resolver o problema do despacho econômico/ambiental de energia.

Palavras-chave: Despacho Econômico de Energia, Despacho Econômico/Ambiental de Energia, Sistemas Imunológicos Artificiais, Algoritmos Culturais, Otimização Multiobjetivo.

Abstract

GONÇALVES, Richard Aderbal. **Cultural Algorithms for the Electrical Energy Dispatch**. 2010. 202f. Tese (Doutorado em Informática Industrial) - Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná. Curitiba, 2010.

In this thesis, Artificial Immune Systems are applied to solve different instances of the economic and environmental/economic load dispatch problems. The immune systems considered here are based on the clonal selection principle and use a real coded representation with pure aging operator and hypermutation operators utilizing Gaussian and Cauchy distributions. Cultural Algorithms using normative, situational, historical and topographical knowledge sources are incorporated to improve the global optimization capability of immune systems. All the proposed approaches have several points of self-adaptation and most of them use a local search operator that is based on a quasi-simplex technique. A chaotic sequence is also considered as a potential source of improvement to the cultural variation. Repair procedures represent another contribution of this work and are applied to avoid dealing with infeasible solutions in all the considered problems. In the first part of the experiments, four instances of the economic load dispatch problem are considered. In all the cases, a non-smooth fuel cost function which takes into account the valve-point loading effects is utilized. One of instances also considers energy transmission losses. In the experiments conducted to compare the proposed approaches, the immune-cultural based approaches outperformed the pure immune version. The proposed cultural method which presents the best performance is chosen to be compared with other modern optimization techniques reported in the recent literature. In all the mono-objective cases considered, the proposed approach is capable of finding the minimum fuel cost value. The second part of the experiments deals with the environmental/economic load dispatch problem. This is a multi-objective version of the economic load dispatch where pollution emission is added as an objective, it is formulated as a non-linear constrained multi-objective optimization problem. Cultural immune algorithms based on scalarizing factors and Pareto-dominance are proposed for this case. Several instances of the problem are considered, some dealing with energy transmission losses. The proposed algorithms are favorably compared with a state-of-art algorithm for multi-objective optimization (the Non-dominated Sorting Genetic Algorithm II - NSGA-II). The best proposed algorithm is also compared with methods reported in recent literature. The comparisons demonstrate the good performance of the best proposed approach and confirm its potential to solve the environmental/economic load dispatch problem.

Keywords: Economic Load Dispatch, Environmental/Economic Load Dispatch, Artificial Immune Systems, Cultural Algorithms, Multi-objective Optimization.

Lista de Figuras

1	Camadas da plataforma para sistemas imunológicos artificiais	27
2	Princípio da seleção clonal	30
3	Memória imunológica	31
4	Maturação da afinidade	32
5	Execução de um algoritmo cultural com três populações, dois nichos e um meta-espaço de crenças	39
6	Funcionamento básico de um algoritmo cultural	40
7	Representação de conhecimento situacional	44
8	Representação de conhecimento normativo	44
9	Representação alternativa de conhecimento normativo	45
10	Representação do conhecimento topográfico	46
11	Representação alternativa do conhecimento topográfico	47
12	Representação de conhecimento histórico	48
13	Protocolo Votar-Herdar-Promover	50
14	Resultado da comparação entre três amostras.	62
15	Representação de um espaço de decisões de variáveis e o seu correspondente espaço objetivo	64
16	Possível <i>trade-off</i> para o problema de decisão da compra de um carro	65
17	Vetores objetivo ideal (z^*), utopia (z^{**}) e nadir (z^{nad})	66
18	Exemplos de relações de dominância entre soluções no espaço objetivo	67
19	Exemplo do uso de função de conquista	75
20	Exemplo do indicador hipervolume	77
21	Custo de um gerador sem efeito de ponto de válvula.	86
22	Custo de um gerador com efeito de ponto de válvula.	86
23	Representação do conhecimento histórico	108
24	Árvore binária representando as regiões no conhecimento topográfico.	108
25	Representação do conhecimento topográfico após 2 divisões.	109
26	Representação do espaço multiobjetivo para uma versão escalarizada	113
27	Divisão do espaço de crenças do algoritmo SICMO _p	116
28	Médias dos <i>ranks</i> para diferentes configurações de tamanho de população.	126
29	Médias dos <i>ranks</i> para diferentes configurações de taxa de aceitação.	127
30	Médias dos <i>ranks</i> para diferentes configurações de número de anticorpos mantidos no conhecimento situacional.	127
31	Médias dos <i>ranks</i> para diferentes configurações de idade máxima dos anticorpos.	128
32	Médias dos <i>ranks</i> para diferentes configurações de número de clones por anticorpo.	129

33	Resultados estatísticos para o teste Kruskal-Wallis para o caso com 13 geradores e demanda de 1800 MW, mostrando que o SIC e o SIC _c obtiveram melhores resultados do que o SI.	131
34	Comportamento do algoritmo SIC	133
35	Médias dos <i>ranks</i> para as configurações com e sem a busca local.	134
36	Médias dos <i>ranks</i> para diferentes configurações de função de influência principal.	135
37	Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 13 geradores e demanda de 2520 MW, mostrando que o SIC _c é melhor do que o SIC e o SI.	137
38	Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 40 geradores e demanda de 10500 MW, mostrando que o SIC e o SIC _c obtiveram melhores resultados do que o SI.	140
39	Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 20 geradores e demanda de 2500 MW, mostrando que o SIC obteve melhores resultados.	144
40	Multiobjetivo - Estudo de Caso 1: Exemplos de Conjunto de Aproximação Obtidos	149
41	Probabilidades de aplicação das funções de influência no SICMO _e	150
42	Probabilidades de aplicação das funções de influência no algoritmo SICMO _p	150
43	Valores Mínimos e Médios para o Custo e a Emissão	151
44	Multiobjetivo - Estudo de Caso 2 - Exemplos dos Conjuntos de Aproximação Obtidos	154
45	Multiobjetivo - Estudo de Caso 3 - Exemplos dos Conjuntos de Aproximação Obtidos	157
46	Multiobjetivo - Estudo de Caso 4: Exemplos dos Conjuntos de Aproximação Obtidos	160
47	Multiobjetivo - Estudo de Caso 5 - Exemplos dos Conjuntos de Aproximação Obtidos	163
48	Multiobjetivo - Estudo de Caso 6 - Exemplos dos Conjuntos de Aproximação Obtidos	167

Lista de Tabelas

1	Relações de dominância	67
2	Resultados para o ajuste do tamanho da população. Os custos são dados em US\$	125
3	Resultados para o ajuste da taxa aceitação - $accept_{perc}$. Os custos são dados em US\$	126
4	Resultados para o ajuste número de melhores - $pbest$. Os custos são dados em US\$	127
5	Resultados para o ajuste da idade máxima dos anticorpos - τ_B . Os custos são dados em US\$	128
6	Resultados para o ajuste do número de clones - dup . Os custos são dados em US\$	128
7	Parâmetros utilizados nos algoritmos propostos	130
8	Resultados das abordagens propostas para o estudo de caso 1. Os custos são dados em US\$	131
9	Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 13 geradores e demanda de 1800 MW	132
10	Efeito da busca local Quase-Simplex. Os custos são dados em US\$	134
11	Efeito da função de influência principal proposta. Os custos são dados em US\$	134
12	Resultados da comparação com a literatura para o estudo de caso 1. Os custos são dados em US\$	136
13	Resultados das abordagens propostas para o estudo de caso 2. Os custos são dados em US\$	137
14	Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 13 geradores e demanda de 2520 MW	138
15	Resultados da comparação com a literatura para o estudo de caso 2. Os custos são dados em US\$	138
16	Resultados das abordagens propostas para o estudo de caso 3. Os custos são dados em US\$	139
17	Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 40 geradores e demanda de 10500 MW	141
18	Resultados da comparação com a literatura para o estudo de caso 3. Os custos são dados em US\$	142
19	Mono-objetivo - Estudo de Caso 4 - Custos em US\$	143
20	Melhor Resultado para o caso com 20 geradores, considerando perdas, e demanda de 2500 MW usando o SIC.	145
21	Mono-objetivo - Estudo de Caso 4 - Comparação com a Literatura	145
22	Multiobjetivo - Estudo de Caso 1 - Solução com o melhor valor para o custo de combustível	147

23	Multiobjetivo - Estudo de Caso 1 - Solução com o melhor valor para a emissão de poluentes	148
24	Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador hipervolume.	148
25	Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador épsilon unário.	148
26	Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador R_2	148
27	Multiobjetivo - Estudo de Caso 1: Solução que apresenta o melhor valor para o custo de combustível	152
28	Multiobjetivo - Estudo de Caso 1: Solução que apresenta o melhor valor para a emissão de poluentes	152
29	Multiobjetivo - Estudo de Caso 2: Solução que apresenta o melhor valor para o custo de combustível	153
30	Multiobjetivo - Estudo de Caso 2: Solução que apresenta o melhor valor para a emissão de poluentes	153
31	Multiobjetivo - Estudo de Caso 2: P-valores para o indicador hipervolume.	153
32	Multiobjetivo - Estudo de Caso 2: P-valores para o indicador épsilon unário.	153
33	Multiobjetivo - Estudo de Caso 2: P-valores para o indicador R_2	153
34	Multiobjetivo - Estudo de Caso 2 - Solução que apresenta o melhor valor para o custo de combustível	154
35	Multiobjetivo - Estudo de Caso 2 - Solução que apresenta o melhor valor para a emissão de poluentes	155
36	Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para o custo de combustível	156
37	Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para a emissão de poluentes	156
38	Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador hipervolume.	156
39	Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador épsilon unário.	156
40	Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador R_2	156
41	Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para o custo de combustível	157
42	Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para a emissão de poluentes	158
43	Multiobjetivo - Estudo de Caso 4: Solução que apresenta o melhor valor para o custo de combustível	158
44	Multiobjetivo - Estudo de Caso 4: Solução que apresenta o melhor valor para a emissão de poluentes	159
45	Multiobjetivo - Estudo de Caso 4: P-valores para o indicador hipervolume.	159
46	Multiobjetivo - Estudo de Caso 4: P-valores para o indicador épsilon unário.	159
47	Multiobjetivo - Estudo de Caso 4: P-valores para o indicador R_2	159
48	Multiobjetivo - Estudo de Caso 4 - Solução que apresenta o melhor valor para o custo de combustível	160
49	Multiobjetivo - Estudo de Caso 4 - Solução que apresenta o melhor valor para a emissão de poluentes	161
50	Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para o custo de combustível	162

51	Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para a emissão de poluentes	162
52	Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador hipervolume.	162
53	Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador épsilon unário.	163
54	Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador R_2	163
55	Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para o custo de combustível	164
56	Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para a emissão de poluentes	164
57	Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para o custo de combustível	165
58	Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para a emissão de poluentes	165
59	Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador hipervolume.	166
60	Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador épsilon unário.	166
61	Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador R_2	166
62	Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para o custo de combustível	167
63	Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para a emissão de poluentes	168
64	Mono-objetivo - Estudo de Caso 1 e 2 - Coeficientes do custo de combustível.	194
65	Mono-objetivo - Estudo de Caso 3 - Coeficientes do custo de combustível. .	195
66	Mono-objetivo - Estudo de Caso 4 - Coeficientes do custo de combustível. .	196
67	Mono-objetivo - Estudo de Caso 4 - Coeficientes de perda na transmissão de energia - Matriz B - Devem ser multiplicados por $1 * 10^{-5}$	197
68	Multiobjetivo - Estudo de Caso 1, 2 e 3 - Coeficientes do custo de combustível.	198
69	Multiobjetivo - Estudo de Caso 1, 2 e 3 - Coeficientes da emissão de poluentes.	198
70	Multiobjetivo - Estudo de Caso 2 - Coeficientes de perda na transmissão de energia - Matriz B, Vetor B_0 e Coeficiente B_{00}	198
71	Multiobjetivo - Estudo de Caso 3 - Coeficientes de perda na transmissão de energia - Matriz B	198
72	Multiobjetivo - Estudo de Caso 4 - Coeficientes do custo de combustível. .	199
73	Multiobjetivo - Estudo de Caso 4 - Coeficientes da emissão de poluentes. .	199
74	Multiobjetivo - Estudo de Caso 4 - Coeficientes de perda na transmissão de energia - Matriz B	199
75	Multiobjetivo - Estudo de Caso 5 e 6 - Coeficientes do custo de combustível.	199
76	Multiobjetivo - Estudo de Caso 5 - Coeficientes da emissão de poluentes. .	200
77	Multiobjetivo - Estudo de Caso 5 e 6 - Coeficientes de perda na transmissão de energia - Matriz B, Vetor B_0 e Coeficiente B_{00}	201
78	Multiobjetivo - Estudo de Caso 6 - Coeficientes da emissão de poluentes. .	202

Lista de Algoritmos

1	Forma básica do algoritmo Clonalg	p. 33
2	Plataforma básica de um algoritmo cultural	p. 40
3	Algoritmo imunológico proposto	p. 95
4	Procedimento de reparo - sem perdas	p. 99
5	Procedimento de reparo - com perdas	p. 101
6	Algoritmo imuno-cultural proposto	p. 102
7	Algoritmo SICMO _e	p. 114
8	Algoritmo SICMO _p	p. 114
9	Algoritmo da arquivo de grade adaptativa	p. 119

Lista de abreviaturas e siglas

PDE	Problema do Despacho de Energia	p. 23
AC	Algoritmo Cultural	p. 26
AE	Algoritmo Evolucionário	p. 26
AEMO	Algoritmo Evolucionário Multiobjetivo	p. 27
AGA	Adaptive Grid Archiving	p. 51
CN	Computação Natural,	p. 23
POM	Problema de Otimização Multiobjetivo	p. 19
SIA	Sistema Imunológico Artificial	p. 23
SIAMO	Sistema Imunológico Artificial Multiobjetivo	p. 27
ACMO	Algoritmo Cultural Multiobjetivo	p. 27

Sumário

1	Introdução	19
1.1	Objetivos	21
1.2	Contribuições do Trabalho	22
1.3	Estrutura da Tese	23
2	Algoritmos Evolucionários	24
2.1	Sistemas Imunológicos Artificiais	26
2.1.1	Paradigmas	27
2.1.2	Princípio da Seleção Clonal	29
2.1.2.1	Clonalg	32
2.1.3	Aplicações	34
2.2	Algoritmos Culturais	36
2.2.1	Inspiração Natural	37
2.2.2	Plataforma Básica de Algoritmos Culturais	37
2.2.3	Principais Características	40
2.2.4	Microevolução × Macroevolução	42
2.2.4.1	Espaço Populacional	42
2.2.4.2	Espaço de Crenças e Meta-Espaço de Crenças	42
2.2.4.3	Tipos de Conhecimento	43
2.2.4.4	Conhecimento Situacional	43
2.2.4.5	Conhecimento Normativo	44
2.2.4.6	Conhecimento do Domínio	45
2.2.4.7	Conhecimento Topográfico	46
2.2.4.8	Conhecimento Histórico	48
2.2.5	<i>Exames Culturais</i>	49
2.2.6	Protocolos de Intercomunicação	49
2.2.6.1	Funções de Aceitação	50
2.2.6.2	Funções de Influência	51
2.2.7	Aplicações de Algoritmos Culturais	52
2.2.7.1	Suporte a Re-engenharia de Sistemas Especialistas	53
2.2.7.2	Otimização Usando Algoritmos Culturais Fuzzy	53
2.2.7.3	Otimização com Restrições usando Algoritmos Culturais	54
2.2.7.4	Modelagem do Impacto da Cultura nas Interações Sociais e na Solução de Problemas	55
2.2.7.5	Escalonamento de Tarefas	56
2.2.7.6	Tecido Social	56

3	Otimização por Algoritmos Evolucionários	58
3.1	Princípios de Otimização Mono-objetivo	58
3.2	Algoritmos Evolucionários Mono-objetivo	59
3.2.1	Testes Estatísticos	60
3.3	Princípios de Otimização Multiobjetivo no Contexto de AEs	63
3.3.1	Dominância e Pareto-optimalidade	65
3.3.1.1	Soluções Especiais	66
3.3.1.2	Pareto Dominância e Fronteiras não-dominadas	67
3.3.2	Métodos não Baseados em Dominância de Pareto	68
3.3.2.1	Soma Ponderada	68
3.3.2.2	ϵ -Constraint	69
3.4	Algoritmos Evolucionários Multiobjetivo	70
3.4.1	Características dos Algoritmos Evolucionários Multiobjetivo	71
3.4.1.1	Avaliação e Seleção de Soluções em AEMO	72
3.4.1.2	Diversidade da População de AEMO	72
3.4.1.3	Elitismo	73
3.4.2	Análise e Comparação de Resultados em AEMO	73
3.4.2.1	<i>Ranking</i> de Dominância	74
3.4.2.2	Função Empírica de Conquista (<i>Empirical Attainment Function</i>)	74
3.4.2.3	Indicadores de Qualidade sobre a Fronteira de Pareto	76
3.4.3	Classificação de Algoritmos Evolucionários Multiobjetivo	78
3.4.4	Exemplos de Algoritmos Evolucionários Multiobjetivo	78
3.4.4.1	Sistemas Imunológicos Artificiais Multiobjetivo	80
3.4.4.2	Algoritmos Culturais Multiobjetivo	82
4	Problema do Despacho de Energia Elétrica	85
4.1	Despacho Econômico de Energia Elétrica	85
4.1.1	Problema do Despacho de Energia considerando Perdas	87
4.2	Problema do Despacho Econômico/Ambiental de Energia	88
4.3	Abordagens para os Problemas de Despacho de Energia	89
4.3.1	Abordagens para o Problema do Despacho Econômico de Energia	89
4.3.2	Abordagens para o Problema do Despacho Econômico/Ambiental de Energia	91
5	Algoritmos Culturais para o Problema do Despacho de Energia	94
5.1	O Sistema Imune Proposto	94
5.2	O Sistema Imuno-cultural Proposto	102
5.2.1	Protocolos de Comunicação: Funções de Aceitação e de Influência	103
5.2.2	Fontes de Conhecimento	104
5.3	O Sistema Imuno-Cultural Caótico Proposto	111
5.4	O Sistema Imuno-Cultural Multiobjetivo Proposto	112
5.4.1	Sistema Imuno-Cultural Multiobjetivo Escalarizado	112
5.4.2	Sistema Imuno-Cultural Multiobjetivo Baseado em Dominância de Pareto	113
5.4.2.1	Grade Adaptativa	118
5.4.3	Sistema Imuno-Cultural Caótico Multiobjetivo Baseado em Dominância de Pareto	120
5.4.4	Comparação com a Literatura	120
5.4.4.1	Algoritmos não Baseados em Dominância de Pareto	120
5.4.4.2	Algoritmos Baseados em Dominância de Pareto	121

6	Simulações e Resultados Obtidos	123
6.1	Ajustes de Parâmetros	125
6.1.1	Ajuste do Tamanho da População	125
6.1.2	Ajuste da Taxa Aceitação - accept_{perc}	126
6.1.3	Ajuste do Número de Melhores - $pbest$	126
6.1.4	Ajuste da Idade Máxima dos Anticorpos - τ_B	128
6.1.5	Ajuste do Número de Clones	128
6.1.6	Parâmetros Utilizados nos Experimentos	129
6.2	Resultados para o Despacho Econômico de Energia	129
6.2.1	Mono-objetivo - Estudo de Caso 1	130
6.2.1.1	Comparação entre as abordagens propostas	130
6.2.2	Análise do Algoritmo SIC	132
6.2.2.1	Comportamento dos Conhecimentos no Algoritmo SIC	133
6.2.2.2	Efeito da Busca Local Quase-Simplex no Algoritmo SIC	133
6.2.2.3	Efeito da Função de Influência Principal Proposta no Algoritmo SIC	134
6.2.2.4	Comparação com as abordagens da literatura	135
6.2.3	Mono-objetivo - Estudo de Caso 2	135
6.2.3.1	Comparação entre as abordagens propostas	137
6.2.3.2	Comparação com as abordagens da literatura	138
6.2.4	Mono-objetivo - Estudo de Caso 3	139
6.2.4.1	Comparação entre as abordagens propostas	139
6.2.4.2	Comparação com as abordagens da literatura	140
6.2.5	Mono-objetivo - Estudo de Caso 4	143
6.2.5.1	Comparação entre as abordagens propostas	143
6.2.5.2	Comparação com as abordagens da literatura	144
6.2.6	Conclusões sobre os Estudos de Caso Mono-objetivo	145
6.3	Resultados para o Despacho Econômico/Ambien-tal de Energia	146
6.3.1	Multiobjetivo - Estudo de Caso 1	147
6.3.1.1	Influência dos Conhecimentos nos Algoritmos Imuno-Culturais	149
6.3.1.2	Comparação com os Algoritmos da Literatura	151
6.3.2	Multiobjetivo - Estudo de Caso 2	152
6.3.3	Multiobjetivo - Estudo de Caso 3	155
6.3.4	Multiobjetivo - Estudo de Caso 4	158
6.3.5	Multiobjetivo - Estudo de Caso 5	161
6.3.6	Multiobjetivo - Estudo de Caso 6	165
6.3.7	Conclusões sobre os Estudos de Caso Multiobjetivo	168
7	Conclusões e Trabalhos Futuros	170
	Referências Bibliográficas	173
	Apêndice A – Coeficientes Utilizados pelos Estudos de Caso	194
A.1	Dados dos Estudos de Caso Mono-objetivo	194
A.2	Dados dos Estudos de Caso Multiobjetivo	198

1 *Introdução*

As questões econômicas estão presentes na maioria dos problemas práticos abordados na área de otimização. Otimizar recursos de forma a reduzir os custos de operação e aumentar o lucro do sistema é um dos principais objetivos para aqueles que trabalham nesta área de pesquisa. No entanto, nos últimos anos vem crescendo a preocupação com questões ambientais, as quais passaram a ser reconhecidas como parte determinante dos aspectos econômicos. Assim, nesta tese são abordados dois tipos de problemas de otimização no contexto de geração de energia elétrica: o despacho econômico e o despacho econômico/ambiental de energia.

O objetivo principal do despacho econômico de energia é determinar a quantidade ótima de energia que deve ser produzida por cada unidade geradora de maneira a atender à demanda requisitada e às restrições operacionais de cada gerador com o menor custo operacional possível (PARK; JEONG; LEE, 2006).

A maioria das formulações recentes para o problema do despacho econômico de energia leva em consideração os efeitos de ponto de válvula, o que torna a função que descreve o custo operacional da produção de energia altamente não-linear (COELHO; MARIANI, 2009). Considerando a não-linearidade do problema, os métodos clássicos (exatos) de otimização se mostra ineficiente na resolução do problema; principalmente se as perdas de energia na distribuição forem consideradas, o que introduz uma restrição de igualdade não-linear ao problema (PALANICHAMY; BABU, 2008).

Na ausência de métodos exatos eficientes, os métodos estocásticos (particularmente aqueles que fazem uso de técnicas de computação natural - as quais são técnicas de computação inspiradas em princípios naturais) se tornaram opções viáveis para a solução do problema do despacho econômico de energia. Exemplos de métodos de computação natural aplicados ao problema do despacho econômico de energia são: sistemas imunológicos artificiais (LIU et al., 2006; GONÇALVES et al., 2007; GONÇALVES et al., 2009), algoritmos genéticos (LING et al., 2003), programação evolutiva (CHAKRABARTI; CHATTOPADHYAY, 2003), enxame de partículas (VICTOIRE; JEYAKUMAR, 2004; HOU et al., 2005; COELHO; MARIANI, 2008; SELVAKUMAR; THANUSHKODI, 2007; SINHA;

PURKAYASTHA, 2004; MENG et al., 2010; CHATURVEDI; PANDIT; SRIVASTAVA, 2009; VLACHOGIANNIS; LEE, 2009), evolução diferencial (COELHO; MARIANI, 2006; PEREZ-GUERRERO; CEDENIO-MALDONADO, 2005; WANG; CHIOU; LIU, 2007; NOMAN; IBA, 2008; COELHO; SOUZA; MARIANI, 2009; BALAMURUGAN; SUBRAMANIAN, 2007) e algoritmos culturais (GONÇALVES et al., 2007; COELHO; ALMEIDA; MARIANI, 2008; GONÇALVES et al., 2009; COELHO; SOUZA; MARIANI, 2009).

A crescente preocupação da opinião pública com questões ambientais motivou uma variação do despacho econômico de energia onde a questão da emissão de poluentes é tratada como um objetivo adicional do problema, ou seja, os objetivos passam a ser minimizar o custo operacional e a emissão de poluentes (ABIDO, 2009).

Existem várias maneiras de se lidar com o problema do despacho econômico/ambiental de energia, mas em linhas gerais elas podem ser divididas em três categorias: tratar a emissão de poluentes como uma restrição do problema (conhecido por problema do despacho econômico de energia com restrições de emissão (GUERRERO, 2004)), resolver múltiplos problemas mono-objetivo (através de técnicas de escalarização ou ϵ – *constraints*, por exemplo) ou considerando o problema na sua formulação multiobjetivo.

A vantagem da primeira opção é que ela pode fazer uso de técnicas para otimização de problemas mono-objetivo com restrições. A sua principal desvantagem é a dificuldade de se determinar os efeitos que a emissão possui sobre o custo de geração de energia, ou seja, é difícil encontrar soluções com diferentes balanceamentos entre os objetivos considerados (estas soluções são chamadas soluções-compromisso), visto que apenas uma otimização mono-objetivo é aplicada. A segunda abordagem possui a mesma vantagem da primeira, mas pode encontrar soluções-compromisso boas; pois executa diferentes versões do problema mono-objetivo, cada uma com um compromisso diferente. Sua desvantagem reside no custo computacional de se executar várias vezes um otimizador mono-objetivo (uma para cada vetor de escalarização ou de restrições). A terceira abordagem possui como vantagem a capacidade de encontrar boas soluções-compromisso a um custo computacional, geralmente, mais baixo do que a segunda. A desvantagem desta abordagem reside no fato de que otimizadores multiobjetivo são mais complexos de se projetar. Além disto, a literatura de otimizadores multiobjetivo baseados em técnicas de computação natural é menos vasta do que sua contrapartida para otimizadores mono-objetivo. Nesta tese a segunda e a terceira abordagem serão adotadas.

Técnicas de computação natural aplicadas na resolução do despacho econômico/ambiental podem ser encontradas em (DAS; PATVARDHAN, 1998; ABIDO, 2003b; ABIDO, 2003a; ABIDO, 2003b; ABIDO, 2003; VENKATESH; GNANADASS; PADHY, 2003; CAIA et al., 2009; KING; RUGHOPUTH; DEB, 2005; ABIDO, 2009; CETINKAYA, 2009).

Tanto o despacho econômico quanto o despacho econômico/ambiental serão tratados utilizando sistemas imunológicos artificiais (DE CASTRO; VON ZUBEN, 2002; CUTELLO; NICOSIA; PAVONE, 2004a; TIMMIS et al., 2004) e algoritmos culturais (REYNOLDS; PENG, 2005; REYNOLDS; ALI, 2008b).

O uso de sistemas imunológicos artificiais foi, em parte, inspirado pela prova de que, sob certas condições, os sistemas imunológicos artificiais são capazes de convergir para o ótimo global de problemas de otimização (CUTELLO et al., 2007). Os algoritmos imunológicos propostos são baseado no princípio de seleção clonal (DE CASTRO; VON ZUBEN, 2002) e possuem algumas características peculiares tais como o uso de operador de *aging* (CUTELLO et al., 2005a; CUTELLO et al., 2006; ALMEIDA; GONÇALVES; DELGADO, 2007), de uma busca local quase-simplex e de dois tipos de hipermutação (Cauchy e Gaussiana).

Para melhorar o desempenho do sistema imunológico artificial, ele foi integrado como espaço populacional de um algoritmo cultural (REYNOLDS, 2003; REYNOLDS; PENG, 2005; BECERRA; COELLO, 2006a). Foram utilizados os conhecimentos situacional, normativo, histórico e topográfico (SALEEM, 2001; ALI, 2008).

1.1 Objetivos

O objetivo principal da tese é o desenvolvimento de algoritmos imuno-culturais adequados para a resolução de problemas de despacho econômico e econômico/ambiental de energia.

Outros objetivos da tese são:

- Combinar as vantagens das diferentes funções de influência principal propostas na literatura para a criação de uma nova função de influência principal para os algoritmos culturais propostos;
- Propor procedimentos eficientes de reparação de soluções ineficazes tanto para os problemas que não consideram perdas de energia quanto para aqueles que consideram as perdas;
- Determinar se o uso da busca local quase-simplex pode melhorar os resultados do sistema imuno-cultural;
- Determinar se o uso de sequências caóticas em conjunto com algoritmos imuno-culturais pode ser vantajoso;

- Analisar o impacto das fontes de conhecimento nos resultados dos algoritmos propostos.

1.2 Contribuições do Trabalho

Inicialmente o aluno trabalhou em conjunto com uma aluna de mestrado utilizando Sistemas Imunológicos Artificiais (Clonalg e Rede Imune) e Algoritmos Transgenéticos para a predição da estrutura de proteínas num modelo baseado em *lattice* (3D HP). Dessa colaboração resultaram duas publicações: Almeida, Gonçalves e Delgado (2007) e Almeida et al. (2007).

Ainda nesse período o aluno passou a trabalhar em conjunto com outro aluno de mestrado no desenvolvimento de uma versão inicial de um Sistemas Imunológico Artificial e de um Algoritmo Cultural. Essas versões iniciais foram aplicadas ao problema do despacho econômico com efeito de ponto de válvula e resultaram em artigos publicados nas seguintes conferências:

- *6th International Conference on Artificial Immune Systems* (GONÇALVES et al., 2007);
- XL Simpósio Brasileiro de Pesquisa Operacional (GONÇALVES et al., 2008);
- *15th International Conference on Intelligent System Applications to Power Systems* (GONÇALVES et al., 2009).

Deste trabalho conjunto também resultou a defesa de uma dissertação de mestrado (KUK, 2009).

As principais contribuições desta tese são:

- Proposta de um Algoritmo Cultural com espaço populacional formado por um Sistema Imunológico Artificial;
 - Uma nova e eficiente formulação para a influência principal utilizada nos Algoritmos Culturais;
 - Formalização das funções de influência dos Algoritmos Culturais para Sistemas Imunológicos Artificiais;
 - Nova forma de representação do conhecimento topográfico, bem como um novo procedimento de atualização deste conhecimento;

- Um novo algoritmo imuno-cultural baseado em dominância de Pareto para problemas multiobjetivo;
- Algoritmos imuno-culturais mono-objetivo eficientes para o problema do despacho econômico de energia;
- Algoritmos imuno-culturais multiobjetivo eficientes para o problema do despacho econômico/ambiental de energia;
- Desenvolvimento de um novo procedimento de reparo para os problemas de despacho, procedimento este que considera a perda de energia.

1.3 Estrutura da Tese

Os Capítulos 2 a 4 apresentam a base teórica para o entendimento da metodologia proposta. A visão geral dos Algoritmos Evolucionários utilizados neste trabalho é mostrada no Capítulo 2. No Capítulo 3 os princípios e os algoritmos para problemas de otimização mono e multiobjetivo são brevemente descritos. Já o Capítulo 4 descreve o problema tratado nesta tese. O Sistema Imunológico Artificial e os Algoritmos Culturais propostos neste trabalho são apresentados no Capítulo 5. No Capítulo 6 encontram-se os estudos de caso considerados, as simulações realizadas com os algoritmos propostos juntamente com os resultados obtidos. Por fim, no Capítulo 7 são traçadas as conclusões e alguns apontamentos para trabalhos futuros.

2 *Algoritmos Evolucionários*

A nomenclatura associada aos algoritmos inspirados em mecanismos da evolução natural, denominados algoritmos evolucionários (DASGUPTA DIPANKAR; MICHALEWICZ, 2001; BACK, 1996), vem sofrendo várias modificações ao longo dos anos. O tema pode ser encontrado em áreas relacionadas à Computação Evolucionária (do inglês *Evolutionary Computation*) (MITCHELL; TAYLOR, 1999), Computação Flexível (do inglês *Soft Computation*) (ALIEV; ALIEV, 2001), Computação Bio-inspirada (do inglês *Bio-inspired Computing*) (DECASTRO et al., 2004), Computação Natural (do inglês *Natural Computing*) (DE CASTRO, 2006), e ainda à Inteligência Computacional (*Computational Intelligence*) (ENGELBRECHT, 2007). Neste trabalho optou-se por manter o nome algoritmos evolucionários por se entender ser esta uma forma mais específica de tratar os algoritmos discutidos neste trabalho.

Os Algoritmos Evolucionários (AEs) surgiram como soluções eficientes, genéricas e robustas para uma gama de problemas NP-Árduos (CHRISTIAN; ANDREA, 2003) e seus mecanismos de busca são inspirados em processos biológicos, em especial nos processos de evolução dos seres vivos. Os AEs são baseados no processamento de diversas soluções (população) que vão sofrendo alterações por meio de operadores específicos, realizando assim buscas eficientes no espaço de soluções. Podem ser citados como exemplos de grande utilização na área de Algoritmos Evolucionários: os Algoritmos Genéticos (HOLLAND, 1962; LACERDA; CARVALHO, 1999; GOLDBERG, 1989; WHITLEY, 2004) e a Programação Genética (KOZA, 1990; KOZA, 1992; KOZA, 1994; KOZA et al., 1999; KOZA et al., 2003), os Algoritmos de Enxames de Partículas (KENNEDY; EBERHART, 1995; EBERHART; SHI; KENNEDY, 2001; PARSOPOULOS; VRAHATIS, 2002b; PARSOPOULOS; VRAHATIS, 2002a), as Colônias de Formigas (DORIGO; MANIEZZO; COLORNI, 1991; BONABEAU; DORIGO; THERAULAZ, 1999; DORIGO; STUTZLE, 2004; MANIEZZO; CARBONARO, 1999), a Evolução Diferencial (PRICE; STORN; LAMPINEN, 2005; ONWUBOLU; BABU, 2004; KARABOGA; OKDEM, 2004), os Sistemas Imunológicos Artificiais (DE CASTRO, 2001; NUNES, 2003; TIMMIS, 2004) e os Algoritmos Culturais (REYNOLDS, 1994).

A seguir serão descritos de forma sucinta alguns destes paradigmas.

- **Algoritmos Genéticos (*Genetic Algorithm*):** Os Algoritmos Genéticos (AG) foram propostos por John Holland (HOLLAND, 1962). Os AG são algoritmos estocásticos de busca inspirados na evolução de Darwin. Os AG tratam basicamente da simulação da evolução de estruturas individuais (cromossomos - em geral codificados na forma de vetores), via processo de seleção e operadores de busca, chamados operadores genéticos (mutação e cruzamento). Assim, as soluções vão sendo modificadas através dos operadores genéticos até que uma condição de parada seja atingida e ao final o(s) melhor(es) indivíduo(s) representa(m) a solução do problema. São algoritmos robustos podendo ser aplicados a uma vasta lista de problemas práticos (FREEMAN, 1998).
- **Programação Genética (*Genetic Programming*):** O algoritmo de Programação Genética (PG) foi proposto por Koza (1990). Este é similar ao Algoritmo Genético mas as soluções, em geral, são codificadas na forma de árvores e representam programas de computador. Quando associada a linguagens de programação, a PG permite a manipulação de um programa computacional na forma de uma estrutura de dados (árvore, por exemplo), possibilitando que novos dados do mesmo tipo e recém-criados sejam imediatamente executados como programas computacionais.
- **Nuvem de Partículas (*Particle Swarm Optimization*):** O algoritmo de Nuvem ou Enxame de Partículas (NP) foi proposto originalmente por Kennedy e Eberhart (1995). NP é uma técnica de busca heurística que simula o comportamento social (movimentos) de bando de pássaros em revoada ou cardume de peixes. A técnica se baseia em modificações num conjunto de soluções (partículas) através de operadores que consideram o comportamento individual (memória) e coletivo das soluções obtidos. Embora a NP tenha sido originalmente adotada para equilíbrio de "pesos" em redes neurais, rapidamente se tornou um popular otimizador global, principalmente em problemas em que as variáveis de decisão são números reais (REYES-SIERRA; COELLO, 2006).
- **Colônia de Formigas (*Ant Colony Optimization*):** A técnica de Colônia de Formigas foi proposta por Dorigo, Maniezzo e Coloni (1991) e se inspira no comportamento das formigas em busca de alimento. As formigas inicialmente percorrem caminhos aleatórios mas vão, ao longo dos ciclos evolutivos, se organizando em torno dos melhores caminhos através da deposição do feromônio. A aplicação mais usual de Colônia de Formigas é na área de redes de computadores, em especial em problemas de roteamento (DORIGO; STUTZLE, 2003; WANG; LIU, 2006).
- **Evolução Diferencial (*Differential Evolution*):** A Evolução Diferencial (ED) foi proposta por Storn e Price (1995), e surgiu de tentativas de resolver o problema

de ajuste polinomial de Chebychev. É um algoritmo evolucionário aplicado em problemas de busca e otimização. Kenneth Price introduziu a idéia de usar diferenças de vetores para perturbar a população de soluções codificadas na forma de vetores, resultando em um método que requer poucas variáveis de controle, é de rápida convergência, fácil de usar e robusto (PRICE; STORN; LAMPINEN, 2005). Suas aplicações incluem diversas áreas conforme discutido em Oliveira (2006) e Arantes, Oliveira e Saramago (2006).

- **Sistemas Imunológicos Artificiais (*Artificial Immune System*):** Os Sistemas Imunológicos Artificiais (SIAs) foram propostos por Timmis (2001) e se inspiram no funcionamento do Sistema Imune para a solução de diversos problemas. Existem vários modelos de Sistemas Imunológicos Artificiais. Neste trabalho será utilizado um Sistema Imunológico Artificial baseado na Teoria da Seleção Clonal (Clonalg). Detalhes podem ser encontrados na Seção 2.1.
- **Algoritmos Culturais (*Artificial Cultural Algorithms*):** Os algoritmos Culturais (ACs) foram inicialmente idealizados por Reynolds (1994) e baseiam-se em teorias sociais e arqueológicas que modelam a evolução cultural dos povos (BECERRA; COELLO, 2006b). Os Algoritmos Culturais consistem de dois componentes principais (REYNOLDS, 1999b): o espaço populacional que corresponde ao conjunto de possíveis soluções podendo ser modelado utilizando qualquer técnica que faça uso de uma população de indivíduos e o espaço de crenças que consiste de um repositório de informações (conhecimentos) no qual os indivíduos podem armazenar suas experiências para que outros indivíduos possam aprendê-las indiretamente. A interação entre o espaço populacional e o espaço de crenças ocorre através de protocolos de comunicação. Neste trabalho será utilizado um Algoritmo Cultural cujo espaço populacional é formado por um SIA. Detalhes adicionais podem ser obtidos na Seção 2.2.

Assim, este capítulo pretende abordar de forma mais detalhada os dois métodos da Computação Natural que serão utilizados na proposta deste trabalho: os Sistemas Imunológicos Artificiais e os Algoritmos Culturais.

2.1 Sistemas Imunológicos Artificiais

Sistemas Imunológicos Artificiais (SIAs) são sistemas adaptativos inspirados pela imunologia teórica e pelas funções, princípios e modelos imunes, os quais são aplicados a domínios de problemas complexos (TIMMIS, 2004).

Existem outras definições para Sistemas Imunológicos Artificiais (ver, por exemplo, (TIMMIS et al., 2004)), contudo esta foi considerada a mais interessante por focar-se na aplicabilidade do método.

Em Castro e Timmis (2002b) foi proposto um *framework* ou plataforma básica para Sistemas Imunológicos Artificiais. Esta plataforma se baseia no fato de que os sistemas necessitam de, pelo menos, os seguintes elementos (TIMMIS et al., 2004):

- Uma representação para os elementos do sistema (células, antígenos, anticorpos, epitopos, paratopos, entre outros);
- Um conjunto de mecanismos para avaliar a interação entre o ambiente e os componentes e entre um componente e outro (funções ou medidas de afinidade);
- Procedimentos de adaptação que governem a dinâmica do sistema, ou seja, como o comportamento do sistema varia com o tempo (os algoritmos propriamente ditos).

A plataforma pode ser vista como um conjunto de camadas a partir das quais o sistema é construído (Figura 1).

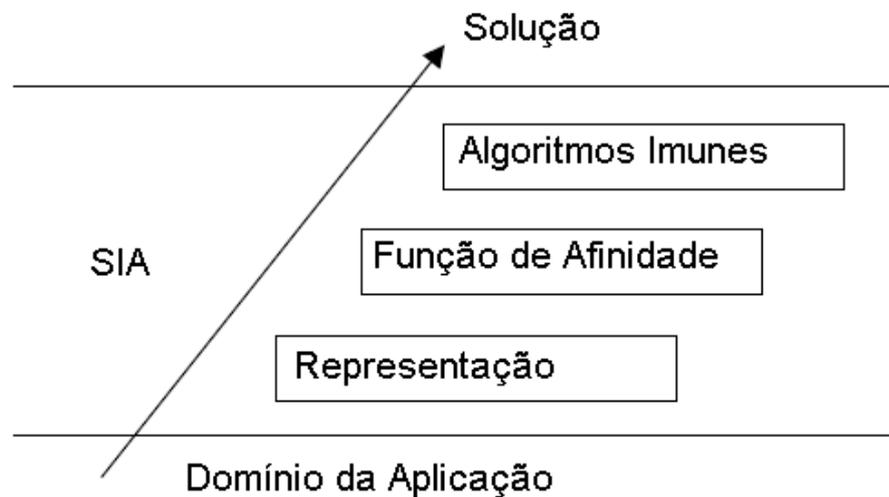


Figura 1: Camadas da plataforma para sistemas imunológicos artificiais (adaptado de (NUNES, 2003)).

2.1.1 Paradigmas

Existem inúmeros algoritmos para Sistemas Imunológicos Artificiais. Apesar de existir uma plataforma básica proposta em Castro e Timmis (2002b) e de existirem plataformas

propostas em Stepney et al. (2005), praticamente todo algoritmo de Sistemas Imunológicos Artificiais é um algoritmo diferente. Mesmo assim, eles normalmente se encaixam em um dos paradigmas que serão discutidos a seguir.

Os paradigmas nos quais os algoritmos se encaixam são:

- **Medula Óssea:** se baseiam na maneira como a medula óssea produz as Células B e T (são células responsáveis pelo reconhecimento e eliminação de antígenos) de forma a garantir uma boa diversidade das mesmas. Um exemplo de algoritmo que se enquadra dentro deste paradigma pode ser encontrado em DE CASTRO (2001);
- **Seleção Negativa:** faz a distinção entre próprio e não-próprio e visa eliminar as células do sistema imune com alta afinidade com células próprias. Este paradigma é utilizado na detecção de invasores em sistemas computacionais (HOFMEYR; FORREST, 1999; DASGUPTA; GONZALEZ, 2002; GONZALEZ; DASGUPTA, 2003; BALTHROP; FORREST; GLICKMAN, 2002a; BALTHROP; FORREST; GLICKMAN, 2002b);
- **Seleção Positiva:** também faz a distinção entre próprio e não-próprio mas visa a geração de células capazes de reconhecer (resolver) os antígenos do problema. A Seleção Positiva é usada para os mesmos casos que a seleção negativa (para detalhes de quando uma abordagem é mais eficiente do que a outra e das principais diferenças entre ambas ver Esponda, Forrest e Helman (2004)). Em Kim, Ong e Overill (2003) a seleção positiva foi utilizada como parte de um algoritmo que visa detectar fraudes;
- **Seleção Clonal:** é o paradigma mais comumente utilizado (FRESCHI; COELLO; REPETTO, 2009a). Se baseia no princípio de Seleção Clonal para selecionar e expandir células (geralmente anticorpos) através do processo de Hipermutação Somática (processo intensivo de mutação). Foram utilizados para reconhecimento de padrões (DE CASTRO, 2001), otimização (DE CASTRO, 2001; PACHECO, 2004; COELLO; CORTÉS, 2005a), detecção de invasores (KIM; BENTLEY, 2001), detecção de falhas em redes (SARAJANOVIC; BOUDEC, 2003), Predição da Estrutura de Proteínas (CUTELLO et al., 2005b; CUTELLO et al., 2004; CUTELLO; NICOSIA; PAVONE, 2004b; MICHAUD et al., 2001), entre outros;
- **Redes Imunológicas:** baseiam-se nas interações entre os inúmeros elementos do sistema imunológico, em particular na relação entre os paratopos (receptores) e os epitopos (parte sendo reconhecida). Este paradigma modela melhor a dinâmica existente dentro do Sistema Imune. Foi aplicado no reconhecimento de padrões (WATKINS; TIMMIS; BOGGESS, 2004a; WATKINS; TIMMIS, 2002), otimização

(CASTRO; TIMMIS, 2002a), agrupamentos genômicos (SOUSA et al., 2004; BEZERRA; CASTRO, 2003b; BEZERRA; CASTRO, 2003a), entre outros;

- **Teoria do Perigo:** é um paradigma inspirado nas pesquisas de Matzinger (ver, por exemplo, (MATZINGER, 1998)). Neste paradigma ao invés da distinção próprio/não-próprio existe a distinção entre agentes nocivos (perigosos) ou não. Apenas agentes perigosos disparam uma resposta imunológica. O uso mais comum deste paradigma é na mineração de dados. Aickelin e Cayzer (2002) descrevem algumas formas de se aplicar a teoria em Sistemas Imunológicos Artificiais enquanto em Secker, Freitas e Timmis (2005), Secker, Freitas e Timmis (2003) são descritos algoritmos baseados nesta teoria capaz de classificar *e-mails*;
- **Sistema Imune Inato:** geralmente modelam a cooperação entre o Sistema Imune Inato e o Sistema Imune Adaptativo. Em Twycross e Aickelin (2005) é proposto um meta-modelo para este tipo de Sistema Imune Artificial e em Kephart, Sorkin e Swimmer (1997) e Kephart et al. (1998) a metáfora do Sistema Imune Inato é usada na detecção de vírus em arquivos executáveis de computador;
- **Vacina:** visa vacinar o sistema computacional contra alguma forma de problema (antígeno). O conceito de vacinação e Seleção Clonal é utilizado em May, Mander e Timmis (2003b), May, Mander e Timmis (2003a) para melhorar testes de *software*.

Recentemente foi provado que alguns algoritmos imunológicos são capazes de convergir para ótimos globais se certas condições forem atendidas (CUTELLO et al., 2007). Tais condições são: o algoritmo deve ser elitista e as hipermutações devem permitir que qualquer solução possa ser gerada a partir de uma determinada solução. Em particular, os algoritmos baseados no princípio da seleção clonal, como o Clonalg (CASTRO; ZUBEN, 2002; CUTELLO et al., 2006), se encaixam nessa classe de algoritmos.

A subseção seguinte explica brevemente os principais conceitos da seleção clonal e descreve o principal algoritmo baseado nesse princípio, o Clonalg uma vez que este serve de base para o Sistema Imunológico Artificial proposto neste trabalho.

2.1.2 Princípio da Seleção Clonal

O princípio da seleção clonal explica as características básicas da resposta imune adaptativa ao ataque de um antígeno. Este princípio estabelece que somente a célula que é capaz de reconhecer um determinado estímulo antigênico deve proliferar-se, sendo selecionada em detrimento das outras (DE CASTRO; VON ZUBEN, 2000).

Quando ocorre o ataque de um antígeno a um organismo, as células do Sistema Imunológico (SI) respondem a esse ataque através de um conjunto (população) de anticorpos. Cada célula é capaz de reconhecer um número restrito de invasores, além de ter o poder de reconhecer antígenos próprios daquele organismo. As células reconhecedoras de antígenos próprios são eliminadas do repertório de anticorpos antes que sejam capazes de exercer uma resposta (seleção negativa). Já aquela célula que melhor reconhece o antígeno estranho é selecionada para proliferar-se. Após a proliferação, os clones sofrem um processo de mutação somática a altas taxas na tentativa de melhorar a afinidade dos clones com o antígeno invasor, chamada de hipermutação somática. A prole (clones) gerada a partir deste anticorpo diferencia-se em células de memória (células com um longo ciclo de vida) ou em plasmócitos, células que excretam anticorpos que irão atacar o invasor. A Figura 2 ilustra o princípio da seleção clonal (DE CASTRO; TIMMIS, 2002). Na presença de um antígeno estranho os anticorpos que reconhecem este antígeno proliferam-se. Na sequência os anticorpos proliferados sofrem hipermutação e diferenciam-se em células de memória e plasmócitos.

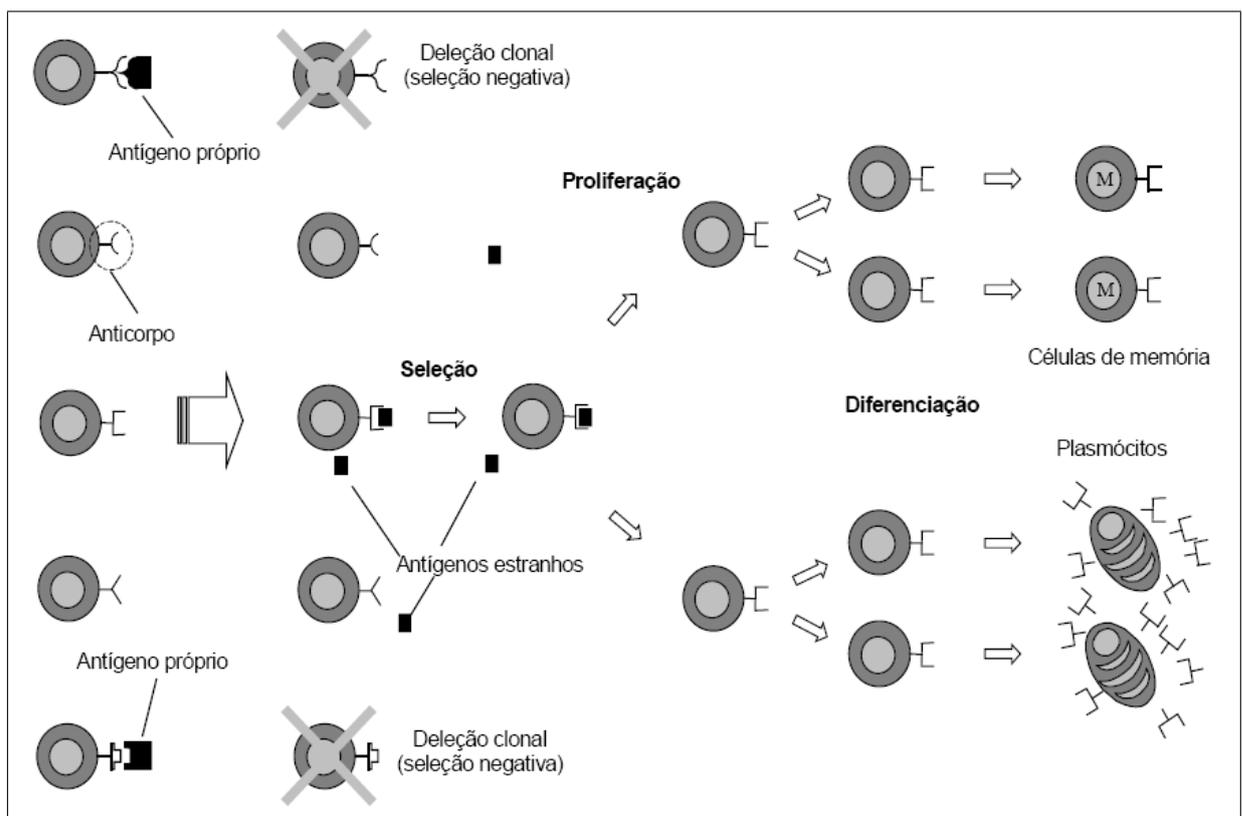


Figura 2: Princípio da seleção clonal (DE CASTRO, 2001).

Para que o sistema imunológico seja eficiente é necessário que ele seja capaz de adaptar a resposta imune a cada ataque sofrido pelo organismo e também que o número de anticorpos que irá atacar uma população de antígenos seja suficiente para eliminá-la.

No processo evolutivo do SI um antígeno é apresentado a um organismo diversas vezes. Em uma primeira exposição a quantidade de anticorpos, com alta afinidade com tal antígeno, disponíveis para a resposta imune é pequena. Em um segundo ataque o sistema imunológico já é capaz de produzir uma resposta mais rápida e precisa, pois ele guarda no organismo células capazes de gerar anticorpos (células B) com alta afinidade àquele antígeno, essas células são chamadas de células de memória. Então quanto maior é a exposição de um organismo a determinado invasor mais eficiente é a resposta imunológica gerada, o que demonstra a capacidade de aprendizagem dos SIs. A Figura 3 ilustra a memória imunológica e a aprendizagem por reforço apresentada pelo sistema imunológico biológico.

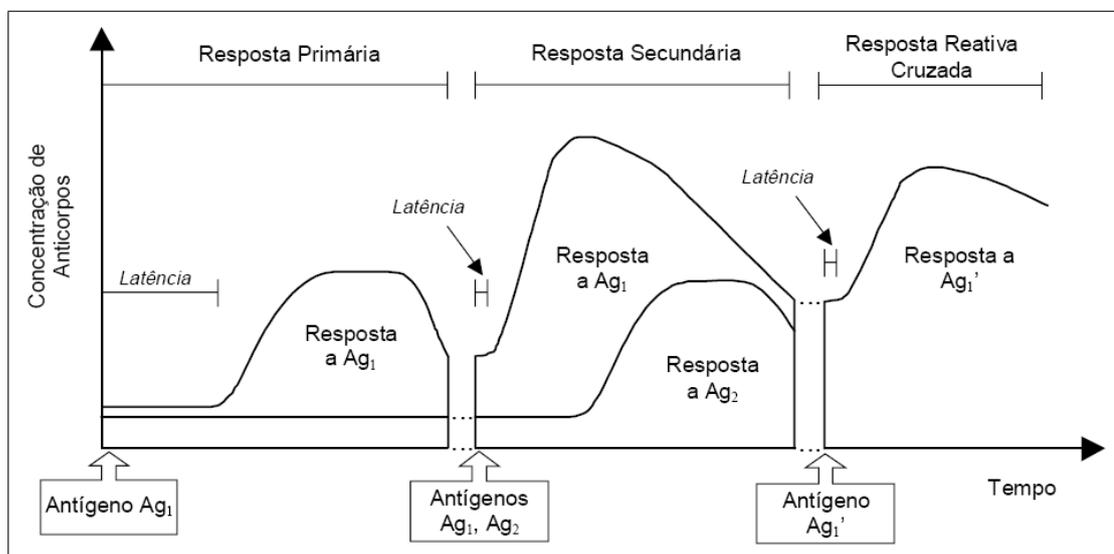


Figura 3: Resposta primária, secundária e reativa cruzada. Após a exposição primária de um antígeno Ag_1 , novas exposições do mesmo antígeno (resposta secundária) ou de um antígeno similar Ag'_1 (reação cruzada) farão com que o sistema imunológico gere uma resposta rápida e eficiente. A resposta primária de Ag_2 , um antígeno diferente de Ag_1 , é bem semelhante à resposta primária de Ag_1 (DE CASTRO, 2001).

Os anticorpos que fazem parte de uma resposta imunológica secundária possuem, no geral, maior afinidade com o antígeno do que os anticorpos presentes na resposta primária. Isso é possível graças ao fenômeno chamado maturação da afinidade. Esta maturação dos anticorpos acontece devido a alterações nas células B durante a expansão clonal (hipermutação somática). Essas alterações podem gerar células que irão resultar em anticorpos de maior afinidade com o antígeno ou então gerar células de baixa afinidade ou que reconheçam antígenos próprios. Essas células que resultarão em anticorpos de baixa afinidade são eliminadas do repertório de células imunológicas (DE CASTRO, 2001). A maturação acontece a cada re-exposição do antígeno ao organismo e faz com que a afinidade dos anticorpos cresça constantemente (Figura 4).

A quantidade de mutação que uma célula B recebe é regulada pelo processo de seleção.

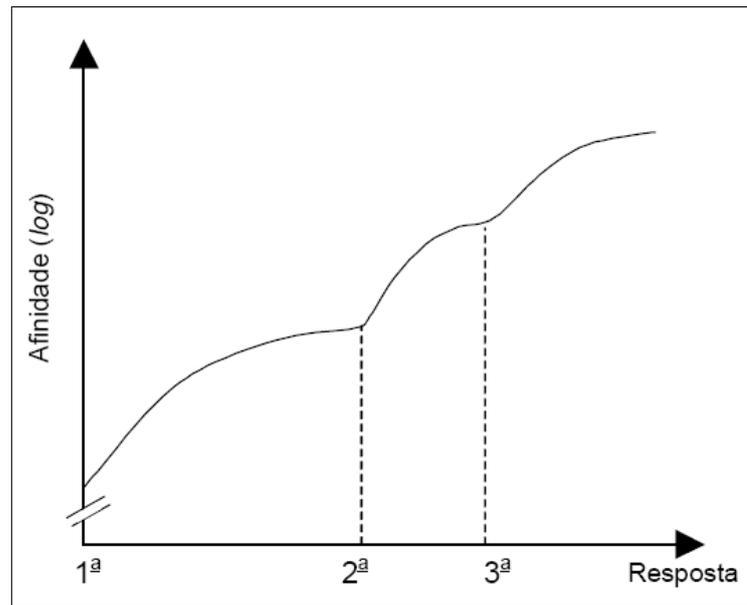


Figura 4: Maturação da afinidade ao longo das respostas imunes (DE CASTRO, 2001).

A seleção das células B de melhor afinidade diz qual célula ainda deve continuar sendo mutada ou não. As células que possuem alta afinidade com o antígeno devem ser pouco (ou nada) mutadas enquanto aquelas que possuem baixa afinidade devem continuar sofrendo mutações a altas taxas (DE CASTRO; TIMMIS, 2002).

2.1.2.1 Clonalg

O algoritmo Clonalg (*Clonal Selection Algorithm*) é o algoritmo baseado em sistemas imunológicos mais amplamente difundido. Como o nome sugere, ele foi inspirado pelo princípio da seleção clonal, em particular na geração e proliferação de clones, na presença de antígenos e na modificação dos clones através da aplicação da hipermutação somática.

O Clonalg foi inicialmente proposto para resolver problemas de aprendizado de máquina e reconhecimento de padrões e foi posteriormente adaptado para resolver problemas de otimização (DE CASTRO, 2001).

Existem algumas diferenças entre a versão do algoritmo utilizada para problemas de reconhecimento de padrões e a versão utilizada para otimização. Neste trabalho será focada apenas a versão para resolver problemas de otimização. A versão canônica do algoritmo Clonalg para otimização é dada pelo Algoritmo 1.

O algoritmo Clonalg recebe como parâmetros o tamanho da população (Tam_{POP}), a taxa de hipermutação ($Taxa_{HIP}$) e o número de clones que será gerado a partir de cada anticorpo da população ($Fator_{DUP}$). Além disso, é necessário definir a representação

Algoritmo 1 Forma básica do algoritmo Clonalg (DE CASTRO; VON ZUBEN, 2000)

```

Clonalg(TamPOP, FatorDUP, TaxaHIP)
  gerações ← 0;
  POP = Inicialização(TamPOP);
  Avalia(POP);
  Enquanto(!Condição_de_Parada) faça
    POPc ← Clonagem(POP, FatorDUP);
    POPh ← Hipermutação(POPc, TaxaHIP);
    Avalia(POPh);
    POP ← Seleção(POP, POPh);
    Substitui d piores anticorpos por anticorpos novos;
  Fim Enquanto

```

dos anticorpos (que pode ser binária, real ou inteira) e a medida de afinidade entre os anticorpos e os antígenos.

A geração da população inicial costuma ser aleatória, mas pode fazer uso da teoria das bibliotecas gênicas (ou qualquer outra metaheurística), a qual é seguida pela avaliação da população. A clonagem replica cada um dos anticorpos da população FATOR_{DUP} vezes. Esta é a principal característica que diferencia a versão do Clonalg para otimização da versão para reconhecimento de padrões (DE CASTRO, 2001), já que no segundo o número de clones gerados por cada anticorpo é proporcional à afinidade deste anticorpo com o antígeno atual. Os clones são então submetidos a um processo de hipermutação somática, o qual geralmente corresponde a uma mutação com taxa inversamente proporcional à afinidade do clone. A população de clones é avaliada e os melhores anticorpos entre a população original e os clones hipermutados são selecionados para compor a próxima população. Após isto, os *d* piores anticorpos são retirados da população e substituídos por novos anticorpos gerados aleatoriamente. Este processo se repete até que a condição de parada seja alcançada.

O algoritmo possui algumas características importantes, dentre as quais destacam-se a capacidade de auto-adaptação da mutação, introdução de diversidade na população e preservação dos melhores anticorpos.

O algoritmo básico tem sido modificado recentemente. Em Cutello, Nicosia e Pavone (2004a) são analisados diferentes tipos de operadores de hipermutação. Uma versão adaptativa do algoritmo Clonalg é proposta em Garrett (2004). O uso de um operador de *aging* é proposto em Cutello et al. (2005a). Em Watkins (2005) são propostas diferentes versões paralelas de um algoritmo Clonalg bastante modificado (o AIRS, que é utilizado para o reconhecimento de padrões). O uso da metáfora da seleção clonal para a solução de problemas de otimização multi-objetivo é investigado em Coello e Cortés (2005a).

2.1.3 Aplicações

Apesar de ser uma área relativamente nova (cerca de dez anos) já existem aplicações dos Sistemas Imunológicos Artificiais em diversas áreas (DE CASTRO, 2001). A seguir serão comentadas algumas áreas e aplicações de SIAs para elas:

- **Mineração de Dados:** é o processo de análise de conjuntos de dados que tem por objetivo a descoberta de padrões interessantes e que possam representar informações úteis. Existem diversos SIAs desenvolvidos para tratar este problema. Em Alves et al. (2004) é proposto um novo algoritmo de indução de regras de classificação em mineração de dados, chamado IFRAIS. O IFRAIS busca descobrir um conjunto de regras *fuzzy*, através de um processo evolucionário baseado em Sistema Imunológico Artificial e procedimentos usuais em mineração de dados. O trabalho reporta resultados para várias bases de dados, alcançando resultados comparáveis a populares algoritmos para mineração de dados. Um algoritmo baseado nas metáforas de seleção negativa e seleção clonal é proposto em Lee et al. (2003) para consultas adaptativas em *web* semântica. O trabalho de Alves et al. (2007) apresenta um Sistema Imunológico Artificial para descobrir um conjunto de regras para problemas de classificação hierárquica e multi-camadas. Neste sentido, uma simples regra de classificação pode determinar várias classes (camadas) em diferentes níveis hierárquicos;
- **Busca e Otimização:** são problemas que consistem na determinação de uma solução ótima dentro de um espaço de busca. Exemplos de uso de SIAs são encontrados em: (FRANÇA; VON ZUBEN; DE CASTRO, 2005), (COELLO; CORTÉS, 2005a), (WALKER; GARRETT, 2003), (CUTELLO; NICOSIA; PAVONE, 2003) e (COSTA et al., 2002). Em França, VON ZUBEN e DE CASTRO (2005) é proposta uma rede imune que estende a rede opt-aiNet para ambientes dinâmicos. Esta rede foi denominada dopt-aiNet. A dopt-aiNet possui algumas características relevantes como ajuste automático do limiar de supressão e da taxa de mutação e novos tipos de mutação e foi aplicada a problemas de otimização estáticos e dinâmicos obtendo bons resultados. Uma alternativa para a solução de problemas multiobjetivo com ou sem restrições é proposta em Coello e Cortés (2005a). Neste trabalho um algoritmo baseado no princípio da seleção clonal, fazendo uso de memória secundária e função de aptidão baseada na dominância dos anticorpos é aplicado a funções clássicas de *benchmarks*. O problema de coloração de grafos é resolvido por um algoritmo imune baseado no princípio da seleção clonal que incorpora um processo de busca local em Cutello, Nicosia e Pavone (2003);

- **Aprendizagem de Máquina:** é uma área da ciência preocupada com o desenvolvimento de algoritmos que permitem que os sistemas computacionais evoluam comportamentos a partir de dados empíricos. Nos seguintes trabalhos dessa área foram utilizados SIAs: Watkins, Timmis e Boggess (2004b) e Bersini (1999). Em Watkins, Timmis e Boggess (2004b) é proposto um algoritmo para classificação que faz uso do princípio da seleção clonal e ARB (*Antigenic Recognition Ball*). Os ARBs são utilizados para restringir a proliferação de anticorpos redundantes;
- **Robótica:** algumas aplicações de SIAs para esse problema podem ser encontradas em Ishiguro et al. (1998), Canham, Jackson e Tyrrell (2003) e Vargas et al. (2003). Ishiguro et al. (1998) propuseram um Sistema Imunológico Artificial baseado no princípio de moderacionismo para controlar a movimentação de um robô com seis pernas. A implementação de uma rede imuno-genética para a movimentação autônoma de um robô do tipo Khepera II é proposta em Vargas et al. (2003). Nesta rede o sistema imune implementa o processo de decisão enquanto o algoritmo evolutivo define a estrutura da rede. Lau e Ko (2007) introduzem um Sistema Imunológico Artificial para controle de um sistema robótico de baixo custo baseado no mecanismo de supressão dos sistemas imunes;
- **Controle de Processos:** criação de sistemas que automatizam o controle de máquinas e processos quer sejam industriais ou não. Aplicações de SIAs nessa área são feitas em: Ding e Ren (2000), Kim (2001) e Law e Wong (2004). Um controle *fuzzy* auto ajustável inspirado no mecanismo de *feedback* do sistema imune é proposto em Ding e Ren (2000) para controlar a temperatura da pele de pessoas com hipertermia. Em Kim (2001) é proposta uma rede imune para ajustar os parâmetros de um controlador PID (Proporcional-Integral-Derivativo);
- **Segurança Computacional e de Rede:** visa identificar e tratar possíveis invasores nos sistemas computacionais. Nos seguintes trabalhos dessa área foram utilizados SIAs: Anchor et al. (2002), Wang e Hirsbrunner (2002) e Paula, Castro e Geus (2004). Em Anchor et al. (2002) é proposto um algoritmo baseado no princípio da seleção positiva que identifica possíveis ataques a uma rede de computadores. Um sistema que faz uso de princípios do sistema imune inato, da teoria do perigo, de seleção negativa é proposto em Paula, Castro e Geus (2004) para identificar ataques de *buffer overflow*.;
- **Detecção de Falhas e Anomalias:** visa identificar possíveis invasores em sistemas computacionais e defeitos em qualquer tipo de sistema. O uso de sistemas imunológicos artificiais para a solução deste tipo de problema é abordado em Bradley e Tyrrell (2000), Lee et al. (2005), González e Dasgupta (2003) e Greensmith, Aickelin e Twycross (2004). Em Lee et al. (2005) a detecção de falha num contador é

resolvida pelo uso de um sistema imune com ênfase na diversidade populacional, a qual é evoluída por evolução simbiótica. A detecção de intrusos é tratada através dos princípios da teoria do perigo em Greensmith, Aickelin e Twycross (2004);

- **Bioinformática:** O algoritmo Clonalg Cutello, Nicosia e Pavone (2004a) foi aplicado ao Problema da Predição da Estrutura de Proteínas no modelo 2D HP. Em (CUTELLO et al., 2005a; CUTELLO et al., 2006) o algoritmo Clonalg, usando operador de *aging* puro, foi aplicado ao problema da predição de proteínas nos modelos 2D e 3D HP. Modelos baseados no algoritmo Clonalg trabalhando em representações livres de *lattice* foram utilizados nos trabalhos de Cutello, Narzisi e Nicosia (2006) e Anile et al. (2006). O primeiro trabalho trata o PPEP como sendo um problema de otimização multi-modal, enquanto o segundo trata o problema como sendo multi-objetivo. Os dois utilizam o CHARMM¹ (MACKERELL et al., 1998) como função de energia a ser minimizada. Em Alves, Delgado e Freitas (2008) é apresentado um algoritmo que utiliza princípios imunológicos para gerar um classificador capaz de resolver problemas de classificação hierárquica e *multilabel*. O algoritmo é aplicado na predição da função de proteínas;
- **Outros:** diferentes aplicações de sistemas imunes podem ser encontradas em: Zhou, Dasgupta D.; Zhiling e Hongmei (2006), Rahman, Yasin e Abdullah (2004), Ishak, Abidin e Rahman (2004) e Campelo et al. (2006).

2.2 Algoritmos Culturais

Os Algoritmos Culturais (ACs) são algoritmos evolucionários baseados no processo de evolução cultural da humanidade (REYNOLDS, 1994). Os ACs foram propostos por Robert Reynolds (REYNOLDS, 1999a; REYNOLDS, 2001b; REYNOLDS, 2001a; REYNOLDS, 2003) como um complemento a metáfora evolutiva utilizada na Computação Evolutiva, metáfora essa que se concentra nos aspectos genéticos da evolução e na teoria da seleção natural proposta por Darwin. Em contrapartida, os Algoritmos Culturais baseiam-se principalmente em teorias sociais e arqueológicas que tentam modelar a evolução cultural dos povos (BECERRA, 2002).

A seguir é brevemente descrita a inspiração por detrás dos Algoritmos Culturais.

¹CHARMM (*Chemistry at HARvard Macromolecular Mechanics*) é um programa para simulações de macromoléculas, incluindo minimização de energia, dinâmica molecular e simulações de Monte Carlo.

2.2.1 Inspiração Natural

As pesquisas da evolução cultural focam-se nos níveis micro e macro evolucionários. No nível microevolucionário é modelada a evolução dos aspectos biológicos da **população**. Já no nível macroevolucionário modela-se o **conhecimento** adquirido pelos indivíduos ao longo das gerações e que codificados e armazenados servem para guiar o comportamento dos indivíduos em suas populações (sociedades). É importante salientar que a transmissão de informações culturais ocorre tanto entre indivíduos de uma sociedade (mesma população) quanto entre sociedades distintas (populações diferentes) (DURHAM, 1994; RENFREW, 1994), o que pode levar a diferentes estruturas sociais.

Os Algoritmos Culturais visam explorar esse mecanismo de herança dual com a intenção de acelerar a taxa de convergência dos Algoritmos Evolutivos ou melhorar a qualidade da população gerada (respostas mais precisas).

A seção que segue visa introduzir a plataforma básica que descreve a implementação computacional dos Algoritmos Culturais.

2.2.2 Plataforma Básica de Algoritmos Culturais

Os Algoritmos Culturais consistem de dois componentes principais:

- **Espaço Populacional:** conjunto de possíveis soluções que pode ser modelado utilizando qualquer técnica que faça uso de uma população de indivíduos (Algoritmos Genéticos, Estratégias Evolutivas, Programação Evolutiva, entre outras);
- **Espaço de Crenças:** repositório de informações (conhecimento) no qual os indivíduos podem armazenar suas experiências para outros indivíduos aprendê-las indiretamente.

Assim, cada indivíduo é descrito tanto por um conjunto de características e comportamentos quanto pelos conhecimentos adquiridos por aquele indivíduo. As características e comportamentos de um indivíduo são alterados por operadores genéticos que podem ser socialmente influenciados. Da mesma forma, os conhecimentos individuais podem ser unidos e modificados para formar o espaço de crenças da população. Essas unificações e modificações também são feitas através do uso de operadores especiais. Esses operadores podem tanto ser genéricos (aplicáveis a resolução de qualquer problema) quanto específicos (aplicáveis a um problema ou classe de problemas em particular).

Os símbolos utilizados para descrever as características e o conhecimento (espaço de crenças) de uma população também podem ser modificados ao longo do processo evolucionário. Desse modo, é possível remover ou adicionar novas características e esquecer ou adquirir conhecimentos. Ou seja, tanto as características quanto os conhecimentos são evoluídos através das experiências adquiridas pela população.

A cada iteração do modelo existe um conjunto de indivíduos descritos pelas características atualmente aplicáveis à população. O desempenho (capacidade) de cada indivíduo na resolução do problema em questão é avaliado (como geralmente ocorre na maioria dos algoritmos baseados em população²). Além disso, cada indivíduo produz uma representação (conhecimento) generalizada da experiência adquirida durante aquela iteração³. Esse processo é descrito por Reynolds (1994) como Esquematização. As crenças mais gerais de um conhecimento são chamadas de crenças dominantes. O conhecimento de um indivíduo pode então ser unido ao espaço de crenças da população através de operadores específicos. Quando ocorre essa união, a aptidão do conhecimento adquirido corresponde a média dos desempenhos dos indivíduos que possuem aquele conhecimento em particular. Após essa união, ocorre um ajuste do espaço de crenças para um conjunto aceitável de conhecimentos (conhecimentos que contribuíram para a melhoria da população), ou seja, conhecimentos não aceitáveis são removidos do espaço de crenças. Se os conhecimentos removidos forem o que Reynolds chama de reforçados (ou seja, conhecimentos que a população considera não desejáveis), então não poderá mais haver indivíduos associados aqueles conhecimentos; caso contrário, indivíduos com aqueles traços poderão surgir em populações futuras. Para isto seria necessário armazenar mais uma camada de conhecimentos para armazenar os conhecimentos descartados da camada original de conhecimentos e garantir que novos indivíduos não tivessem características contidas nesta nova camada ou extrair conhecimentos dos piores indivíduos e criar um espaço de anti-crenças (ver (TANG; LI, 2008) para um exemplo simples desta última técnica e (HUANG; LIU; GU, 2008) para um exemplo de uso de múltiplas camadas de conhecimento, mas não para conhecimentos reforçados). Neste trabalho não foi utilizado o reforço de conhecimentos removidos devido a dificuldade de lidar com este tipo de conhecimento em problemas de otimização contínua, mas versões futuras dos algoritmos propostos podem vir a fazer uso desta técnica.

O estado atual do espaço de crenças pode ser utilizado para modificar o desempenho dos indivíduos da população, modificar o conjunto de características possíveis, impedir o ressurgimento de conhecimentos removidos, entre outras coisas. A população então sofre um processo de evolução e seleção da próxima população, quando o processo todo citado

²Isso é particularmente verdade para todos os Algoritmos de Computação Evolutiva.

³Neste contexto a experiência é o contato perceptual direto e característico com aquilo que se apresenta a uma fonte cognitiva de informações

no parágrafo anterior é novamente executado. É importante salientar que o processo evolutivo de um Algoritmo Cultural é inerentemente paralelo: um espaço de crenças pode ser associado a uma ou mais populações e diferentes espaços de crenças podem ser compartilhados por diferentes populações. Isso pode dar origem a nichos especializados que podem explorar de forma mais eficiente partes do espaço de busca associado ao problema em questão ou ainda criar uma hierarquia de conhecimentos, dando origem a um meta espaço de crenças (que seria um conjunto de crenças compartilhado pelas várias sociedades, Figura 5).

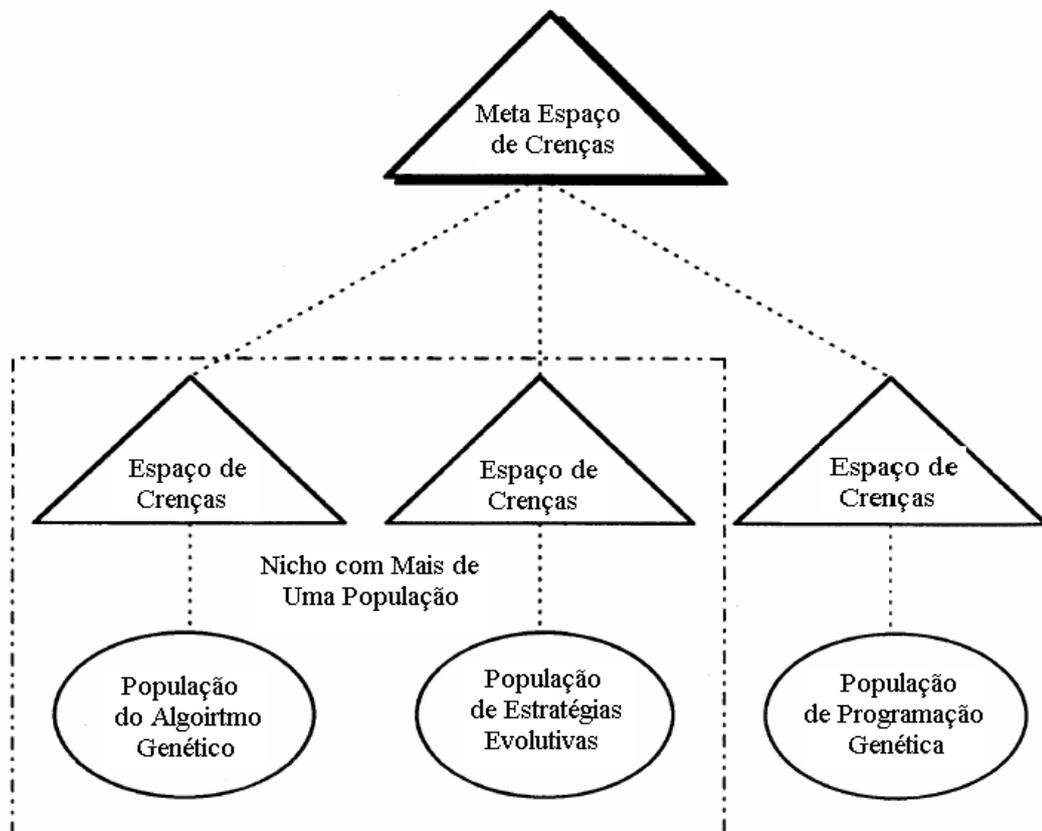


Figura 5: Execução de um algoritmo cultural com três populações, dois nichos e um meta-espaço de crenças (adaptado de Reynolds (2003)).

Tanto o espaço de crença quanto os indivíduos de uma população podem se afetar, ou seja, um influi na evolução e no desempenho obtido pelo outro. Essa interferência entre ambos ocorre através de Protocolos de Interação, ou simplesmente Protocolos. Podem existir diversos tipos de Protocolos, cada um com seus mecanismos de evolução e comunicação entre o espaço de crença e a população de indivíduos⁴. A maneira pela qual a população interage com o espaço de crenças é chamada de Função de Aceitação, enquanto

⁴Cabe ressaltar que também existe um Protocolo de Interação entre os Espaços de Crença e o Meta Espaço de Crenças.

que a maneira pela qual o espaço de crenças interage com a população é chamada de Função de Influência (REYNOLDS, 1999b).

A plataforma básica dos Algoritmos Culturais é dada pela Figura 6 (REYNOLDS, 2003):

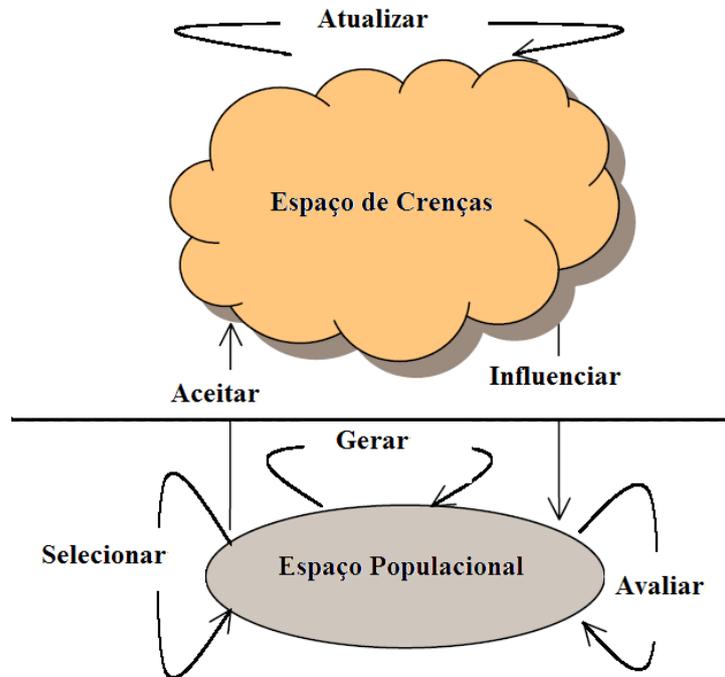


Figura 6: Funcionamento básico de um algoritmo cultural (REYNOLDS; JIN, 2002).

O pseudo-código a seguir ilustra principais passos de um algoritmo cultural.

Algoritmo 2 Plataforma básica de um algoritmo cultural

- 1: Inicializa a População
 - 2: Inicializa o Espaço de Crenças
 - 3: Repita
 - 4: Avalie a População
 - 5: Ajuste o Espaço de Crenças Através da Função de Aceitação
 - 6: Gere a Próxima População a partir da Atual Con-
siderando a Função de Influência
 - 7: Até que a Condição de Término seja Alcançada
-

Cabe salientar que o algoritmo supra descrito é similar ao algoritmo básico de Computação Evolutiva.

2.2.3 Principais Características

De acordo com Reynolds (2003), as principais características demonstradas por um Algoritmo Cultural são:

- **Mecanismo Dual de Herança:** são herdadas características tanto no nível da população quanto no nível do espaço de crenças;
- **Evolução Guiada por Conhecimento:** a população é guiada na direção que, segundo o conhecimento armazenado no espaço de crenças, seja a melhor;
- **Suporta Hierarquização:** tanto a população quanto o espaço de conhecimento podem ser organizados de forma hierárquica, permitindo a criação de nichos e, ao mesmo tempo, uma distribuição do conhecimento adquirido;
- **Conhecimento sobre o Domínio Separado dos Indivíduos:** o conhecimento adquirido é armazenado no espaço de crenças e compartilhado entre os indivíduos; assim, quando um indivíduo é eliminado da população, o conhecimento adquirido pelo mesmo permanece. Além disso, as técnicas de aquisição e manipulação de conhecimento podem ser adaptadas ao domínio da aplicação sem grandes mudanças na manipulação de indivíduos da população ou do algoritmo evolutivo sendo utilizado. Essa característica permite a criação de ontologias para o domínio sem conhecimentos prévios, ou seja, a criação *on-the-fly* (em tempo de evolução) das ontologias. Cabe salientar que no contexto dos ACs uma ontologia é um modelo de dados que representa um conjunto de conceitos dentro de um domínio e os relacionamentos entre estes (REYNOLDS, 2003). No caso dos ACs a ontologia possui o papel de permitir que os conceitos e relacionamentos extraídos possam ser utilizados de forma eficiente no aprimoramento dos indivíduos da população;
- **Suporte a Auto-Adaptação em Vários Níveis:** permite tanto a auto-adaptação da população quanto do conhecimento e da forma como o conhecimento é adquirido. Ou seja, os parâmetros de controle, a representação, os operadores (tanto genéticos quanto sociais), a avaliação dos indivíduos, e o protocolo de intercomunicação podem ser alterados a qualquer momento da evolução. Isso é particularmente útil para técnicas de metaevolução (evolução do processo evolutivo). A metaevolução de parâmetros pode ocorrer de acordo com a taxa de sucesso de cada componente do algoritmo, a dos operadores pode ocorrer através do uso de informações armazenadas no espaço de conhecimentos, o protocolo de comunicação pode ser modificado porque está permitindo a aceitação de muitos indivíduos de baixa qualidade ou poucos indivíduos de boa qualidade e assim por diante;
- **Diferentes Taxas de Evolução:** a evolução das populações e do conhecimento não precisa ocorrer na mesma taxa. Segundo Reynolds e Zandoni (1992), o conhecimento é evoluído mais rapidamente que a população.

O desenvolvimento de um Algoritmo Cultural pode ser dividido em três partes distintas: desenvolvimento do componente de conhecimento, desenvolvimento do componente

populacional e desenvolvimento do protocolo de intercomunicação. A seguir serão brevemente descritos esses componentes e por fim serão comentadas algumas aplicações de Algoritmos Culturais.

2.2.4 Microevolução × Macroevolução

Os Algoritmos Culturais implementam um mecanismo de herança dual. Esse mecanismo permite que os ACs explorem tanto a microevolução quanto a macroevolução. A microevolução é aquela que ocorre no nível populacional, ou seja, a evolução dos indivíduos e seus traços de comportamento. A macroevolução corresponde a evolução da cultura em si, ou seja, a evolução do espaço de crenças. Nos ACs a evolução da população ocorre mais rapidamente do que populações evoluídas sem o mecanismo de macroevolução.

Um dos itens de investigação desta tese é inferir se a afirmação de que populações com conhecimento evoluem mais rapidamente do que populações sem conhecimento é verdadeira para populações de anticorpos utilizando conceitos imunológicos. Ou seja, deseja-se testar se o uso de macroevolução é benéfica aos sistemas imunológicos artificiais, mais especificamente a SIAs baseados no princípio da seleção clonal. Neste trabalho a evolução será considerada mais rápida se com os mesmos recursos computacionais o sistema que faz uso de algoritmos culturais for capaz de encontrar melhores soluções. Detalhes podem ser encontrados nos Capítulos 5 e 6.

2.2.4.1 Espaço Populacional

No Espaço Populacional são representadas as características e os comportamentos dos indivíduos. Essa representação pode ser feita através de qualquer técnica que faça uso de uma população de indivíduos. Neste trabalho será investigado o uso de SIAs no espaço populacional do AC proposto. Conforme já mencionado no Capítulo 1, este representa o primeiro trabalho que utiliza esta técnica no espaço populacional de um AC.

2.2.4.2 Espaço de Crenças e Meta-Espaço de Crenças

O Espaço de Crenças em um Algoritmo Cultural é um repositório de símbolos que representam o conhecimento adquirido pelo Espaço Populacional ao longo do processo evolutivo. Desse modo, o Espaço de Crenças permite que indivíduos sejam removidos da população sem que o conhecimento por eles adquiridos seja perdido. O Espaço de Crenças foi criado para guiar os indivíduos da população na direção das melhores regiões do espaço de busca.

O Meta-espaço de Crenças tem a mesma finalidade do Espaço de Crenças, contudo ele agrega conhecimentos a respeito de outros Espaços de Crenças (e, por conseguinte, de diferentes Espaços Populacionais).

No início das pesquisas na área de Algoritmos Culturais apenas uma fonte de conhecimento era utilizada (REYNOLDS; PENG, 2004). Com o amadurecimento das pesquisas, não somente novas formas de representação foram criadas como percebeu-se que era possível combinar diferentes tipos de conhecimento no mesmo Algoritmo Cultural. Na seção seguinte serão descritos os principais tipos de conhecimentos utilizados nos Algoritmos Culturais mais recentes.

2.2.4.3 Tipos de Conhecimento

Existem cinco principais formas de representar o conhecimento no espaço de crenças. Essas formas são: conhecimento situacional, conhecimento normativo, conhecimento do domínio, conhecimento topográfico e conhecimento histórico. Reynolds sugere que qualquer conhecimento cultural pode ser expresso como alguma combinação desses cinco tipos de conhecimento (REYNOLDS; PENG; BREWSTER, 2003).

2.2.4.4 Conhecimento Situacional

O conhecimento situacional representa os melhores indivíduos encontrados até determinado momento da evolução. Segundo Jacoban, Reynolds e Brewster (2003b) ele contém um conjunto de indivíduos da população que servem como exemplo para o resto da população⁵. A quantidade de exemplos pode variar de implementação para implementação, mas costuma ser pequena.

A estrutura utilizada para representar esse tipo de conhecimento é mostrada na Figura 7. Cada indivíduo utilizado como exemplo (E_1, E_2, \dots, E_n) é armazenado junto com às suas variáveis (X_1, X_2, \dots, X_n) e à sua aptidão ($f(x)$).

O conhecimento situacional é atualizado sempre que é encontrado um indivíduo cuja aptidão supere a aptidão do pior indivíduo armazenado.

⁵Este conhecimento é inspirado por uma linha da psicologia que acredita que a personalidade é mais influenciada por fatores externos e situacionais do que características e motivações pessoais (KRAHE, 2003).

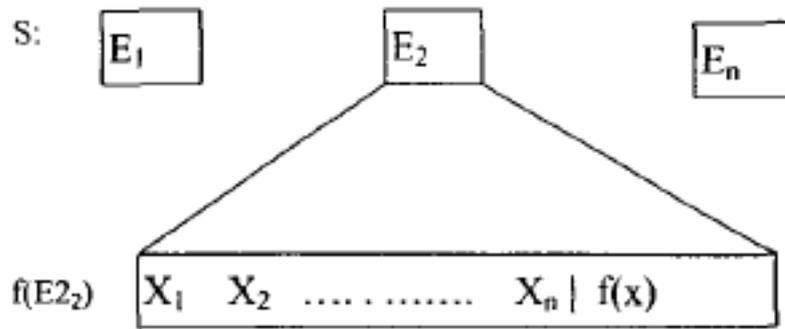


Figura 7: Representação de conhecimento situacional (IACOBAN; REYNOLDS; BREWSTER, 2003b).

2.2.4.5 Conhecimento Normativo

O conhecimento normativo representa um conjunto de intervalos que caracterizam os valores assumidos pelas variáveis que compõem as melhores soluções ⁶. Esses intervalos servem para guiar os ajustes (mutações) que ocorrem nos indivíduos.

A estrutura do conhecimento normativo utilizada nos trabalhos de Reynolds e colaboradores (REYNOLDS; PENG; ALI, 2008) é dada na Figura 8. Nessa estrutura são armazenados os valores mínimos e máximos das n -características que formam o indivíduo ($V_1 \cdots V_n$). Juntamente com os valores mínimos (l_i) e máximos (u_i) são armazenadas as aptidões dos indivíduos que deram origem a cada um desses extremos, L_i e U_i , respectivamente.

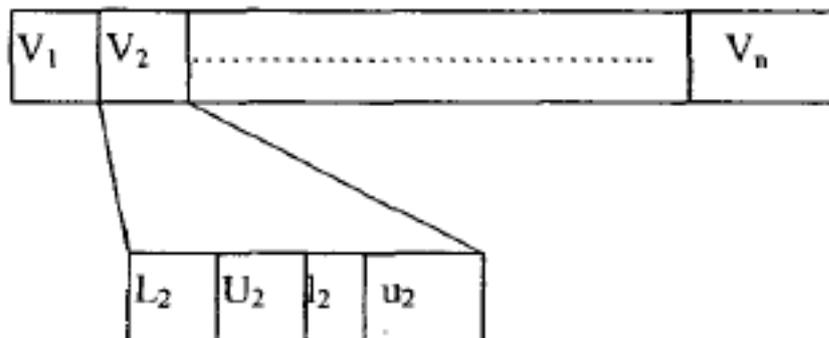


Figura 8: Representação de conhecimento normativo (IACOBAN; REYNOLDS; BREWSTER, 2003b).

Já a estrutura utilizada por Coello e seus colaboradores aparece na Figura 9 (BECERRA; COELLO, 2006a). Nesta estrutura são adicionadas as variáveis dm_i que representam a proporção do espaço de busca explorada pelo intervalo associado para cada um dos componentes da solução. Esta variável é atualizada de acordo com o valor máximo

⁶Neste contexto uma norma é um comportamento esperado dentro de um grupo social (YOUNG, 2008).

da diferença calculada durante a influência de qualquer um dos conhecimentos no i -ésimo componente de uma solução. Cabe salientar que o espaço populacional adotado por estes autores é sempre Evolução Diferencial (ED), fazendo sentido guiar os conhecimentos por diferenças.

No presente trabalho é adotada a representação de Reynolds visto que o espaço populacional adotado não é ED.

l_1	u_1	l_2	u_2	\dots	l_n	u_n				
L_1	U_1	L_2	U_2	\dots	L_n	U_n				
<table border="1" style="margin: 0 auto;"> <tr> <td>dm_1</td> <td>dm_2</td> <td>\dots</td> <td>dm_n</td> </tr> </table>							dm_1	dm_2	\dots	dm_n
dm_1	dm_2	\dots	dm_n							

Figura 9: Representação alternativa de conhecimento normativo (BECERRA; COELLO, 2005a).

O ajuste do conhecimento normativo pode alongar ou estreitar o intervalo das características. Ocorre um alongamento do intervalo sempre que um indivíduo passa pela função de aceitação e se encontra fora do intervalo atual. Nesse ponto o intervalo é alongado para incluir aquele indivíduo. Ocorre um estreitamento do intervalo da i -ésima característica sempre que o indivíduo com o menor (ou maior) valor para essa característica dentre os indivíduos aceitos tiver uma melhor aptidão que a aptidão correspondente ao extremo sendo ajustado.

A Equação 2.1 mostra o processo de decisão expansão/contração do intervalo do conhecimento normativo:

$$l_i = \begin{cases} x_{i,\min_i} & \text{se } x_{i,\min_i} < l_i \vee f(x_{\min_i}) < L_i \\ l_i & \text{caso contrário.} \end{cases} \quad (2.1)$$

$$u_i = \begin{cases} x_{i,\max_i} & \text{se } x_{i,\max_i} > u_i \vee f(x_{\max_i}) < U_i \\ u_i & \text{caso contrário.} \end{cases}$$

2.2.4.6 Conhecimento do Domínio

O conhecimento de domínio, como o próprio nome pressupõe, é específico de cada aplicação. Ele representa conhecimento sobre o domínio do problema para guiar a busca. Esse é o tipo de conhecimento menos utilizado pois é o mais difícil de ser extraído e representado.

2.2.4.7 Conhecimento Topográfico

O conhecimento topográfico foi proposto com o intuito de extrair padrões de comportamento do espaço de busca. Esse tipo de conhecimento pode espalhar indivíduos sobre todo o espaço de busca. O conhecimento topográfico identifica regiões promissoras dentro do espaço de busca e faz com que novos indivíduos as explorem. Ele é o conhecimento que busca explorar diferentes regiões do espaço de busca.

Reynolds e seus colaboradores representam o conhecimento topográfico como células num *grid* multidimensional (IACOBAN; REYNOLDS; BREWSTER, 2003b; REYNOLDS; PENG; BREWSTER, 2003; REYNOLDS; PENG, 2006a). Cada célula corresponde a uma região do espaço de busca e é caracterizada pela aptidão do seu melhor indivíduo e pelos extremos assumidos por cada característica nessa região (apresentado na Figura 10). O *grid* multidimensional é inicializado como tendo um número fixo de partições em todas as dimensões. Para cada partição existe uma árvore k-d associada, onde cada elemento contém os valores mínimos ($l_1 \cdots l_n$) e máximos ($u_1 \cdots u_n$) para cada variável naquela região, o melhor indivíduo (exemplar) daquela região e um ponteiro para possíveis sub-regiões.

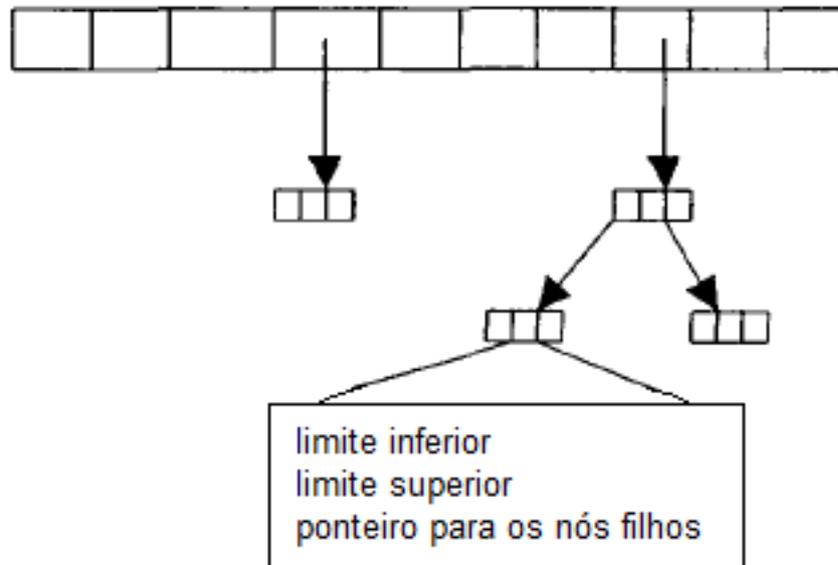


Figura 10: Representação do conhecimento topográfico (IACOBAN; REYNOLDS; BREWSTER, 2003b).

Já Coello e seus colaboradores representam o conhecimento topográfico por uma árvore k-d. Segundo Coello e Becerra (2002a) esta representação é mais eficiente do ponto de vista da memória utilizada para armazená-la, visto que este método utiliza apenas uma árvore k-d enquanto o método de Reynolds utiliza uma árvore k-d por célula do *grid* multidimensional. A Figura 11 apresenta um exemplo de espaço de busca bi-dimensional particionado em oito regiões (de H a O) através de sete divisões (de A a G) e a sua representação em árvore k-d (com k e d iguais a 2).

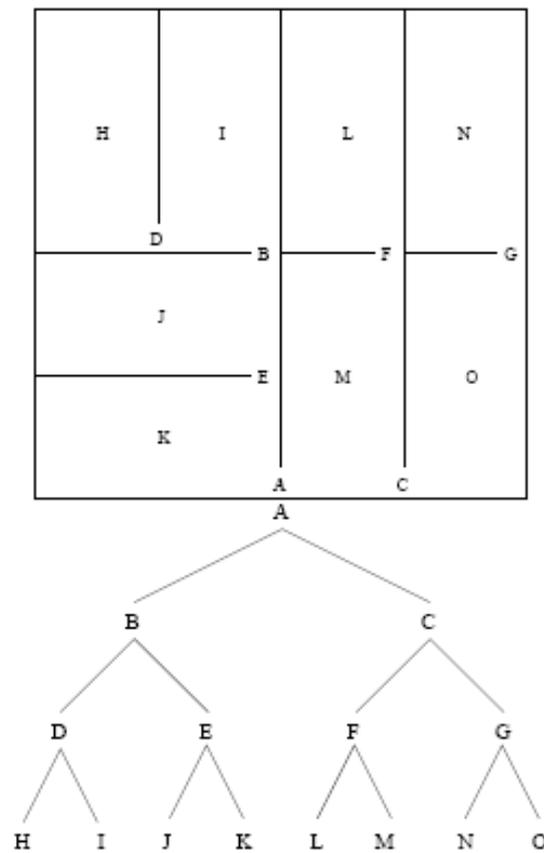


Figura 11: Representação alternativa do conhecimento topográfico (BECERRA; COELLO, 2005a).

Em ambas implementações o ajuste do conhecimento topográfico ocorre quando um novo indivíduo melhor que o melhor indivíduo de uma célula é encontrado. Quando isso ocorre a célula é dividida em k células menores e a população é percorrida para encontrar o melhor indivíduo pertencente a cada região. Pode ocorrer de nenhum indivíduo da população pertencer àquela região, neste caso um conjunto de indivíduos aleatórios é gerado e o melhor deles é escolhido como exemplar da região.

Neste trabalho é proposta uma nova representação para o conhecimento topográfico e uma nova forma de atualização deste conhecimento (ver Capítulo 5, Seção 5.2).

2.2.4.8 Conhecimento Histórico

O conhecimento histórico monitora o processo de busca e guarda importantes eventos na busca. Esse conhecimento foi motivado pela necessidade de desenvolver aprendizado em ambientes dinâmicos. Indivíduos guiados pelo conhecimento histórico podem consultar aqueles eventos armazenados para guiar suas decisões quanto a qual direção seguir.

A estrutura utilizada para representar o conhecimento histórico é demonstrada na Figura 12. e_i representa o melhor indivíduo encontrado antes da i -ésima alteração do ambiente. ds_i é a distância média das mudanças para a característica i e dr_i é a direção média se existem mudanças para a característica i .

e_1	...	e_i	...	e_w
ds_1	ds_2	...	ds_n	
dr_1	dr_2	...	dr_n	

Figura 12: Representação de conhecimento histórico (BECERRA; COELLO, 2005a).

A atualização do conhecimento histórico é dada pela Equação 2.2 (BECERRA; COELLO, 2005a):

$$\begin{aligned}
 ds_i &= (\sum_{k=1}^{w-1} Abs(e_{i,k+1} - e_{i,k})) / (w - 1) \\
 dr_i &= (\sum_{k=1}^{w-1} Sinal(e_{i,k+1} - e_{i,k}))
 \end{aligned}
 \tag{2.2}$$

onde ds_i é a direção média do i -ésimo componente, w é o número de eventos armazenado no conhecimento histórico, $e_{i,k}$ é o i -ésimo componente do k -ésimo exemplar armazenado no conhecimento histórico.

Este trabalho utiliza esta mesma representação e atualização para o conhecimento histórico.

Neste trabalho serão utilizados todos os tipos de conhecimento exceto o de domínio.

2.2.5 *Enxames Culturais*

Com o intuito de pesquisar a influência do Espaço de Crenças no Espaço Populacional foi descoberto o que Reynolds chamou de Enxames Culturais (*Cultural Swarms*) (REYNOLDS, 2001b; REYNOLDS; PENG; BREWSTER, 2003; IACOBAN; REYNOLDS; BREWSTER, 2003a). Enxames Culturais são enxames de indivíduos no Espaço Populacional guiados por grupos de conhecimentos (Enxames de Conhecimento - *Knowledge Swarms*). Conforme as fontes de conhecimento integram-se durante a resolução de um problema, emergem do sistema cultural fases de solução do problema. Para o problema de otimização de funções em ambientes dinâmicos, foi constatada a emergência de três fases: *coarse-grained*, *fine-grained* e *backtracking*.

Na fase *coarse-grained* o sistema identifica regiões no espaço de busca a serem exploradas. Na *fine-grained* essas regiões identificadas na fase anterior são exploradas de forma mais refinada. O *backtracking* ocorre quando há uma estagnação no processo de busca.

Em cada fase há uma dominância de um subconjunto de fontes de conhecimento que são melhores sucedidas em gerar novas soluções naquela fase. Na fase *coarse-grained* o conhecimento topográfico domina, produzindo as melhores soluções em 50% do tempo e o conhecimento situacional fica em segundo produzindo 25%. Na fase *fine-grained* o conhecimento situacional é predominante, seguido de longe pelos conhecimentos normativo e de domínio. No *backtracking* todos os tipos de conhecimento são bem sucedidos (IACOBAN; REYNOLDS; BREWSTER, 2003b).

Além disso, também foi observado que os ACs são capazes de gerar um comportamento semelhante a algumas classes de algoritmos (REYNOLDS; PENG; BREWSTER, 2003; REYNOLDS; PENG, 2004).

2.2.6 **Protocolos de Intercomunicação**

Os Protocolos de Intercomunicação ditam as regras sobre os indivíduos que podem contribuir com suas experiências ao Espaço de Crenças (Função de Aceitação) e a forma como o Espaço de Crenças pode influenciar novos indivíduos (Função de Influência).

O Protocolo de Intercomunicação utilizado nos primeiros trabalhos foi o de Votar-Herdar-Promover (REYNOLDS, 1994; REYNOLDS, 1999a; REYNOLDS, 2003), onde a votação corresponde ao processo de associar o desempenho de um indivíduo a um conhecimento e ao espaço de crenças (Função de Aceitação), permitindo que o conhecimento herde o desempenho dos indivíduos que possuem aquele conhecimento, promovendo (clasificando como melhores) os indivíduos que se ajustam a um conhecimento contido no espaço de crenças (Função de Influência). Esse Protocolo é representado graficamente na Figura 13. Trabalhos mais recentes se baseiam nesse protocolo mas fazem alterações de modo a melhor atender as necessidades de cada problema.

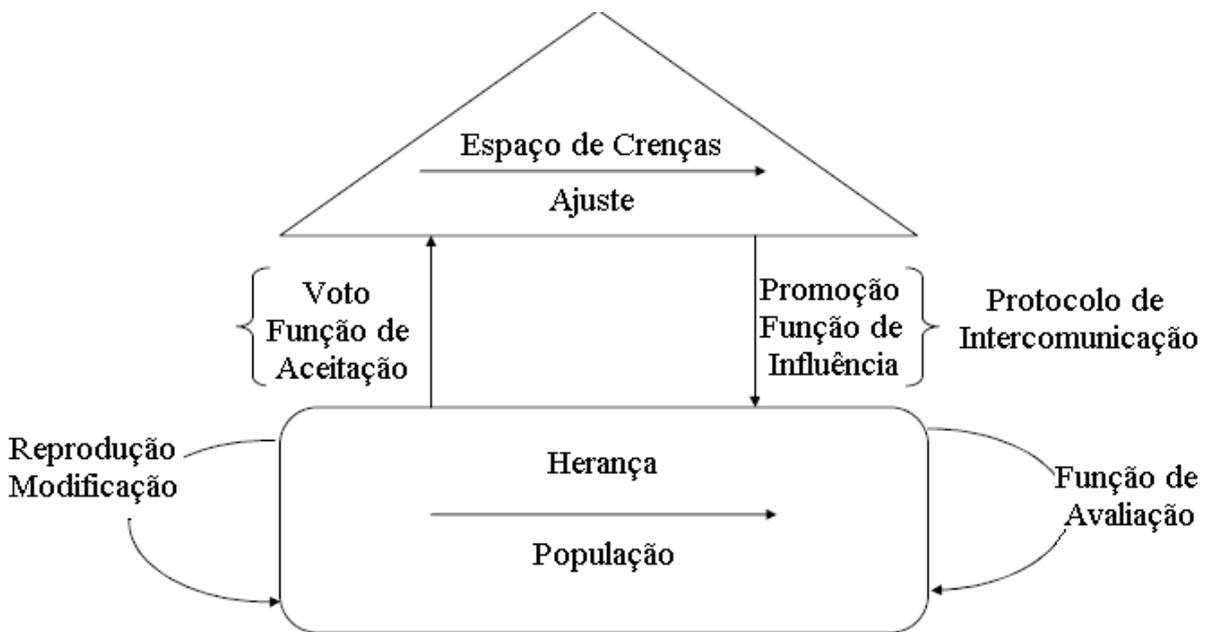


Figura 13: Protocolo Votar-Herdar-Promover (adaptado de Reynolds (2003)).

2.2.6.1 Funções de Aceitação

As Funções de Aceitação selecionam indivíduos que irão ter impacto na formação do Espaço de Crenças atual. Existem diferentes classes de Funções de Aceitação:

1. **Estática:** é utilizado um *ranking* absoluto (p% da população é selecionado) ou um *ranking* relativo (onde os indivíduos com aptidão acima da média são selecionados);
2. **Dinâmica:** o percentual de indivíduos selecionados varia ao longo da evolução. Inicialmente o processo é menos seletivo, se tornando mais restritivo ao longo da evolução;

3. **Fuzzy:** para melhor ajustar a função de aceitação dinâmica foram propostas duas funções de aceitação fuzzy. A primeira (REYNOLDS; CHUNG;, 1997) ajusta o número de indivíduo a serem selecionados de acordo com a convergência da população, quanto melhor a convergência maior o número de indivíduos aceitos. A segunda (REYNOLDS; ZHU, 2001) faz uso de uma Matriz de Similaridade *Fuzzy* e uma Matriz de α -Cortes para aceitar apenas os indivíduos que são equivalentes ao melhor indivíduo atual.

Nos algoritmos propostos neste trabalho utiliza-se uma função de aceitação dinâmica.

2.2.6.2 Funções de Influência

As Funções de Influência modelam como o conhecimento representado no Espaço de Crenças interfere nos operadores do Espaço Populacional. Geralmente, para cada tipo de conhecimento utilizado existe uma Função de Influência correspondente. A Função de Influência pode ser vista como um mecanismo de auto-adaptação do processo evolutivo, já que ela adapta os operadores conforme o conhecimento adquirido.

A Função de Influência, comumente, determina a direção e o tamanho das modificações impostas aos novos indivíduos.

Quando há várias fonte de conhecimento disponíveis, cada uma associada à sua respectiva Função de Influência, a probabilidade de uma determinada Função de Influência ser aplicada é dada pela Função de Influência Principal.

A primeira Função de Influência Principal baseava-se na taxa de sucesso da influência de cada tipo de conhecimento, ou seja, na quantidade de vezes que a aplicação de um determinado tipo de conhecimento resultava em uma melhoria do indivíduo original (SALEEM, 2001; BECERRA; COELLO, 2005b; BECERRA; COELLO, 2006a). A Equação 2.3 apresenta este tipo de Função de Influência Principal.

$$probKS_k = \beta + (1 - NKS * \beta) * succRateKS_k / \sum_{i=1}^{NKS} succRateKS_i \quad (2.3)$$

onde $probKS_k$ é a probabilidade da influência aplicada estar associada ao k-ésimo conhecimento, β é uma probabilidade mínima de cada conhecimento ser aplicado, NKS é o número de conhecimentos utilizados e $succRateKS_k$ é a taxa de sucesso da influência do k-ésimo conhecimento.

Este tipo de função de influência favorece conhecimentos que são capazes de melhorar os indivíduos da população sem levar em consideração a qualidade da melhoria resultante em cada indivíduo.

Uma proposta mais recente que leva em consideração a qualidade da melhoria resultante em cada indivíduo é o uso de um Função de Influência Principal baseada no Teorema do Valor Marginal. A idéia deste tipo de Função de Influência Principal é que conhecimento que geram indivíduos bons (acima da média) devem ser utilizados mais vezes enquanto o uso de conhecimentos que geram indivíduos ruins (abaixo da média) deve ser minimizado (REYNOLDS; PENG, 2006b; REYNOLDS; ALI; ALOMARI, 2006; REYNOLDS; PENG; ALI, 2008). A Equação 2.4 descreve este tipo de Influência Principal.

$$probKS_k = averageFitnessKS_k / \sum_{i=1}^{NKS} averageFitnessKS_i \quad (2.4)$$

onde $probKS_k$ é novamente a probabilidade de se aplicar a influência do k-ésimo conhecimento e $averageFitnessKS_k$ é o *fitness* médio dos indivíduos gerados através da influência do k-ésimo tipo de conhecimento.

Esta Função de Influência Principal pode ser vista como um jogo do tipo predador/presa onde as fontes de conhecimento são os predadores e os ótimos são as presas (REYNOLDS; ALI; ALOMARI, 2006). Então o objetivo dos predadores (fontes de conhecimento) é encontrar as regiões onde existem mais e melhores presas (ótimos).

Recentemente foi proposta uma Função de Influência Principal baseada em Tecidos Sociais (REYNOLDS; ALI, 2008a; REYNOLDS; ALI, 2008b; REYNOLDS; ALI, 2008c; ALI, 2008; ALI; REYNOLDS, 2009). A função se baseia em redes de relacionamento entre os indivíduos da população. Através destas redes eles podem comunicar os conhecimentos que eles estão utilizando e influenciar a escolha de outros indivíduos, ou seja, a escolha do tipo de conhecimento passar a ser uma escolha social.

Nesta tese é proposta uma nova Função de Influência Principal que agrega as vantagens das Funções de Influência Principal baseadas em sucessos e no Teorema do Valor Marginal.

2.2.7 Aplicações de Algoritmos Culturais

Nesta seção serão brevemente descritos alguns trabalhos relevantes da área de Algoritmos Culturais.

2.2.7.1 Suporte a Re-engenharia de Sistemas Especialistas

Em Sternberg e Reynolds (1997) foi utilizado um Algoritmo Cultural para apoiar modificações num Sistema Especialista para a detecção de fraudes em seguros de automóveis.

Foi utilizada uma função de aceitação dinâmica e foram usados os conhecimentos situacional e normativo. O Espaço Populacional foi a Programação Evolutiva.

O Algoritmo Cultural foi desenvolvido para identificar fraudes, não-fraudes, falsos positivos e falsos negativos. A partir dos resultados obtidos e da análise do conhecimento normativo representado no Espaço de Crenças foi possível gerar novas regras capazes de aperfeiçoar o funcionamento do Sistema Especialista. Na verdade, a maioria das regras correspondem diretamente ao conhecimento armazenado no Espaço de Crenças.

Um resultado importante desse trabalho foi avaliar a capacidade dos Algoritmos Culturais em uma aplicação comercial. Outro resultado importante foi a capacidade do Algoritmo Cultural proposto de adaptar-se rapidamente a mudanças do ambiente.

2.2.7.2 Otimização Usando Algoritmos Culturais Fuzzy

Reynolds e Chung; (1997), Reynolds e Zhu (2001) utilizaram Algoritmos Culturais *Fuzzy* para a otimização de funções de parâmetros reais.

Em Reynolds e Chung; (1997) foi utilizada uma função de aceitação *fuzzy* num Algoritmo Cultural com Espaço Populacional de Programação Evolutiva. Um sistema de inferência *fuzzy* foi desenvolvido para determinar o número de indivíduos aceitos a cada geração. Esse sistema possuía como variáveis de entrada o número da geração atual e a taxa de sucesso dos indivíduos. As funções de pertinência utilizadas eram lineares.

A intuição básica utilizada para desenvolver as regras de inferência *fuzzy* é que se estiver no princípio do processo evolutivo e a taxa de sucesso for baixa, então serão aceitos um número mediano de indivíduos da população (aproximadamente 30%); se o processo evolutivo estiver próximo do fim e a taxa de sucesso for baixa, um número pequeno de indivíduo é aceito (aproximadamente 20%).

Os resultados obtidos nesse trabalho mostram que o Algoritmo Cultural com função de aceitação *fuzzy* foi mais rápido em todas as instâncias testadas do que o Algoritmo Cultural *crisp*.

Já em Reynolds e Zhu (2001), fez-se uso da teoria *fuzzy* na função de aceitação, na representação e ajuste do Espaço de Crenças e na função de influência.

Na função de aceitação uma matriz de similaridade *fuzzy* e uma matriz de α -Cortes são utilizadas para classificar os indivíduos em duas subclasses: elite e não-elite. Apenas os indivíduos que fazem parte da elite são aceitos, isto é, apenas aqueles equivalentes ao ótimo atual.

São implementados apenas os conhecimentos normativo e situacional. Apenas o conhecimento normativo é adaptado a teoria dos conjuntos *fuzzy*. Para isso os extremos dos intervalos das características são transformados em números *fuzzy*.

A função de influência faz uso de um coeficiente de ajuste *fuzzy* para calcular o tamanho e a direção do passo na mutação. A idéia básica disso é perturbar pouco em direção aleatória quando o valor da característica do pai está dentro do intervalo aceitável, caso contrário a perturbação acontece na direção do intervalo aceitável.

O trabalho conclui que a maior influência de sistemas *fuzzy* em Algoritmos Culturais ocorre na função de aceitação. Além disso, o Algoritmo Cultural totalmente *fuzzy* obteve desempenho superior a qualquer outra versão testada.

O trabalho apresentado em Gonçalves et al. (2007) representa a base para o algoritmo cultural proposto nesta tese. Entretanto, a versão publicada em Gonçalves et al. (2007) é preliminar e não apresenta a possibilidade do uso de caos e de otimização multiobjetivo. O sistema é aplicado ao problema do despacho econômico de energia elétrica, na versão mono-objetivo e sem perdas. Os resultados obtidos motivaram a continuidade da pesquisa que resultou na versão melhorada a qual será detalhada no Capítulo 5.

Em Gonçalves et al. (2008) e Kuk (2009) é apresentado um algoritmo cultural *fuzzy* para solucionar um problema de otimização contínua com restrições. O sistema também é aplicado ao problema do despacho econômico de energia elétrica. O mecanismo de inferência *fuzzy* é utilizado na função de influência principal a qual define o possível percentual de aplicação de cada um dos conhecimentos. A versão apresentada em Gonçalves et al. (2008) também está diretamente relacionado à proposta desta tese. Entretanto, nesta tese foi excluído o mecanismo de inferência *fuzzy*, o qual, apesar de trazer alguns benefícios, não justificou a complexidade introduzida no modelo com o uso da inferência *fuzzy*.

2.2.7.3 Otimização com Restrições usando Algoritmos Culturais

Em Becerra e Coello (2004), Becerra e Coello (2005a), Becerra e Coello (2006b) é proposto um algoritmo cultural cujo espaço populacional é representado por evolução diferencial.

Nestes trabalhos foram utilizadas funções de aceitação dinâmicas. Foram utilizados quatro fontes de conhecimento: situacional, normativo, topográfico e histórico. Para cada uma dessas fontes foi criada uma função de influência.

A função de influência associada ao conhecimento situacional substitui um dos indivíduos escolhidos aleatoriamente pelo líder (melhor indivíduo encontrado até o momento). Isso tem como efeito a aproximação do indivíduo gerado ao melhor ponto encontrado.

A função de influência do conhecimento normativo faz com que os indivíduos gerados tenham suas características dentro do intervalo de aceitação representado.

O conhecimento topográfico possui uma função de influência que tenta mover os filhos para qualquer uma das melhores regiões do espaço de busca.

A função de influência do conhecimento histórico utiliza o melhor indivíduo armazenado no conhecimento histórico como líder, semelhante ao que ocorre no conhecimento situacional, e realiza uma mutação na direção média e no passo médio (também armazenados no conhecimento histórico). É importante ressaltar que os eventos armazenados correspondem a localização de ótimos locais.

Além dessas, a função de influência principal escolhe a fonte de conhecimento a ser aplicada no operador de variação da evolução diferencial, ou seja, qual é a influência que será aplicada. A princípio todas as fontes possuem a mesma possibilidade de escolha, mas ao longo da evolução uma fonte aumenta ou diminui a sua probabilidade de ser escolhida de acordo com a qualidade dos indivíduos por ela gerados.

O algoritmo foi capaz de ser melhor ou igual a outras três representativas técnicas de otimização com restrições em dez de treze *benchmarks* de problemas. Outro resultado importante encontrado nesses trabalhos foi a redução do número de avaliações necessárias para se chegar próximo ao ótimo global.

Em (YUAN et al., 2008) um algoritmo cultural com espaço populacional baseado em evolução e que usa sequências caóticas é proposto para resolução de problemas de otimização contínua. A principal diferença entre o algoritmo proposto e os algoritmos de Becerra e Coello é o uso de sequências caótica na determinação da taxa de *crossover* e do fator de escala (F). Além disto, o algoritmo apenas utiliza os conhecimentos situacional e normativo. Os resultados obtidos pelo algoritmo se mostraram satisfatórios.

2.2.7.4 Modelagem do Impacto da Cultura nas Interações Sociais e na Solução de Problemas

Em Iacoban, Reynolds e Brewster (2003b) foi investigado como as diversas fontes de conhecimento interagem com o Espaço Populacional representado por enxame de partícu-

las. A função de aceitação era dinâmica. Foram utilizados os cinco tipos básicos de conhecimento. A principal função de influência utilizada é uma roleta estocástica, onde a probabilidade de um conhecimento influenciar o operador de mutação é ajustada de acordo com o valor de aptidão médio dos indivíduos gerados usando esse conhecimento.

A principal contribuição desse trabalho foi a identificação de três fases (conforme descrito na seção 2.2.5) durante a solução do problema que derivam da interação das várias fontes de conhecimento presentes no Espaço de Crenças.

Foi mapeada a tendência central com relação ao tempo (geração) do subconjunto de indivíduos que foi influenciado por cada tipo de conhecimento. O padrão resultante sugere a presença de Enxames Culturais onde várias fontes de conhecimento revezam-se na condução da exploração e refinamento do espaço de busca. Essa interação provê um poderoso veículo para a solução de problemas complexos.

2.2.7.5 Escalonamento de Tarefas

A resolução do problema de escalonamento de tarefas é abordado por um Algoritmo Cultural em Rivera, Becerra e Coello (2007). O espaço populacional utilizado foi composto pela programação evolutiva e foram utilizados dois tipos de conhecimentos: o Conhecimento Situacional e o Conhecimento de Domínio. O Conhecimento de Domínio empregado faz com que os escalonamentos obtidos sejam mais compactos através da redução das pausas de processamento.

Os resultados obtidos com o algoritmo proposto foram comparados com resultados obtidos por algoritmos GRASP (Greedy Randomized Adaptive Search Procedure), Algoritmos Genéticos e Algoritmos Meméticos. O Algoritmo Cultural foi o que obteve o menor desvio médio com relação aos ótimos conhecidos, apesar de não ter sido o algoritmo que encontrou a maioria dos ótimos.

2.2.7.6 Tecido Social

A idéia do tecido social em sistemas culturais é criar e explorar relacionamentos entre os indivíduos da população (ALI, 2008). Estes relacionamentos podem ser representados por redes, assim como ocorre com as redes sociais na *web*. O intuito é que o relacionamento entre os indivíduos interfira na escolha dos conhecimentos que irão influenciá-los. As estruturas das redes (tecidos) sociais utilizados nos algoritmos culturais, por enquanto, se baseiam fortemente nas estruturas adotadas em enxames de partículas (ALI; REYNOLDS,

2009; REYNOLDS; ALI, 2008b). Outra característica atual das redes sociais adotadas é que elas são estáticas, ou seja, permanecem inalteradas ao longo do processo evolutivo.

Os tecidos sociais são implementados como uma nova função de influência (ALI, 2008) e foram aplicados em problemas de otimização contínua na área de engenharia (REYNOLDS; ALI, 2008a) e a um problema de otimização contínua dinâmica (mundo dos cones) (REYNOLDS; ALI, 2008c). Os resultados obtidos dependiam fortemente da estrutura e dos parâmetros da rede adotada. Por isto em Ali, Reynolds e Che (2009) foi proposto o uso de programação genética para acertar os parâmetros e a topologia da rede utilizada no tecido social.

3 *Otimização por Algoritmos Evolucionários*

Dentre as várias áreas englobadas pela Matemática, a Otimização (SUNDARAM, 1996; ABRAHAM; JAIN; GOLDBERG, 2005; COELLO; LAMONT, 2004; BAZARAA; SHERALI; SHETTY, 2006) é uma das que possui a maior quantidade de aplicações práticas. Em especial, toda a área de Pesquisa Operacional é voltada para a resolução de problemas reais através da aplicação de técnicas de otimização.

Neste trabalho pretende-se abordar algumas instâncias de problemas de otimização contínua (mono-objetivo e multiobjetivo) no contexto de geração de energia elétrica. Um problema de otimização contínua pode ser definido como aquele onde as variáveis de decisão são representadas através de valores reais. Na versão mono-objetivo, apenas uma função objetivo deve ser otimizada (ver Seção 3.1). Já problemas multiobjetivo consideram várias funções objetivo (ver Seção 3.3).

Este capítulo pretende discutir alguns conceitos básicos de otimização mono e multi-objetivo no contexto dos Algoritmos Evolucionários (AEs).

3.1 **Princípios de Otimização Mono-objetivo**

De maneira simples pode-se definir um problema de otimização mono-objetivo como sendo um problema no qual o objetivo é uma configuração ou conjunto de parâmetros para os quais o problema apresenta a sua condição ótima, ou seja, um conjunto de parâmetros que não podem ser melhorados no âmbito particular daquele problema (SUNDARAM, 1996).

Mais formalmente pode-se dizer que estes problemas são definidos por:

- **Variáveis de Decisão:** são os valores (parâmetros) que são modificados na busca pela solução do problema;

- **Função Objetivo:** é a função cujas variáveis são as variáveis de decisão e o resultado contém o valor da função que se deseja otimizar. A otimização pode ser uma maximização, caso no qual o valor ótimo é obtido quando a função atinge seu valor máximo dentro do domínio da aplicação, ou de minimização, caso em que o ótimo é o valor mínimo da função. É importante salientar que todo problema de otimização deve possuir pelo menos uma função objetivo;
- **Restrições:** são igualdades ou desigualdades que devem ser respeitadas para se considerar uma solução factível. Um problema de otimização pode conter múltiplas restrições ou não conter nenhuma restrição, caso no qual todas as soluções são consideradas factíveis.

Matematicamente um problema de otimização pode ser expresso por:

$$\begin{aligned}
 & \text{Minimizar/Maximizar } f(\vec{x}) \\
 & \text{sujeito a } g_j(\vec{x}) \geq 0, \quad j = 1, 2, \dots, J; \\
 & \quad h_k(\vec{x}) = 0, \quad k = 1, 2, \dots, K; \\
 & \quad x_i^{(L)} \leq x_i \leq x_i^{(U)} \quad i = 1, 2, \dots, n.
 \end{aligned} \tag{3.1}$$

onde \vec{x} é um vetor de n variáveis de decisão ($x = x_1, x_2, \dots, x_n$) pertencente a X , o espaço de decisões das variáveis. O problema também possui J restrições de desigualdade e K restrições de igualdade, representadas pelas funções de desigualdade ($g_j(\vec{x})$) e igualdade ($h_k(\vec{x})$). Pode-se estabelecer limites inferiores ($x_i^{(L)}$) e superiores ($x_i^{(U)}$) para cada variável de decisão, conforme representado pelo último conjunto de restrições. Uma solução \vec{x} que satisfaz todas as $(J + K)$ restrições e todos os limites para as variáveis de decisão é dita solução factível, caso contrário \vec{x} é chamada de solução infactível. O conjunto de todas as soluções factíveis é chamado de região factível.

3.2 Algoritmos Evolucionários Mono-objetivo

Conforme visto no Capítulo 2, a nomenclatura Algoritmos Evolucionários não é única. Neste Capítulo, optou-se novamente pelo termo Algoritmos Evolucionários para tratar os algoritmos relacionados à busca em um espaço de soluções. A maioria dos algoritmos evolucionários foi proposta para resolver problemas onde um único objetivo é considerado (otimização mono-objetivo). Para mais detalhes ver Jong (2009), Engelbrecht (2007), Grosan e Abraham (2007), Fogel (2005), Cordon, Herrera e Stutzle (2002), Dasgupta Dipankar; Michalewicz (2001), Mitchell e Taylor (1999), Bäck, Schwefel e Informatik (1996), Spears et al. (1993). Um levantamento bibliográfico mais amplo de algoritmos

idealizados para solucionar problemas mono-objetivo está fora do escopo desta tese, mas alguns casos especiais são discutidos nas Seções 2.1.3, 2.2.7 e 4.3.

3.2.1 Testes Estatísticos

Os resultados de algoritmos evolucionários mono-objetivo são estocásticos e podem ser modelados como variáveis aleatórias. Para comparar dois algoritmos evolucionários utilizam-se testes estatísticos sobre indicadores (média, mediana, etc.) das amostras coletadas durante as execuções dos algoritmos.

Segundo García et al. (2009), na comparação entre diferentes abordagens evolucionárias há uma série de condições que devem ser assumidas para o uso seguro de testes estatísticos paramétricos:

- Independência: Em estatística, dois eventos são independentes quando o fato de um ocorrer não modifica a probabilidade de ocorrência do outro.
- Normalidade: Uma observação é normal quando seu comportamento se assemelha a uma distribuição Gaussiana com um certo valor de média μ e variância σ . Um teste de normalidade aplicado sobre uma amostra pode indicar a presença ou ausência destas condições nos dados observados. García et al. (2009) propõem três testes de normalidade:
 - Kolmogorov-Smirnov: Este teste compara a distribuição acumulada dos dados observados com a distribuição acumulada esperada para uma distribuição Gaussiana, obtendo um p -valor¹ baseado nas duas discrepâncias;
 - Shapiro-Wilk: Este teste analisa os dados observados para computar o nível de simetria e formato da curva com o objetivo de computar a diferença com respeito a uma distribuição Gaussiana obtendo o p -valor da soma dos quadrados destas discrepâncias;
 - D’Agostino-Pearson: Este teste, primeiro computa a falta de simetria e o formato da curva para quantificar quão longe de uma distribuição Gaussiana os dados estão em termos de assimetria e formato. Então o teste calcula o quanto cada um destes valores difere do valor esperado para uma distribuição Gaussiana, e computa um único p -valor a partir da soma destas discrepâncias.
- Homocedasticidade (*Homoscedasticity*): Esta propriedade indica a existência da hipótese de igualdade das variâncias. O teste de Levene (LEVENE, 1960) é usado

¹O p -valor de um teste estatístico indica a confiabilidade do teste.

para verificar se as amostras testadas apresentam homogeneidade (*homoscedasticity*) ou heterogeneidade de variâncias (*heteroscedasticity*).

Assumindo que as condições descritas anteriormente são satisfeitas, pode-se aplicar diferentes testes estatísticos paramétricos: teste-t (CASELLA; BERGER, 2001) ou teste-z (CASELLA; BERGER, 2001) para a comparação entre dois algoritmos (amostras); quando se deseja comparar três ou mais algoritmos pode-se utilizar o teste ANOVA (CASELLA; BERGER, 2001).

No caso da violação das condições sugeridas por García et al. (2009), deve-se utilizar um teste não-paramétrico. Os testes não-paramétricos são considerados ferramentas úteis quando os dados resultantes de um experimento não satisfazem as condições discutidas anteriormente. No caso de duas amostras, os testes não paramétricos de *Wilcoxon signed-rank* ou *Mann-Whitney rank sum* podem ser utilizados (GIBBONS, 1985). Em caso de dados não-pareados e um conjunto com mais de duas amostras, o teste de Kruskal-Wallis é bem aceito quando as condições para testes paramétricos são satisfeitas e apresenta um poder maior na presença de distribuições assimétricas, outliers, etc.

O teste de Kruskal-Wallis também é baseado na média dos *ranks* obtidos por cada amostra. Considerando n_i como sendo o tamanho da i -ésima amostra e N como sendo a soma dos tamanhos de todas as amostras, o teste primeiramente obtém o *rank* de cada uma das amostras (R_i , para a i -ésima amostra) e calcula o indicador H conforme a Equação 3.2.

$$H = \frac{12}{N * (N + 1)} * \sum \frac{R_i^2}{n_i} - 3 * (N + 1) \quad (3.2)$$

Se a hipótese nula de que todas as amostras apresentam a mesma distribuição for verdadeira então o indicador H pertence à distribuição χ^2 com o número de amostras - 1 graus de liberdade. Assim a hipótese nula é rejeitada se o valor de H calculado for grande demais para pertencer a distribuição.

Como exemplo do uso do teste de Kruskal-Wallis, pode-se observar a Figura 14.

Pela análise dos intervalos de confiança entre as médias de ranks das três amostras, verifica-se que as amostras 1 e 2 são consideradas semelhantes e ambas estatisticamente diferentes da amostra 3. Ao considerar que os ranks mostrados na Figura 14 indicam a qualidade de algoritmos evolucionários sendo comparados, e quanto mais baixos os valores de rank melhor são os algoritmos associados, tem-se que os algoritmos 1 e 2 possuem desempenho semelhante e ambos são superiores ao algoritmo 3.

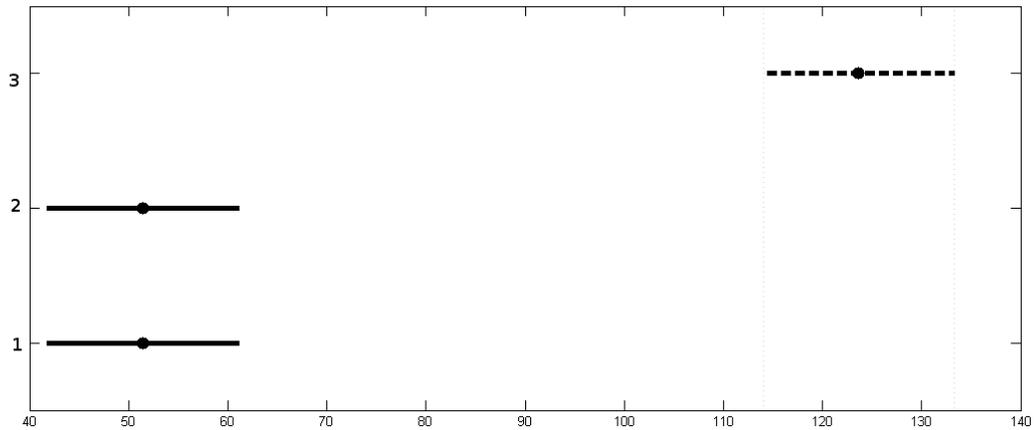


Figura 14: Resultado da comparação entre três amostras.

Apesar dos testes de Kruskal-Wallis e Mann-Whitney serem eficientes, existem formas mais modernas e eficiente para lidar com a não-normalidade dos dados, como o Bootstrap (MOORE et al., 2003).

A idéia de um método Bootstrap vem do fato de que uma amostra representa a população da qual ela foi amostrada. Então, re-amostragens desta amostra representam o que seria obtido se várias amostragens da população fossem feitas. A distribuição de Bootstrap de um indicador é a distribuição obtida para aquele indicador através das várias re-amostragens. Ela representa a distribuição estatística do indicador baseada em várias amostragens da população. Isto é válido por causa do princípio de *plug-in* que diz que se pode utilizar um indicador calculado da amostra para aproximar o mesmo indicador para a população (MOORE et al., 2003). Por exemplo, pode-se usar a média da amostra para estimar a média da população. Como os métodos Bootstrap não impõem nenhuma restrição sobre a amostra, eles podem ser utilizados em qualquer tipo de dado. Geralmente, os métodos Bootstrap são utilizados para calcular intervalos de confiança dos indicadores. Estes intervalos possuem equivalência com os testes estatísticos: se um valor não está contido no intervalo de confiança de um indicador então pode-se afirmar com a mesma precisão com que o intervalo foi gerado que o valor não é equivalente ao indicador (WOOD, 2005). Por exemplo, suponha que um intervalo de confiança para a média da amostra de um algoritmo foi obtido por um método de Bootstrap com 95% de confiabilidade como sendo $[10 - 20]$. Deseja-se então testar se o valor de média X (obtido da amostra de um outro algoritmo) é estatisticamente equivalente à média do primeiro algoritmo. Para isso, basta testar se X está ou não dentro do intervalo: se ele estiver as médias são equivalentes, senão pode-se afirmar com 95% de confiança que existe diferença significativa entre as médias dos dois algoritmos.

Considerando as questões levantadas anteriormente, optou-se neste trabalho pelo uso do teste Kruskal-Wallis (CASELLA; BERGER, 2001; KNOWLES; THIELE; ZITZLER,

2006) para a comparação entre as abordagens mono-objetivo e multiobjetivo propostas (três ou mais amostras sendo comparadas). Para as abordagens mono-objetivo também foi utilizada a abordagem Bootstrap (BRADLEY; TIBSHIRANI, 1994).

3.3 Princípios de Otimização Multiobjetivo no Contexto de AEs

A otimização multiobjetivo trata problemas caracterizados pela presença de múltiplas funções objetivo. Tais problemas, chamados de multiobjetivo ou multicritério, são comumente encontrados em diversas áreas, tais como engenharia, administração, biologia, entre outras (COELLO; LAMONT, 2004; COELLO; LAMONT; VELDHUIZEN, 2007).

Muitas vezes os múltiplos objetivos do problema são conflitantes, ou seja, não existem soluções ótimas no sentido de minimizarem individualmente todas as funções objetivo. Então, a idéia da otimização multiobjetivo consiste em estabelecer uma maneira de tomar decisões que sejam razoáveis aos objetivos mensurados. Assim, utiliza-se a definição de ponto ótimo introduzida por Pareto onde um ponto é considerado Pareto ótimo quando não é possível melhorar nenhum objetivo sem piorar algum outro (ZITZLER; LAUMANN; BLEULER, 2004).

Um Problema de Otimização Multiobjetivo (POM) possui um número de funções objetivo ($M > 1$) as quais deverão ser minimizadas ou maximizadas. Assim como em problemas de otimização mono-objetivo, usualmente há restrições que qualquer solução factível deve atender. Um POM (também chamado de problema de otimização multicritério, multiperformance ou problema de otimização de vetor) pode ser definido conforme a Equação 3.3 (DEB, 2001):

$$\begin{aligned} & \text{Minimizar/Maximizar } f_m(\vec{x}) \quad m = 1, 2, \dots, M; \\ & \text{sujeito a } g_j(\vec{x}) \geq 0, \quad j = 1, 2, \dots, J; \\ & \quad h_k(\vec{x}) = 0, \quad k = 1, 2, \dots, K; \\ & \quad x_i^{(L)} \leq x_i \leq x_i^{(U)} \quad i = 1, 2, \dots, n. \end{aligned} \tag{3.3}$$

A definição de um POM é análoga à definição de um problema de otimização mono-objetivo, porém, existem M funções objetivo $f(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \dots, f_M(\vec{x}))^M$ que podem ser minimizadas ou maximizadas². Essas funções mapeiam o conjunto de soluções factíveis em R^M , com $M > 1$ sendo o número de objetivos. O vetor pertencente a R^M , resultado do mapeamento, será chamado de agora em diante de vetor de adequação ou

²Pelo princípio da dualidade, no contexto de otimização, diz que se pode converter um problema de maximização em um problema de minimização multiplicando sua função objetivo por -1.

ponto. O conjunto imagem R^M é muitas vezes chamado de espaço de objetivos, enquanto o conjunto X , onde estão todas as possíveis soluções do problema a ser resolvido, é chamado espaço de soluções ou espaço de decisões.

Diferentemente do que ocorre nos problemas de otimização mono-objetivo, em problemas de otimização multiobjetivo as funções objetivo formam um espaço objetivo multidimensional, Z . Para cada solução \vec{x} no espaço de decisões das variáveis existe um ponto no espaço objetivo, denotado por $f(\vec{x}) = z = (z_1, z_2, \dots, z_M)^T$. O mapeamento é feito entre um vetor solução n -dimensional e um vetor objetivo M -dimensional. A Figura 15 ilustra o esquema para três variáveis de decisão ($n=3$) em um problema biobjetivo ($M=2$).

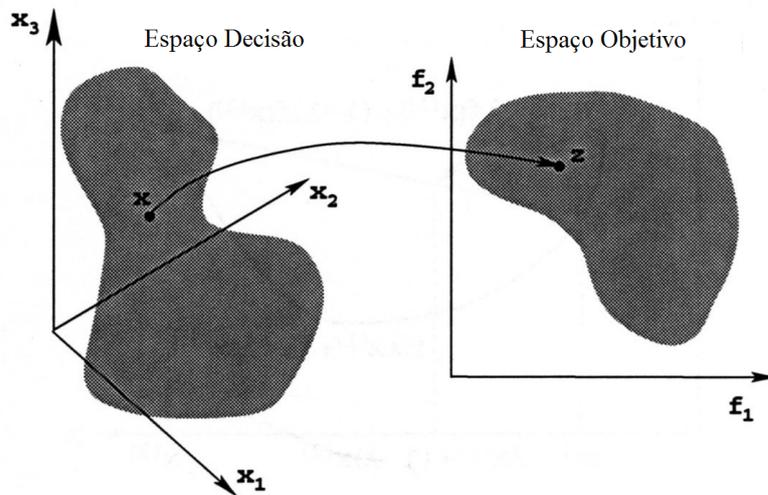


Figura 15: Representação de um espaço de decisões de variáveis e o seu correspondente espaço objetivo (DEB, 2001).

Um POM é dito convexo se todas as suas funções objetivo são convexas e a região factível é convexa (ou todas as restrições de desigualdade são não-convexas e as restrições de igualdade são lineares). Esta característica de um POM é importante, pois existem algoritmos que apresentam bom desempenho em problemas de otimização multiobjetivo convexas, mas enfrentam dificuldades na solução de problemas de otimização multiobjetivo não-convexos (DEB, 2001).

No processo de otimização multiobjetivo não existe necessariamente uma única solução ótima para o problema e sim um conjunto delas. Essa questão fica mais clara ao se observar o exemplo da Figura 16, que mostra um exemplo de otimização multiobjetivo na decisão da compra de um carro.

Os preços dos carros disponíveis variam de dezenas a centenas de milhares de reais, assim como o conforto do carro pode variar também. A Figura 16 mostra que as diferentes soluções produzem balanceamentos *trade-offs* diferentes entre os objetivos. Se o cliente

está interessado apenas no valor do carro, ele decidiria pela solução 1, mas com isso estaria abrindo mão do conforto. Já para clientes que têm por objetivo a compra de carros confortáveis a escolha ideal seria a solução 2, ao custo de um preço maior. Entre essas duas soluções extremas existem outras soluções onde existe um compromisso entre o custo e o conforto do carro (como por exemplo as soluções A, B e C). Como pode-se perceber, entre qualquer par de soluções, uma é melhor que a outra em relação a um objetivo, mas isso acontece somente com a perda de qualidade no outro objetivo.

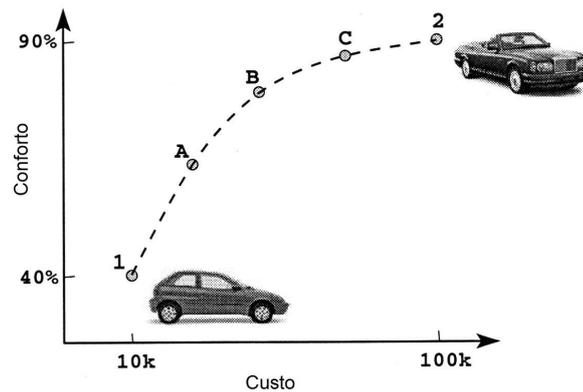


Figura 16: Possível *trade-off* para o problema de decisão da compra de um carro. Adaptado de (DEB, 2001).

Apesar de um otimizador multiobjetivo retornar um conjunto de soluções ótimas, do ponto de vista prático um usuário precisa apenas de uma solução para o seu problema. Portanto, o usuário (ou tomador de decisão) deve escolher entre as soluções *trade-off* aquela que melhor atende às suas necessidades. Esta escolha, geralmente leva em consideração informações qualitativas e não técnicas sobre o problema que permitem avaliar os prós e contras de cada uma dessas soluções encontradas. Considerando o problema da compra de um carro visto anteriormente, essas informações poderiam ser a distância percorrida com o carro diariamente, a quantidade de financiamento disponível, o consumo médio e o preço do combustível entre outros fatores. Este processo de tomada de decisão pode ser dispendioso, portanto existem formas de automatizá-lo (SISKOS; SPYRIDAKOS, 1999). Neste trabalho a tomada de decisão não será automatizada e as soluções das instâncias multiobjetivo do problema considerado, serão mostradas na forma de fronteira de Pareto ou soluções extremas (ver Capítulo 6).

3.3.1 Dominância e Pareto-otimalidade

A maioria dos algoritmos de otimização multiobjetivo usa o conceito de dominância em seu processo de busca. Esta subseção define o conceito de dominância e os termos relacionados a ele.

3.3.1.1 Soluções Especiais

Algumas soluções especiais ou vetores especiais, são geralmente utilizados por algoritmos de otimização multiobjetivo. Entre essas soluções estão: solução/vetor ideal, solução/vetor utopia e solução/vetor nadir. Para cada um dos M objetivos existe uma solução ótima diferente. O vetor composto por essas soluções ótimas é dito vetor objetivo ideal (z^*), este vetor é ilustrado na Figura 17. Na prática, esta solução não existe (é infactível), pois a menos que todos os objetivos sejam não-conflitantes não é possível representar a solução ótima para todos os objetivos com um único vetor. Portanto, este vetor é utilizado como um vetor de referência para os algoritmos de otimização multiobjetivo (KNOWLES; CORNE; DEB, 2008).

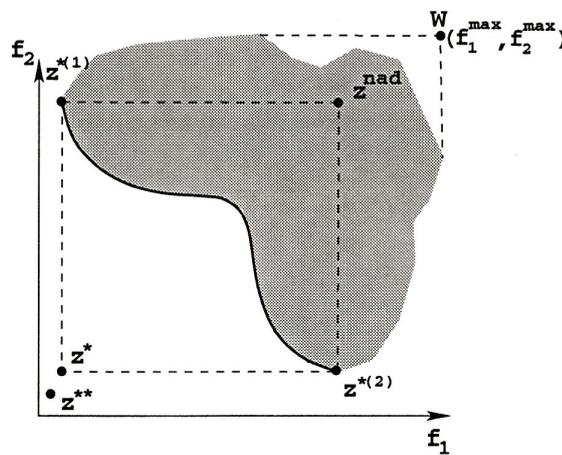


Figura 17: Vetores objetivo ideal (z^*), utopia (z^{**}) e nadir (z^{nad}) (DEB, 2001).

O vetor objetivo chamado de utopia (z^{**}) possui todos os seus componentes ligeiramente menores que o vetor objetivo ideal, ou seja, $z^{**} = z^* - \epsilon$ com $\epsilon \geq 0$ para todo $i = 1, 2, \dots, M$. A Figura 17 mostra um vetor utopia. Assim como o vetor objetivo ideal, o vetor objetivo utopia também é uma solução inexistente (DEB, 2001).

O vetor objetivo dito nadir (z^{nad}) é composto pelos limites superiores para cada objetivo considerando todo o conjunto Pareto-ótimo (diferentemente do vetor ideal que considera os limites inferiores para todo o espaço de busca factível) (KNOWLES; CORNE; DEB, 2008). Este vetor é difícil de ser computado e pode ser uma solução factível ou infactível dependendo da continuidade e da convexidade do conjunto Pareto-ótimo. Um vetor objetivo nadir também pode ser observado na Figura 17, cabe salientar que o vetor (z^{nad}) não está relacionado ao ponto (W) correspondente aos limites superiores das funções objetivo (DEB, 2001).

3.3.1.2 Pareto Dominância e Fronteiras não-dominadas

Quando os algoritmos evolucionários fazem uso do conceito de dominância em seu processo de busca duas soluções são comparadas no sentido de verificar se uma solução domina a outra ou não. Considerando um problema de minimização com M funções objetivo, uma solução a domina outra solução b (matematicamente $a \prec b$) se a não é pior que b em nenhum dos M objetivos e a é estritamente melhor que b em pelo menos um objetivo. No caso de a ser melhor que b em algum objetivo, mas b ser melhor que a em outro objetivo, diz-se que as duas soluções são incomparáveis ou não-dominadas. A Tabela 1 representa matematicamente o conceito de dominância, considerando um problema de minimização de todos os objetivos.

Tabela 1: Relações possíveis entre duas soluções $x, y \in X$. As relações são baseadas nos vetores de adequação $a = f(x)$ e $b = f(y)$ (KNOWLES; THIELE; ZITZLER, 2006)

Relação	Simbologia	Interpretação no espaço dos objetivos
domina estritamente	$x \prec\prec y$	x é melhor que y em todos os objetivos, ou seja, $\forall i \mid 0 < i \leq m \Rightarrow x_i < y_i$
domina	$x \prec y$	x não é pior que y em todos os objetivos e é melhor em ao menos um, ou seja $\forall i \mid 0 < i \leq m \Rightarrow x_i \leq y_i$ $\wedge \exists k \mid 0 < k \leq m \Rightarrow x_k < y_k$
domina fracamente	$x \preceq y$	x não é pior que y em nenhum objetivo, ou seja, $\forall i \mid 0 < i \leq m \Rightarrow x_i \leq y_i$
incomparáveis	$x \parallel y$	nem $x \preceq y$ e nem $y \preceq x$
indiferentes	$x \sim y$	x é igual a y em todos os objetivos, ou seja, $\forall i \mid 0 < i \leq m \Rightarrow x_i = y_i$

Na Figura 18 o retângulo cinza claro engloba a região do espaço objetivo que domina o vetor B. O retângulo cinza escuro contém os vetores objetivo que são dominados pelo vetor B. Todos os vetores que não estão nestes retângulos são incomparáveis em relação a B.

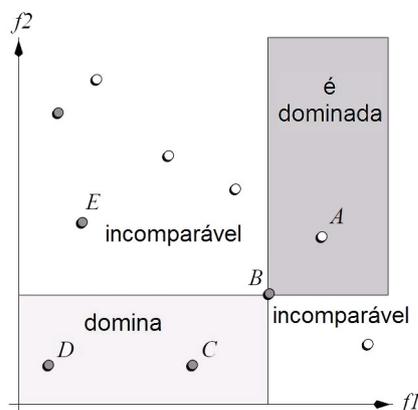


Figura 18: Exemplos de relações de dominância entre soluções no espaço objetivo (ZITZLER, 1999).

Um vetor de decisão $x \in X$ é dito não-dominado com relação a um conjunto $A \subseteq X$ se $\nexists a \in A : a \prec x$. Além disso, x é considerado Pareto ótimo se x é não-dominado com relação a X . O conjunto de todas as soluções Pareto ótimas é chamado de conjunto Pareto ótimo (ou conjunto de soluções eficientes) e os vetores correspondentes no espaço objetivo formam a superfície ou fronteira de Pareto.

O conjunto Pareto ótimo é composto pelas soluções ótimas globais. Porém, assim como em problemas com um único objetivo, existem conjuntos não-dominados compostos por soluções ótimas locais, ou seja, soluções não-dominadas entre si, mas dominadas por um outro conjunto de soluções.

Cabe salientar que existem múltiplas soluções Pareto ótimas para um problema multiobjetivo somente se os objetivos são conflitantes entre si (DEB, 2001). Se os objetivos não são conflitantes o conjunto Pareto ótimo é composto por apenas uma solução (por exemplo a solução z^* na Figura 17 para um problema de minimização), pois a solução mínima para todos os objetivos é a mesma (ao otimizar um objetivo, automaticamente otimizam-se todos os objetivos).

3.3.2 Métodos não Baseados em Dominância de Pareto

Os métodos não baseados em dominância de Pareto para a geração do conjunto Pareto-ótimo agregam os múltiplos objetivos em um único objetivo. Entretanto, os parâmetros destas funções não são definidos pelo tomador de decisão, mas sim variados sistematicamente pelo próprio algoritmo otimizador. Nestes métodos o conjunto de soluções que aproximam o conjunto Pareto-ótimo é gerado por diversas execuções do algoritmo otimizador com diferentes configurações de parâmetros.

Alguns exemplos de métodos não baseados em dominância de Pareto para a aproximação do conjunto Pareto-ótimo são: método da soma ponderada, método ϵ -constraint, *goal programming* (DEB, 2001), método de Benson (DEB, 2001) e método *value function* (DEB, 2001). A seguir os métodos de soma ponderada e ϵ -constraint serão brevemente discutidos.

3.3.2.1 Soma Ponderada

Na soma ponderada um POM é convertido em um problema de otimização mono-objetivo através de uma combinação linear de seus objetivos. Esta conversão é represen-

tada na Equação 3.4 (DEB, 2001).

$$\begin{aligned}
 &\text{Minimizar } F(x) = \sum_{m=1}^M w_m f_m(x) \\
 &\text{sujeito a } g_j(x) \geq 0, j = 1, 2, \dots, J; \\
 &\quad h_k(x) = 0, k = 1, 2, \dots, k; \\
 &\quad x_i^{(L)} \leq x_i \leq x_i^{(U)} i = 1, 2, \dots, n.
 \end{aligned} \tag{3.4}$$

Onde w_m é o peso da m -ésima função objetivo. É usual escolher pesos tais que $\sum_{m=1}^M w_m = 1$. Um conjunto de soluções é obtido ao se resolver o problema acima para diferentes combinações de pesos (ZITZLER, 1999).

Quando o algoritmo otimizador é um algoritmo exato é provado que todas as soluções encontradas pelo método fazem parte do conjunto Pareto-ótimo (ZITZLER, 1999).

A vantagem deste método é a sua simplicidade de implementação e entendimento, além da garantia de encontrar soluções no conjunto Pareto-ótimo para problemas com fronteira de Pareto convexa³, desde que o otimizador mono-objetivo seja capaz de encontrar soluções ótimas (DEB, 2001).

As soluções ótimas para esse problema escalarizado são chamadas de soluções suportadas e pertencem ao conjunto de soluções eficientes, porém podem existir soluções eficientes que não são ótimas para nenhum vetor w (denominadas soluções não-suportadas). A principal desvantagem deste método é a sua incapacidade de gerar todas as soluções Pareto-ótimas quando a superfície da fronteira de Pareto não é convexa (DEB, 2001). Além disso, ao se utilizar um vetor de pesos uniformemente distribuído não necessariamente o conjunto de soluções Pareto-ótimas associadas a esse vetor será também uniformemente distribuído (DEB, 2001). Então, é difícil identificar o vetor de pesos para a geração de soluções em uma determinada região da fronteira.

3.3.2.2 ϵ -Constraint

Devido às dificuldades encontradas pelo método de soma ponderada na resolução de problemas com espaços objetivo não-convexos, o método **ϵ -constraint** é normalmente utilizado nestes casos. O método reformula um problema de otimização multiobjetivo considerando apenas um objetivo e tratando o restante como restrições, sendo que os

³Uma fronteira de Pareto é dita convexa se o conjunto de soluções Pareto-ótimas que originou está fronteira se comporta de maneira convexa.

valores das restrições são especificados pelo usuário (DEB, 2001). O método pode ser formulado conforme a Equação 3.5.

$$\begin{aligned}
 & \text{Minimizar } f_{\mu}(x), \\
 & \text{sujeito a } f_m \leq \varepsilon_m \quad m = 1, 2, \dots, M \text{ e } m \neq \mu; \\
 & \quad g_j(x) \geq 0, \quad j = 1, 2, \dots, J; \\
 & \quad h_k(x) = 0, \quad k = 1, 2, \dots, k; \\
 & \quad x_i^{(L)} \leq x_i \leq x_i^{(U)} \quad i = 1, 2, \dots, n.
 \end{aligned} \tag{3.5}$$

A vantagem do **ε -constraint** consiste no fato do método poder ser aplicado tanto a problemas com espaços objetivo convexos quanto não-convexos. Em termos de informações necessárias para a utilização do método ele é semelhante à soma ponderada. O primeiro necessita do vetor de pesos, enquanto o segundo necessita de um vetor de valores épsilon (ε).

A principal desvantagem do método reside na dependência da escolha do vetor épsilon para o seu sucesso. Além disso, quanto maior é o número de objetivos maior é o número de elementos no vetor épsilon, o que dificulta ainda mais a sua escolha.

Existem algoritmos que permitem que a escolha do vetor épsilon seja feita de forma automática, o que torna o método útil do ponto de vista prático (LAUMANNNS; THIELE; ZITZLER, 2006).

3.4 Algoritmos Evolucionários Multiobjetivo

Os algoritmos evolucionários multiobjetivo (AEMO) aparecem como alternativa aos métodos clássicos de otimização. Estes métodos clássicos podem ser definidos como sendo aqueles que trabalham somente com a transformação de uma única solução a cada iteração e estas transformações são determinísticas (geralmente envolvendo informações do gradiente da função) (KORTE; VYGEN, 2008). Entretanto, os métodos clássicos de otimização comumente encontram dificuldades na solução de problemas, tais como:

- A convergência em uma solução ótima depende da escolha da solução inicial;
- A maioria dos algoritmos tende a estagnar em ótimos locais;
- Um método que é eficiente na solução de um determinado problema, pode não ser eficiente na solução de outro problema;

- Os métodos clássicos, em geral, não são eficientes na solução de problemas discretos.

Portanto, os métodos clássicos encontram dificuldades na solução de problemas práticos de otimização, a menos que sejam feitos grandes ajustes. Alternativamente, os algoritmos evolucionários (AEs) solucionam os problemas de otimização multiobjetivo imitando os processos evolucionários naturais para guiar seu processo de busca. Diferentemente dos métodos clássicos, os AE processam uma população de soluções a cada iteração. Desta forma, um algoritmo evolucionário é capaz de retornar uma saída composta por diversas soluções. Esta habilidade de encontrar múltiplas soluções ótimas em uma única iteração favorece o uso de AEs na solução de problemas de otimização multiobjetivo (COELLO; LAMONT; VELDHUIZEN, 2007), pois a solução destes problemas requer um conjunto de soluções com diferentes compromissos entre os objetivos.

3.4.1 Características dos Algoritmos Evolucionários Multiobjetivo

Devido à complexidade da maioria dos problemas multiobjetivo, muitas vezes não é possível gerar todo o conjunto Pareto ótimo de forma eficiente. Portanto, o que se busca na otimização de um POM pode ser reformulado conforme a seguir:

- A distância do conjunto não-dominado resultante para a fronteira de Pareto ótima deve ser minimizada;
- É desejada uma boa distribuição (geralmente uniforme) das soluções encontradas;
- O espalhamento da fronteira de Pareto encontrada deve ser maximizado, ou seja, para cada objetivo um grande intervalo de valores deve ser coberto pelas soluções não-dominadas.

De acordo com esses três objetivos duas questões que precisam ser consideradas na construção de algoritmos evolucionários multiobjetivo (AEMO) são (ZITZLER, 1999): como modelar uma função de avaliação e de seleção para guiar o processo de busca pelas soluções pertencentes ao conjunto Pareto ótimo e como manter a diversidade da população evitando a convergência prematura e permitindo que a fronteira de Pareto obtida tenha pontos bem distribuídos e bem espalhados.

3.4.1.1 Avaliação e Seleção de Soluções em AEMO

Em problemas de otimização com múltiplas funções objetivo o cálculo da adequação (*fitness*) e a seleção das soluções deve levar em consideração os vários objetivos do problema. Existem diversas formas de realização do processo de seleção e do cálculo do *fitness* em um AEMO, alguns exemplos são brevemente descritos a seguir (ZITZLER, 1999):

- **Seleção com agregação dos objetivos e variação dos parâmetros:** implementa um método clássico para encontrar a fronteira de Pareto, onde os objetivos são combinados em uma única função objetivo parametrizada para a avaliação e seleção de uma solução. Esses parâmetros correspondem aos pesos designados para cada objetivo. Para que diversos pontos sejam obtidos é necessário a execução do algoritmo otimizador diversas vezes com os parâmetros da função objetivo variados sistematicamente;
- **Seleção por escolha dos objetivos:** ao invés de combinar os objetivos em uma única função objetivo, esta técnica seleciona um dos objetivos durante a etapa de seleção. Assim, a cada iteração um dos objetivos tende a decidir qual elemento da população fará parte do conjunto de soluções;
- **Seleção baseada no conceito de Pareto:** avalia e seleciona uma solução com base no conceito de Pareto dominância. Esta técnica é uma das mais utilizadas e teoricamente é capaz de encontrar qualquer fronteira de Pareto, porém é sensível ao número de funções objetivo conflitantes do problema e ao tamanho da fronteira de Pareto.

Neste trabalho são utilizadas as técnicas baseadas em agregação dos objetivos e baseadas em dominância de Pareto.

3.4.1.2 Diversidade da População de AEMO

Manter a diversidade da população é crucial para a eficácia dos AEMO. Algoritmos evolucionários clássicos tendem a convergir na direção de uma única solução e perdem soluções que poderiam compor a fronteira de Pareto. Portanto, existem técnicas para manter a diversidade da população, melhorando o desempenho dos AEMO. Alguns exemplos destas técnicas são brevemente descritos a seguir (ZITZLER, 1999):

- **Compartilhamento de *fitness*:** promove a formulação e a manutenção de subpopulações (nichos). O indivíduo compartilha o seu *fitness* com todos os seus vizinhos, ou seja, quanto mais populoso é um nicho mais degradado será o *fitness* dos

seus indivíduos. Uma vizinhança é definida em termos de uma medida de distância e especificada por um raio. Corresponde a uma das técnicas para geração de diversidade mais utilizadas, estando presente na maioria do AEMO;

- **Isolamento por distância:** separa os indivíduos em regiões. As regiões podem ser evoluídas em uma única população ou evoluídas separadamente (neste caso, cada uma correspondendo a uma população diferente);
- **Reinicialização:** reinicializa toda a população ou apenas parte dela após um certo período de tempo ou quando a busca encontra-se estagnada;
- **Superespecificação:** nesta técnica um indivíduo possui partes ativas e inativas. As partes ativas são aquelas que especificam o vetor solução enquanto as partes inativas são redundantes e não têm função. No entanto, as partes inativas podem tornar-se ativas e vice-versa;
- **Crowding:** nesta técnica o *fitness* de um indivíduo depende de quão povoada (*crowded*) é a região onde ele se encontra, indivíduos em regiões menos povoadas possuem um melhor valor de *fitness*. O *crowding* pode ser medido tanto no espaço objetivo quanto no espaço decisão.

Neste trabalho é utilizada a técnica *crowding* para a geração da diversidade da população (implementada pelo algoritmo de armazenamento de soluções não dominadas AGA, conforme detalhado no Capítulo 5).

3.4.1.3 Elitismo

A técnica de elitismo mostra-se eficiente em grande parte dos algoritmos evolucionários e consiste em manter a melhor solução encontrada pelo algoritmo a cada iteração. O elitismo em AEMO corresponde a manter um conjunto de soluções de elite (em geral não comparáveis entre si). As duas principais políticas na construção deste conjunto de elite em AEMO: guardar as melhores soluções para cada um dos objetivos a cada iteração do algoritmo ou então guardar as soluções não-dominadas até o momento a cada iteração do algoritmo. Neste trabalho é utilizado o segundo caso, ou seja, as soluções não-dominadas são armazenadas.

3.4.2 Análise e Comparação de Resultados em AEMO

As medidas de desempenho de algoritmos evolucionários multiobjetivo levam em consideração os objetivos da otimização de um problema multiobjetivo, isto é, a distância

da fronteira de Pareto ótima, e uma boa distribuição e espalhamento dos pontos. Existem basicamente três aspectos para a análise da eficiência do desempenho de AEMO, são eles: *ranking* de dominância, função empírica de conquista (*empirical attainment function*) e indicadores de qualidade sobre a fronteira de Pareto. A seguir algumas medidas de comparação sobre estes aspectos serão brevemente descritos.

3.4.2.1 *Ranking* de Dominância

Considerando-se todos os conjuntos de aproximação resultantes das execuções de cada um dos AEMO que se deseja comparar o desempenho é possível determinar uma relação de ordem parcial entre esses conjuntos. Ou seja, pode-se determinar que alguns conjuntos são melhores, piores ou incomparáveis com relação a outros. Essa ordem parcial pode ser utilizada para determinar um *rank* para cada um dos elementos dos conjuntos, quanto menores os *ranks* melhor é o conjunto. Esses *ranks* podem ser relativos a diferentes relações de dominância, tais como:

- Por quantos indivíduos um indivíduo é dominado?
- Quantos indivíduos um indivíduo domina?
- Qual o valor obtido pela aplicação de algum indicador a um indivíduo?

Baseando-se nos *ranks* obtidos por cada AEMO e utilizando testes estatísticos baseados em *rank* pode-se inferir qual ou quais AEMO são estatisticamente melhores.

O *ranking* de dominância se baseia somente no conceito de dominância de Pareto e, portanto, é independente de informações de preferência (diferentemente de outras métricas como o hipervolume que será tratado adiante). Esta métrica é recomendada porque se um AEMO obtém diferença significativa nesta métrica já é possível concluir que ele é melhor ou pior que os demais (KNOWLES; THIELE; ZITZLER, 2006). Caso a diferença não seja significativa deve-se utilizar as outras métricas. De qualquer modo, outras métricas podem agregar mais informações sobre as diferenças de desempenho.

3.4.2.2 Função Empírica de Conquista (*Empirical Attainment Function*)

Uma função de conquista (*attainment function*) é uma medida de momento de primeira ordem (como a média e a mediana) para a avaliação de resultados de AEMO (FONSECA; FONSECA; HALL, 2001). Elas requerem dados estatísticos sobre a evolução dos AEMO

para apresentar uma medida gráfica da eficiência destes algoritmos. Uma função de conquista empírica define uma superfície que divide o espaço objetivo em metas (pontos), como por exemplo a fronteira de Pareto ótima, e a probabilidade de atingir cada meta.

Pode-se utilizar esta métrica para identificar em quais regiões do espaço objetivo um conjunto de aproximação é melhor que o outro e qual a probabilidade disto acontecer. Por aproximar a distribuição de probabilidade de um algoritmo estocástico, uma função de conquista pode responder perguntas do tipo: qual a probabilidade de se encontrar soluções que dominem um determinado ponto em uma única execução ou em quantas execuções, aproximadamente, serão encontradas soluções dominadas por um determinado ponto. Nesse sentido a função de conquista é uma métrica mais robusta que as demais, porém possui um custo computacional elevado (COELLO; LAMONT; VELDHUIZEN, 2007).

Um exemplo do uso de função de conquista para a comparação entre conjuntos de aproximação da fronteira de Pareto pode ser visto na Figura 19. É possível observar que o conjunto A é melhor que o conjunto B na região central do espaço objetivo, enquanto o conjunto B apresenta melhores resultados na extremidades do espaço objetivo. As diferenças são representadas em escalas de cinza: quanto mais escura a região maior é a diferença⁴.

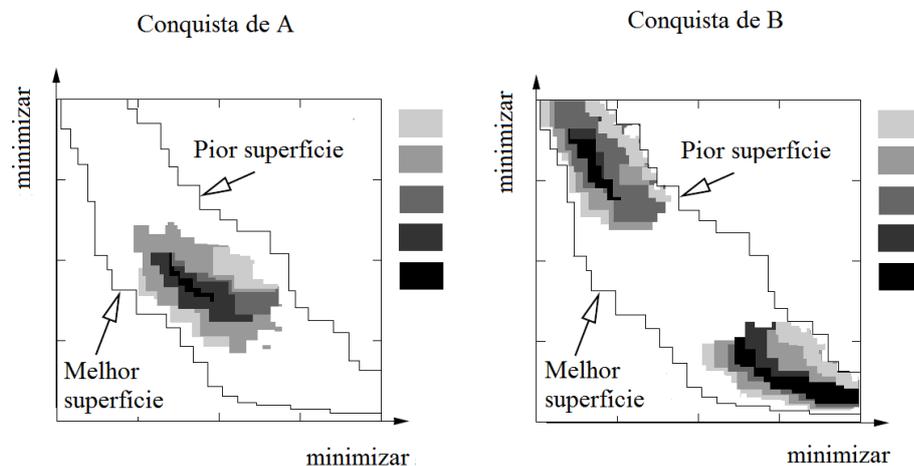


Figura 19: Exemplo do uso de função de conquista (KNOWLES; THIELE; ZITZLER, 2006)

Este método de análise e comparação dos resultados não foi utilizado neste trabalho devido a sua complexidade de implementação e apresentação.

⁴Essas diferenças também poderiam ser mostradas em termos percentuais.

3.4.2.3 Indicadores de Qualidade sobre a Fronteira de Pareto

Os indicadores de qualidade são funções que atribuem um número real a um ou mais conjuntos de aproximação. Os indicadores de qualidade podem ser unários, quando analisam somente um conjunto, ou podem ser n-ários, quando analisam de uma vez n conjuntos de aproximação.

Sua modelagem matemática unifica vários indicadores já existentes na literatura e apresenta um resultado bastante interessante: nem todos os indicadores de qualidade podem ser utilizados para fazer afirmações como $A \prec\prec B$, ou “o conjunto de soluções A é estritamente melhor que o conjunto de soluções B”. Existe até o caso de indicadores que afirmam que A supera B mesmo quando todas as soluções de A são dominadas por uma ou mais soluções de B - um indicador simples para entender este caso é o indicador “número de soluções”, que mapeia um conjunto de soluções no número de soluções do conjunto, e onde se entende que quanto maior o número de soluções, melhor. Indicadores que podem afirmar “A é melhor que B” mesmo quando $B \prec\prec A$ (ou seja, podem dar indicações falsas a respeito de qual algoritmo é melhor) são chamados indicadores Pareto *non-compliant* (ou indicadores Pareto não-concordantes).

Alguns indicadores unários são bastante utilizados por possuírem algoritmos publicamente disponíveis para seu cálculo e por serem Pareto *compliant*, dentre eles estão (KNOWLES; THIELE; ZITZLER, 2006): o indicador hipervolume (I_H), o indicador ε -unário (I_ε), o indicador ε -unário ($I_{\varepsilon+}$), os indicadores R_2 (I_{R_2}) e R_3 (I_{R_3}). Todos eles são executados em relação a um conjunto de referência R, normalmente um conjunto melhor do que os que estão sendo analisados. Mais detalhes conforme descrito a seguir.

- **Indicador Hipervolume (I_H):**

este indicador proposto por Zitzler e Thiele (1999) mede o hipervolume (o qual deve ser maximizado) da porção do espaço objetivo que é fracamente dominada por um conjunto de aproximação A. Para que esse cálculo possa ser feito, o espaço de objetivos deve ser limitado por um ponto limitante, chamado ponto nadir, dominado por todos os outros pontos. Uma das grandes desvantagens do indicador I_H é o tempo computacional para seu cálculo, que cresce exponencialmente com o número de objetivos. Cabe observar que algumas implementações deste indicador consideram a diferença do hipervolume com relação a um conjunto de referência (R), neste caso o valor resultante deve ser minimizado. Um exemplo do indicador I_H pode ser visto na Figura 20. Nesta figura observa-se que o limite das áreas se dá pelo ponto de referência (reference) e que a área dominada pelo ponto A é menor do que a área dominada pelo ponto B, ou seja, $I_H(A) < I_H(B)$;

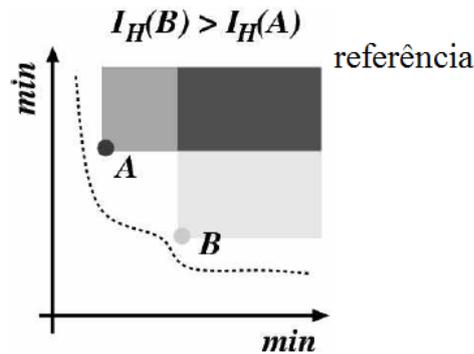


Figura 20: Exemplo do indicador hipervolume (KNOWLES; THIELE; ZITZLER, 2006).

- **Indicadores Épsilon-Unário (I_ϵ) e ($I_{\epsilon+}$):** os indicadores ϵ -unário aditivo ($I_{\epsilon+}$) e multiplicativo (I_ϵ) foram propostos por Zitzler et al. (2003) ⁵. O indicador calcula o menor valor e que, quando multiplicado/adicionado a todas as soluções do conjunto de referência R , faz com que este passe a ser fracamente dominado. Assim, deseje-se que a saída deste indicador seja minimizada. Este indicador é barato de ser computado;
- **Indicadores (I_{R_2}) e (I_{R_3}):** foram propostos por Hansen e Jaskiewicz (1998). Se utilizam de uma série de funções de utilidade (*utility functions*) para calcular várias possíveis preferências do tomador de decisão e analisar quão bem elas estão sendo atingidas. Para isso, são utilizados vetores de escalarização que parametrizam as funções de utilidade. A Equação 3.4.2.3 define matematicamente este indicador. A utilidade $u(\tilde{\lambda}, B)$ do conjunto de aproximação A , no vetor de pontos escalarizados $\tilde{\lambda}$ é a distância mínima entre um ponto do conjunto A e um ponto de referência (esta distância é medida através da projeção deste ponto no vetor de escalarização);

$$R2 = \frac{\sum_{\tilde{\lambda} \in V} u(\tilde{\lambda}, B) - u(\tilde{\lambda}, A)}{|\tilde{\lambda}|} \quad (3.6)$$

Os indicadores de qualidade primeiramente reduzem os conjuntos de aproximação a um único valor de qualidade. Na sequência, testes estatísticos são aplicados sobre as amostras destes valores de qualidade. Os testes estatísticos para a comparação entre os indicadores de qualidade serão os mesmos descritos na Subseção 3.2.1.

⁵Estes indicadores também existem na forma binária.

3.4.3 Classificação de Algoritmos Evolucionários Multiobjetivo

Uma possível classificação para os AEMO foi proposta por Cohon e Marks (1975) e leva em consideração as duas etapas necessárias para a solução de um problema multiobjetivo: encontrar um conjunto de soluções de compromisso (*trade-off solutions*) e a tomada de decisão sobre qual será a real solução do problema dentre as soluções encontradas. Com base nesta classificação as técnicas são agrupadas de acordo com a etapa na qual a informação de alto nível sobre o problema é utilizada para a tomada de decisão.

Os AEMOs que se encaixam no grupo chamado de preferências *a priori* exige que informações sobre a ordem dos objetivos ou os próprios objetivos a serem atingidos sejam definidos antes do processo de busca pelas soluções (COELLO; LAMONT; VELDHUIZEN, 2007). Exemplos deste grupo de AEMO são: *goal programming*, método lexicográfico e método do critério global.

O grupo preferências *a posteriori* engloba AEMO que não requerem informações não técnicas sobre o problema antes da busca pelas soluções *trade-off* (COELLO; LAMONT; VELDHUIZEN, 2007). Exemplos de algoritmos que se encaixam deste grupo são: ϵ -constraint e a combinação linear dos pesos para cada função objetivo.

As técnicas que fazem parte do grupo chamado de preferências progressivas permitem uma interação com o tomador de decisão ao longo do processo de busca pelas soluções *trade-off* (COELLO; LAMONT; VELDHUIZEN, 2007). Estes algoritmos geralmente operam em três etapas: encontram uma solução não-dominada; obtêm a opinião do tomador de decisão com relação a essa solução e modificam esta solução de acordo com as preferências do tomador de decisão; por fim, repetem esses dois passos anteriores até que o tomador de decisão esteja satisfeito ou que não seja possível obter soluções melhores.

3.4.4 Exemplos de Algoritmos Evolucionários Multiobjetivo

Existem diversos algoritmos evolucionários multiobjetivo e um levantamento bibliográfico exaustivo destes se encontra fora do escopo deste trabalho (para um bom levantamento bibliográfico sobre AEMOs ver <http://www.lania.mx/~ccoello/EMOO/>). Aqui serão apresentados alguns algoritmos que são empregados na solução de problemas de despacho econômico/ambiental de energia (ver Capítulo 4, Subseção 4.3.2) ou que utilizam sistemas imunológicos artificiais (Subseção 3.4.4.1) ou algoritmos culturais (Subseção 3.4.4.2).

O NSGA (Nondominated Sorting Genetic Algorithm) é um algoritmo genético multiobjetivo (SRINIVAS; DEB, 1994). Os indivíduos da população têm seu *fitness* assinalado

com base no *rank* da fronteira a qual ele pertence e em um mecanismo de *fitness sharing* aplicado aos indivíduos de uma mesma fronteira, assim indivíduos mais próximos da fronteira de Pareto e mais diversos são privilegiados (possuem *fitness* mais elevado). Os demais passos do algoritmo (*crossover*, mutação e seleção) são análogos aos passos de um algoritmo genético mono-objetivo.

Apesar do nome, o NSGA-II (Nondominated Sorting Genetic Algorithm-II) (DEB et al., 2002) possui poucas semelhanças com o NSGA. Para cada indivíduo da população é assinalado um *rank* baseado na fronteira a qual ele pertence e um valor de *crowding*. O *crowding* de um indivíduo é calculado pela distância média entre os pontos adjacentes a ele em cada direção para todos os objetivos. Um torneio binário baseado em *rank* e *crowding* é aplicado para escolher os indivíduos que sofrerão os efeitos dos operados genéticos na próxima geração. Um indivíduo ganha o torneio se possuir um menor *rank* ou, caso os *ranks* sejam iguais, se o seu valor de *crowding* for inferior. Após a aplicação dos operadores genéticos, a população anterior e os filhos gerados são combinados (formando um conjunto de $2 \cdot N$ indivíduos, onde N é o tamanho da população). Os *ranks* e valores de *crowding* são recalculados. Os indivíduos são então ordenados lexicograficamente⁶ de acordo com o seu *rank* e valor de *crowding*. O NSGA-II é o algoritmo evolucionário multiobjetivo mais utilizado como referência em comparações com novas abordagens (COELLO; LAMONT; VELDHUIZEN, 2007).

O SPEA (Strength Pareto Evolutionary Algorithm) (ZITZLER; THIELE, 1999) incorpora conceitos de elitismo através do uso de um arquivo de soluções não-dominadas. A cada geração do algoritmo os indivíduos não-dominados são copiados para o arquivo. Para cada indivíduo do arquivo é calculado um valor de *strength* que é proporcional a quantidade de indivíduos da população que ele domina. O *fitness* de cada indivíduo da população é computado de acordo com o valor de *strength* dos indivíduos do arquivo que o domina. Para manter o número de indivíduos armazenados no arquivo de soluções não-dominadas abaixo de um certo limite máximo, toda vez que este limite é superado um algoritmo de truncamento baseado em *clusterização* é aplicado para reduzir o tamanho do arquivo para o limite máximo. Uma nova versão chamada de SPEA2 é proposta em Zitzler, Laumanns e Thiele (2001). Nesta nova versão o *fitness* leva em consideração também o número de indivíduos que cada indivíduo domina, o procedimento de truncamento foi aprimorado para melhor explorar os extremos dos objetivos e uma técnica de estimativa de densidade foi incorporada. O sucesso do SPEA motivou muitos dos seus sucessores a também utilizarem um arquivo de solução não-dominadas.

⁶Na ordenação lexicográfica os elementos são ordenados de acordo com seus valores para diferentes componentes, na ordem em que estes componentes são definidos. Por exemplo, se os componentes são *rank* e *crowding*, os elementos são ordenados com relação ao *rank* e, para os elementos que possuem o mesmo *rank*, de acordo com o valor de *crowding*.

A idéia principal do NPGA (Niched Pareto Genetic Algorithm) (HORN; NAFPLIOTIS; GOLDBERG, 1994) é utilizar um torneio binário baseado em dominância de Pareto no processo de seleção. Neste torneio dois indivíduos são escolhidos aleatoriamente e comparados com um subconjunto dos indivíduos da população. Se um dos indivíduos é dominado e o outro, ele é escolhido; caso contrário o resultado do torneio é decidido por um procedimento de *fitness sharing*. O restante do algoritmo corresponde a um AG convencional.

3.4.4.1 Sistemas Imunológicos Artificiais Multiobjetivo

Nos últimos anos surgiram abordagens baseadas em princípios imunológicos para a resolução de problemas multiobjetivo. Segue uma breve descrição de algumas destas abordagens. Para mais detalhes ver (CAMPELO; IGARASHI, 2007; FRESCHI; COELLO; REPETTO, 2009b).

O algoritmo MISA (Multiobjective Immune System Algorithm) pode ser considerado o primeiro Sistema Imunológico Artificial Multiobjetivo (SIAMO) proposto na literatura (FRESCHI; COELLO; REPETTO, 2009b). Este algoritmo foi inicialmente proposto por Coello e Cortés (2002). A primeira versão do algoritmo é bastante fiel ao princípio da seleção clonal, mas versões posteriores do algoritmo sacrificaram os princípios biológicos em prol de um melhor desempenho (CORTÉS; COELLO, 2003a; CORTÉS; COELLO, 2003b; COELLO; CORTÉS, 2005b). No algoritmo MISA a população é codificada por cadeias binárias e é inicializada aleatoriamente. A escolha dos anticorpos a serem clonados é feita por algumas regras. Os critérios utilizados para esta escolha são: preferencialmente indivíduos factíveis e não-dominados, em segundo lugar os indivíduos não-dominados e infactíveis e a última opção corresponde à seleção de indivíduos infactíveis e dominados. O conjunto de células de memória é atualizado com indivíduos não-dominados factíveis. Este repositório possui tamanho limitado, por isso é implementada uma grade para forçar uma distribuição uniforme dos pontos. O número de clones depende da afinidade antígeno-anticorpo. Estes anticorpos clonados são selecionados para uma mutação uniforme com probabilidade proporcional à sua afinidade de acordo com as regras descritas anteriormente para a escolha dos anticorpos a serem clonados. Os anticorpos que não foram selecionados para a mutação uniforme sofrem uma mutação não-uniforme. Por fim, a população é reduzida ao seu tamanho original fazendo uso da seleção dos indivíduos através das regras previamente descritas.

O algoritmo MOCSA (Multiobjective Clonal Selection Algorithm) é outro sistema imunológico artificial multiobjetivo baseado no princípio da seleção clonal (CAMPELO et al., 2004). No MOCSA a população é codificada por valores reais e os anticorpos são

avaliados e *rankeados* de acordo com a relação de dominância de Pareto. Então, clones são gerados para toda a população sendo que o número de clones para cada anticorpo é inversamente proporcional a sua posição no *rank*. O espaço de busca é explorado localmente através da aplicação de uma perturbação aleatória aos clones com desvio padrão fixo. A população intermediária formada pelos anticorpos originais e pelos clones mutados é *rankeada* e armazenada na memória secundária. Na sequência a supressão é aplicada no espaço objetivo. A cada iteração um percentual de anticorpos selecionados aleatoriamente é gerado para substituir os anticorpos que possuem os piores *ranks*. Este procedimento garante a manutenção da diversidade da população.

Uma versão da rede opt-aiNet para tratar problemas multiobjetivo, chamada VIS (Vector Immune System), foi proposta em (FRESCHI; REPETTO, 2005; FRESCHI; REPETTO, 2006). Neste algoritmo, os passos iniciais são baseados no princípio da seleção clonal e são repetidos um certo número de vezes antes das interações de avaliação da rede de anticorpos. A afinidade dos anticorpos é dada pelo número de anticorpos que o dominam. O VIS trabalha com representação real das variáveis e seu operador de hipermutação corresponde a uma perturbação com desvio padrão que depende do valor da afinidade e é adaptado em um esquema semelhante ao presente nas estratégias evolutivas. O mecanismo de seleção não considera toda a população: a cada iteração o melhor anticorpo entre o pai e seus clones é selecionado. Assim, uma solução dominada tem oportunidade de evoluir, o que pode prevenir a convergência prematura do algoritmo. A seguir as células de memória são atualizadas com as soluções não-dominadas encontradas até o momento. Em busca de diversidade da população uma forma de supressão com um limiar de proximidade adaptativo elimina os anticorpos com maior afinidade entre si. A cada iteração um número fixo de anticorpos gerados aleatoriamente são inseridos na população. Além disso, os autores propõem uma forma de lidar com as restrições para manter a factibilidade das soluções durante a mutação. Para isto, sempre que um anticorpo infactível é gerado, a intensidade da mutação é reduzida pela metade e ela é reaplicada ao anticorpo original. Este processo é repetido até que o anticorpo gerado pela mutação seja factível. Este método tende a explorar mais a fronteira entre as regiões de factibilidade e infactibilidade.

Em Wang e Mahfouf (2006) os autores propõem um algoritmo adaptativo baseado no princípio da seleção clonal para otimização multiobjetivo. O algoritmo é denominado ACSAMO (*Adaptive Clonal Selection Algorithm for Multiobjective Optimization*) e mantém como antígenos a melhor solução de todo o processo evolutivo e a melhor solução da iteração anterior. Para identificar essas soluções é utilizado um vetor dinâmico de escalarização (modificado a cada iteração). A afinidade de um anticorpo é definida como a soma das distâncias Euclidianas entre ele e os antígenos medidas no espaço de busca. A afinidade também é utilizada para guiar a amplitude da mutação, quanto melhor é o

indivíduo menor é a mutação que ele sofre. A seleção das células para a próxima geração é baseada na ordenação baseada em dominância e com a supressão dos anticorpos (caso exista um excesso de anticorpos) baseada em uma técnica de *crowding*. O conjunto de células de memória é composto pelos anticorpos não-dominados.

O algoritmo CNMOIA (*Constrained Nonlinear Multiobjective Optimization Immune Algorithm*) foi proposto por Zhang et al. (2006), Zhang (2007) e computa dois indicadores para cada anticorpo: a afinidade antígeno-anticorpo e a densidade do anticorpo. O primeiro valor mede a proximidade de soluções não-dominadas por meio da definição do produto interno e está vinculado ao mecanismo de controle de violações das restrições. O segundo valor conta o número de soluções na proximidade do anticorpo (*crowding*). Estes valores são utilizados para medir o nível de estímulo de cada anticorpo. O número de clones para cada anticorpo e sua probabilidade de mutação são baseados na sua afinidade. A seleção para a próxima geração ocorre de acordo com o nível de estímulo dos anticorpos. Neste algoritmo as células de memória são formadas pelos antígenos.

Em Gao e Wang (2010) o algoritmo WBMOAIS (*Weight-based Multiobjective Artificial Immune System*) é apresentado. Assim como o VIS, este algoritmo é baseado na rede imune artificial opt-aiNet. Porém, a afinidade dos anticorpos no WBMOAIS é calculada através de uma função aleatória para gerar pesos para cada um dos objetivos (soma ponderada dos objetivos). Os anticorpos são codificados por valores reais e a população inicial é gerada aleatoriamente. Os anticorpos são clonados um número fixo de vezes e na sequência é aplicada uma perturbação aleatória com média zero e desvio padrão igual a um. Após a mutação os clones passam por um processo de busca local guiada pelos pesos do vetor utilizado na avaliação do anticorpo. Anticorpos não-dominados são armazenados na memória do algoritmo. Caso o número de anticorpos que formam a memória seja maior que o tamanho da população (Tam_{pop}), são selecionados aleatoriamente Tam_{pop} anticorpos da memória para compor a população da próxima geração. Caso contrário, todos os elementos da memória são copiados para a próxima população e o restante da população é formada por anticorpos dominados selecionados aleatoriamente entre os clones e os elementos da memória. Caso a memória atinja a sua capacidade máxima, um procedimento baseado na distância entre os anticorpos é aplicado para a eliminação dos anticorpos excedentes. Além disto, existe uma probabilidade de novos indivíduos gerados aleatoriamente serem inseridos na população para manter a sua diversidade.

3.4.4.2 Algoritmos Culturais Multiobjetivo

Os algoritmos evolucionários multiobjetivo, assim como os mono-objetivo, podem se beneficiar da incorporação de conhecimentos tanto a priori quanto durante o processo

evolutivo. Segundo Becerra, Santana-Quintero e Coello (2008) existem basicamente dois locais onde os conhecimentos podem ser incorporados em AEs multiobjetivo: na avaliação da função de *fitness* e nos operadores dos algoritmos. A primeira forma é útil para lidar com avaliações custosas (como por exemplo, avaliações que exijam simulações de sistemas complexos) e não será tratada neste trabalho. Já a segunda é amplamente aplicável e pode ser explorada, por exemplo, através do uso de algoritmos culturais multiobjetivo (BECERRA; SANTANA-QUINTERO; COELLO, 2008).

Apesar dos algoritmos culturais terem sido propostos na década de 90 (REYNOLDS, 1994) apenas recentemente versões multiobjetivo têm sido mas exploradas. A primeira abordagem cultural multiobjetivo foi o ϵ CCDE (BECERRA; COELLO, 2006c; BECERRA, 2007). Esta proposta utilizava um espaço populacional baseado em evolução diferencial os conhecimentos situacional, normativo, histórico e topográfico, fazia uso do conceito de ϵ – *constraints* (BECERRA, 2007). Na verdade o algoritmo dividia o problema multiobjetivo em vários problemas mono-objetivo usando o conceito de ϵ – *constraints* e resolvia estes problemas utilizando um algoritmo cultural com evolução diferencial. Os resultados obtidos foram considerados bons, mas o custo computacional do algoritmo era considerado muito alto. Por isto, em (BECERRA et al., 2007) um procedimento auxiliar baseado em conjuntos aproximados foi incorporado para reduzir o custo computacional e o algoritmo resultante foi chamado de ϵ CCDE + MORS.

Em Gao et al. (2007) foi proposto o CPSA, que consiste de um algoritmo cultural multiobjetivo com espaço populacional baseado em nuvem de partículas. Na verdade existem $n + 1$ subpopulações, uma para cada objetivo e uma última que a cada k_0 gerações sorteia o objetivo que irá otimizar. Neste sentido o algoritmo lembra o VEGA (SCHAFFER, 1985). A função de aceitação utilizada é estática e aceita 20% dos melhores indivíduos de cada população. Apenas o conhecimento situacional é utilizado e ele se dá na introdução de um termo na fórmula de atualização da velocidade que utiliza um dos melhores indivíduos armazenados no conhecimento situacional. O algoritmo utiliza um arquivo externo para armazenar as soluções não-dominadas.

Uma proposta que utiliza algoritmos culturais para evoluir os parâmetros de um exame de partículas multiobjetivo é proposto em Daneshyari e Yen (2008). Ele foi denominado Cultural MOPSO e utiliza três tipos de conhecimento: situacional, normativo e topográfico. O conhecimento situacional é representado por uma grade adaptativa associada a cada partícula onde as soluções não-dominadas encontradas por aquela partícula são armazenadas, o conhecimento normativo armazena os extremos de cada objetivo e a velocidade da partícula que encontrou cada um dos extremos e o conhecimento topográfico armazena as soluções não-dominadas utilizando uma grade adaptativa. O conhecimento situacional é utilizado para atualizar o termo cognitivo da atualização da velocidade e para

escolher o melhor local, o conhecimento topográfico é utilizado para atualizar o termo social da atualização da velocidade e para a escolha do melhor global e o conhecimento normativo é utilizado para atualizar a constante de inércia.

O algoritmo MOCDE, um algoritmo cultural com espaço populacional baseado em evolução diferencial, é proposto em Qin et al. (2010). O MOCDE utiliza os conhecimentos situacional, normativo e histórico. Durante a influência do conhecimento situacional apenas indivíduos armazenados no conhecimento situacional são utilizados no operador de mutação. Já durante a influência do conhecimento normativo, se um indivíduo possui um valor fora dos extremos do intervalo do conhecimento normativo aquele valor é modificado para o extremo mais próximo deste intervalo. O conhecimento histórico mede a diferença entre uma métrica de convergência entre a população atual e uma população anterior (esta métrica exige a fronteira de Pareto real ou uma boa aproximação para a fronteira de Pareto) e também mede a diversidade da população. Se ambas forem baixas uma mutação utilizando a distribuição de Cauchy é aplicada. O método de seleção é alterado para privilegiar indivíduos não-dominados.

4 *Problema do Despacho de Energia Elétrica*

Este capítulo descreve brevemente os problemas do Despacho Econômico de Energia e do Despacho Econômico/Ambiental de Energia.

4.1 Despacho Econômico de Energia Elétrica

O Despacho Econômico é um dos mais importantes problemas a ser resolvido durante o planejamento e a operacionalização de um sistema de geração de energia elétrica (PARK; JEONG; LEE, 2006). O objetivo do despacho econômico é minimizar o custo total de combustível de geradores de energia elétrica sujeito a restrições operacionais. A função objetivo desse problema pode ser formulada como:

$$\begin{aligned} &\text{Minimizar } F = \sum_{j=1}^n F_j(P_j) \\ &\text{sujeito a} \\ &PD = \sum_{j=1}^n P_j \text{ e} \\ &P_j^{\min} \leq P_j \leq P_j^{\max}, j = 1, \dots, n \end{aligned} \tag{4.1}$$

onde $F_j(P_j)$ é a função que representa o custo do j th gerador (em \$/h), P_j é a potência de saída do j -ésimo gerador, n é o número de geradores no sistema, PD é a demanda total do sistema, P_j^{\min} e P_j^{\max} são, respectivamente, as saídas mínimas e máximas do j -ésimo gerador. A restrição de igualdade da Equação 4.1 é chamada de restrição de balanceamento de potência enquanto que as restrições de desigualdade são chamadas de restrições operacionais.

A função de custo de uma unidade geradora, sem considerar os efeitos de ponto de válvula, é dada por uma aproximação quadrática (BALAMURUGAN; SUBRAMANIAN, 2007):

$$F_j(P_j) = a_j * P_j^2 + b_j * P_j + c_j \tag{4.2}$$

onde P_j é a saída do j -ésimo gerador e a_j , b_j e c_j são coeficientes que determinam o custo de combustível do j -ésimo gerador.

Mas geradores com turbinas com múltiplas válvulas exibem uma variação muito maior das funções de custo devido ao efeito de ponto de válvula (PARK; JEONG; LEE, 2006). O efeito de ponto de válvula resulta em ondulações na curva de custo de combustível de um gerador provocada pela abertura de válvulas. A função de custo considerando o efeito das válvulas resulta num grau de não-linearidade maior (ver Figuras 21 e 22). Então a Equação 4.2 deve ser substituída pela Equação 4.3, que considera os efeitos das válvulas. Matematicamente tem-se uma combinação de uma função quadrática e uma função senoidal:

$$F_j(P_j) = a_j * P_j^2 + b_j * P_j + c_j + |e_j * \sin(f_j * (P_j^{min} - P_j))| \quad (4.3)$$

onde P_j é a saída da j -ésima unidade geradora, a_j , b_j , c_j , e_j and f_j são os coeficientes da função de custo do j -ésimo gerador quando o efeito de ponto de válvula é considerado.

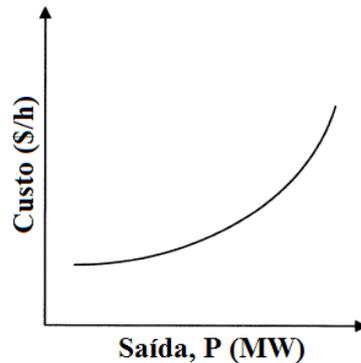


Figura 21: Custo de um gerador sem efeito de ponto de válvula.

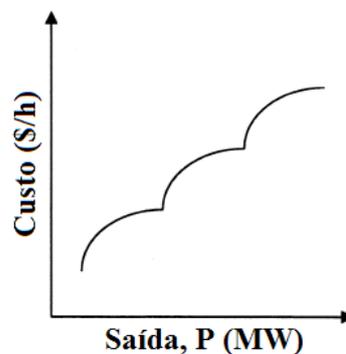


Figura 22: Custo de um gerador com efeito de ponto de válvula.

4.1.1 Problema do Despacho de Energia considerando Perdas

A restrição de balanceamento de potência reduz o sistema de potência a um princípio de equilíbrio entre a quantidade total de energia gerada pelo sistema e a quantidade total de energia demandada ao sistema (PEREZ-GUERRERO; CEDENIO-MALDONADO, 2005).

Quando a demanda e os geradores encontram-se em uma região geográfica pequena as perdas na transmissão da energia podem ser desconsideradas (WOOD; WOLLENBERG, 1996). No problema do despacho de energia que não considera perda (conforme descrito anteriormente) este equilíbrio é alcançado quando a quantidade de energia gerada ($\sum_{j=1}^n P_j$) é igual a demanda do sistema (PD), conforme a restrição de igualdade da Equação 4.1.

Entretanto, em um sistema com grandes distâncias entre os geradores e a demanda, as perdas na transmissão da energia tornam-se significativas e devem ser consideradas (WOOD; WOLLENBERG, 1996). Então, o problema do despacho de energia que considera as perdas na geração de energia elétrica alcança o equilíbrio quando a quantidade de energia gerada ($\sum_{j=1}^n P_j$) é igual a demanda do sistema (PD) mais as perdas do sistema (PL), conforme a Equação 4.4.

$$\sum_{j=1}^n P_j = PD + PL \quad (4.4)$$

As perdas do sistema podem ser determinadas exatamente como um resultado do problema do fluxo de energia. Existem diferentes maneiras para o cálculo das perdas na transmissão de energia. Pode-se calcular a energia fazendo uso de fatores de penalidade, considerando as perdas como uma constante ou então fazendo uso da fórmula de Kron (WOOD; WOLLENBERG, 1996).

Resolvendo o problema do fluxo de energia a perda pode ser calculada como (KING; RUGHOPUTH; DEB, 2005):

$$PL = \sum_{i=1}^N \sum_{j=1}^N [(r_{ij}/V_i V_j) \cos(\delta_i - \delta_j) (Pr_i Pr_j + Qi Q_j) + (r_{ij}/V_i V_j) \sin(\delta_i - \delta_j) (Q_i Pr_j - Pr_i Q_j)] \quad (4.5)$$

onde N é o número de barramentos, r_{ij} é a resistência em série conectando os barramentos i e j , V_i é a magnitude da voltagem no barramento i , δ_{ij} é o ângulo no barramento i , Pr_i é a potência real injetada no barramento i e Q_i é a potência reativa injetada no barramento i .

Já a fórmula de Kron expressa as perdas do sistema como uma função quadrática da potência de saída dos geradores e pode ser aproximada conforme a Equação 4.6.

$$PL = \sum_{i=1}^n \sum_{j=1}^n P_i * B_{ij} * P_j + \sum_{i=1}^n P_j * B_{i0} + B_{00} \quad (4.6)$$

onde o coeficiente B_{ij} é o j -ésimo elemento da i -ésima linha da matriz de perda, ou seja, a contribuição da relação entre os geradores i e j para a perda, B_{i0} é o i -ésimo elemento do vetor de coeficientes de perda, ou seja, a perda inerente ao gerador i , e B_{00} é o coeficiente constante de perda, ou seja, uma perda constante do sistema gerador de energia elétrica. Estes coeficientes são chamados de coeficientes de perda ou coeficientes B (COELHO; LEE, 2008). Assume-se, com um pequeno erro, que estes coeficientes são constantes desde que a operação esteja próxima do valor onde estes coeficientes foram computados.

Neste trabalho as perdas na transmissão de energia são calculadas fazendo uso da fórmula de Kron (Equação 4.6).

4.2 Problema do Despacho Econômico/Ambiental de Energia

O Despacho Econômico/Ambiental de energia visa otimizar tanto o custo de produção da energia gerada quanto a emissão de poluentes decorrentes da geração, atendendo às restrições do problema.

Sob o aspecto matemático o problema pode ser definido pela Equação 4.7 (CAIA et al., 2009).

$$\begin{aligned} &\text{Minimizar [Custo, Emissão]} \\ &\text{sujeito a} \\ &PD = \sum_{j=1}^n P_j - P_L \text{ e} \\ &P_j^{min} \leq P_j \leq P_j^{max} \end{aligned} \quad (4.7)$$

onde P_j é a potência de saída do j -ésimo gerador, n é o número de geradores no sistema, PD é a demanda total do sistema, P_L são as perdas decorrentes da transmissão, P_j^{min} e P_j^{max} são, respectivamente, as saídas mínimas e máximas do j -ésimo gerador. As perdas para o problema do despacho econômico/ambiental de energia são calculadas da mesma forma que para o problema do despacho econômico de energia com perdas (ver

Equação 4.6). Para instâncias do problema onde as perdas não são consideradas o valor de P_L utilizado é zero.

O custo e a emissão, os quais devem ser minimizados, são definidos nas Equações 4.8 e 4.9.

$$Custo = \sum_{j=1}^n F_j(P_j) = \sum_{j=1}^n a_j * P_j^2 + b_j * P_j + c_j \quad (4.8)$$

$$Emissão = \sum_{j=1}^n E_j(P_j) = \sum_{j=1}^n \alpha_j * P_j^2 + \beta_j * P_j + \gamma_j + \zeta_j * \exp(\lambda_j * P_j) \quad (4.9)$$

onde a_j , b_j e c_j são coeficientes do custo de geração de energia no j -ésimo gerador e α_j , β_j , γ_j , ζ_j e λ_j são coeficientes de emissão de poluentes do respectivo gerador.

4.3 Abordagens para os Problemas de Despacho de Energia

A seguir são descritas algumas das abordagens presentes na literatura para solucionar o problema do despacho econômico e econômico/ambiental de energia.

4.3.1 Abordagens para o Problema do Despacho Econômico de Energia

O método λ -iteration é aplicado em (WOOD; WOLLENBERG, 1996) para problemas de despacho de energia.

Em Su e Lin (2000) uma rede neural do tipo Hopfield é utilizada para resolver o problema do despacho econômico de energia.

O IGA (LING et al., 2003) é um algoritmo genético com novos operadores de *crossover* e mutação para a solução do problema do despacho econômico de energia.

Propostas baseadas em programação evolutiva e mutações Gaussianas e de Cauchy são apresentadas em Chakrabarti e Chattopadhyay (2003).

Em Victoire e Jeyakumar (2004) são propostos um algoritmo baseado em enxame de partículas (PSO) e outro que combina enxame de partículas com programação quadrática sequencial (PSO-SQP) para a resolução do despacho econômico de energia.

O CEP-PSO (SINHA; PURKAYASTHA, 2004) combina programação evolutiva e enxame de partículas para resolver o problema do despacho econômico.

Em Khamsawang, Pothiya e Boonseng (2004) uma busca tabu dinâmica (DTSA) é aplicada ao problema do despacho econômico de energia.

Um algoritmo básico de evolução diferencial para o problema do despacho econômico (DEvol) é proposto em Perez-Guerrero e Cedenio-Maldonado (2005).

O MPSO (Modified Particle Swarm Optimization) resolve o problema do despacho econômico de energia utilizando um enxame de partículas com uma estratégia dinâmica de redução do espaço de busca e um mecanismo para lidar com as restrições de igualdade e desigualdade (HOU et al., 2005)

Um algoritmo que combina evolução diferencial, sequências caótica e uma programação quadrática sequencial, denominado DEC-SQP, é proposto em Coelho e Mariani (2006).

O HDE é um algoritmo baseado em evolução diferencial e que possui um mecanismo de controle de diversidade e aceleração de convergência. O STHDE é uma versão auto-adaptativa do HDE onde o fator de escala é auto-adaptado para cada indivíduo (WANG; CHIOU; LIU, 2007).

A idéia do NPSO-LRS é introduzir o conceito de busca local aleatória em um algoritmo de enxame de partículas que faz uso de conceitos de melhores e piores indivíduos e não somente de melhores indivíduos como ocorre classicamente em algoritmos de enxame de partículas (SELVAKUMAR; THANUSHKODI, 2007).

O SDE (BALAMURUGAN; SUBRAMANIAN, 2007) é um algoritmo de evolução diferencial com mecanismo de auto-adaptação da taxa de *crossover* (CR) e do fator de escala (F).

Um algoritmo inspirado em enxames de partículas, teorias quânticas e oscilações harmônicas chamado HQPSO é proposto em Coelho e Mariani (2008).

Em Noman e Iba (2008) é utilizada uma evolução diferencial para resolver vários problemas de despacho econômico de energia. São propostos mecanismos de reparação de indivíduos para os casos com perdas e sem perdas de energia.

O CDEMD é uma abordagem que mistura evolução diferencial e algoritmos culturais utilizando conhecimentos situacional e normativo para resolver o problema do despacho econômico (COELHO; SOUZA; MARIANI, 2009). O CDEMD também utiliza um mecanismo para controlar a diversidade da população.

O IHS (COELHO; MARIANI, 2009) é uma versão do algoritmo Harmony Search (GEEM, 2009) onde uma estratégia de adaptação da taxa de *pitch* e uma distribuição exponencial são utilizados para resolver problemas de despacho econômico de energia.

Um enxame de partículas com ajuste automático dos coeficientes de aceleração de acordo com a geração atual (PSO-TVAC) é proposto em Chaturvedi, Pandit e Srivastava (2009).

Em Vlachogiannis e Lee (2009) um algoritmo de enxame de partículas onde as partículas são atraídas por outras partículas melhores do que elas, com exceção da melhor partícula que se move de maneira aleatória, é apresentado para o problema do despacho econômico de energia. Esta abordagem é chamada de ICA-PSO.

O CSO (Civilized Swarm Optimization) combina conceitos do algoritmo de sociedade civil (RAY; LIEW, 2003) e de enxame de partículas, criando um híbrido onde os conceitos de sociedade são utilizados para modificar os operadores de enxame de partículas (SELVAKUMAR; THANUSHKODI, 2009).

Em Bhattacharya e Chattopadhyay (2010) um método baseado em conceitos biogeográficos é aplicado para resolução do problema do despacho econômico. Os dois principais operadores da técnica são a migração e a mutação. Além disto, é proposta uma forma de reparação para problemas com perdas de energia.

O QPSO é proposto em Meng et al. (2010). Ele é um enxame de partículas que utiliza conceitos de computação quântica e mutações baseadas em sequências caóticas. Conceitos de afinidade e concentração das partículas são utilizados na seleção.

4.3.2 Abordagens para o Problema do Despacho Econômico/Ambiental de Energia

Em Farag, Al-Baiyat e Cheng (1995) um método de programação linear (LP) é utilizado para tratar o caso do despacho econômico/ambiental com seis geradores.

Os mesmos casos são tratados em Das e Patvardhan (1998) onde o MOSST foi proposto. Ele é um algoritmo evolucionário multiobjetivo que mistura conceitos de algoritmos genéticos e *simulated annealing*. Ele faz uso de uma memória externa para armazenar as soluções e foi aplicado em um caso com 6 geradores.

Uma versão do algoritmo NSGA (Nondominated Sorting Genetic Algorithm) é utilizada em Abido (2003b) na solução do problema do despacho econômico/ambiental de

energia para instâncias com 6 geradores. O autor gera conjuntos de aproximação da fronteira de Pareto para uma instância no padrão IEEE e na sequência extrai a melhor solução compromisso destes conjuntos através de um mecanismo *fuzzy*.

No trabalho de Abido (2003a) o algoritmo NPGA (Niche Pareto Genetic Algorithm) é aplicado na solução do problema do despacho econômico/ambiental de energia para uma instância do problema com 6 geradores. A mesma técnica adotada em Abido (2003b) é utilizada para escolher a melhor solução compromisso.

Em Abido (2003) o algoritmo SPEA é utilizado na resolução de problemas do despacho econômico/ambiental para instâncias com 6 geradores.

Em Venkatesh, Gnanadass e Padhy (2003) técnicas de programação evolutiva e de escalarização são utilizadas para resolver um caso do problema do despacho econômico/ambiental de energia com 14 geradores.

Em Guerrero (2004) duas técnicas baseadas em evolução diferencial são propostas. A primeira (ECED) transforma o problema multiobjetivo em um problema mono-objetivo onde a emissão de poluentes é transformada em restrição. Já a segunda abordagem resolve o problema na sua formulação multiobjetivo (MEED). Foram tratadas instâncias com 6 e 14 geradores.

O NSGA-II é um dos mais utilizados algoritmos evolucionários multiobjetivo, em boa parte por causa do seu bom desempenho. Em King, Rughooputh e Deb (2005) ele é utilizado para resolver instâncias com 6 geradores do problema do despacho econômico/ambiental.

Em Palanichamy e Babu (2008) um método analítico (ADS) é proposto para resolver o problema do despacho econômico/ambiental. Cabe salientar que o método somente funciona se tanto a função custo quanto de emissão de poluentes forem diferenciáveis.

O MOPSO (ABIDO, 2009) é uma versão multiobjetivo do algoritmo de enxame de partículas (PSO). As principais diferenças para o algoritmo mono-objetivo residem nas escolhas dos ótimos locais e globais e no arquivamento das soluções não dominadas. Para cada partícula (solução) existe um conjunto de soluções não-dominadas que foram geradas por aquela partícula ao longo do processo evolutivo. O ótimo local da partícula é a solução não-dominada contida neste conjunto cuja distância Euclidiana é mínima com relação à partícula atual. Este conjunto possui um tamanho máximo e um algoritmo de *clusterização* (semelhante ao do SPEA) é utilizado para reduzir o tamanho do conjunto sempre que necessário. De maneira análoga existe um conjunto de soluções não-dominadas globais, que consiste da união dos conjuntos de soluções não-dominadas locais de todas as

partículas da população corrente. Este conjunto também possui um tamanho máximo e o mesmo procedimento adotado para reduzir o tamanho dos conjuntos locais é adotado neste conjunto. Um conjunto externo é mantido com as soluções não-dominadas encontradas ao longo do processo evolucionário. Este arquivo também possui um tamanho máximo e utiliza o método de *clusterização* para reduzir o seu tamanho sempre que necessário.

O MOCPSO (CAIA et al., 2009) é um enxame de partículas multiobjetivo com busca local caótica, valores de inércia adaptativos e que faz uso do compartilhamento de *fitness* e de uma grade adaptativa para armazenar os melhores indivíduos.

O PCEED (CETINKAYA, 2009) inicialmente resolve separadamente o despacho econômico e o despacho ambiental, depois utilizando diferentes vetores de escalarização encontra uma possível fronteira de Pareto para o problema do despacho econômico/ambiental. O PCEED foi testado em uma instância com 6 geradores.

5 Algoritmos Culturais para o Problema do Despacho de Energia

Conforme destacado no Capítulo 1, a proposta apresentada nesta tese tem por base os seguintes trabalhos: Almeida, Gonçalves e Delgado (2007), Gonçalves et al. (2007), Gonçalves et al. (2008), Kuk (2009) e Gonçalves et al. (2009). Assim, o algoritmo imune apresentado em Almeida, Gonçalves e Delgado (2007) representa uma versão preliminar do algoritmo imune (IS) proposto. Os algoritmos imuno-culturais propostos em Gonçalves et al. (2007), Gonçalves et al. (2008), Kuk (2009) e Gonçalves et al. (2009) também representam versões anteriores as quais foram aprimoradas nesta tese.

Assim, este capítulo apresenta a descrição do algoritmo imune e dos algoritmos culturais como estão sendo propostos neste trabalho.

5.1 O Sistema Imune Proposto

Conforme já destacado no Capítulo 2, os algoritmos Imunes são algoritmos estocásticos inspirados na imunologia, funções imunes e princípios observados na natureza (CUTELLO et al., 2007). Neste trabalho há um interesse particular em algoritmos imunes baseados no princípio da seleção clonal, como o Clonalg (DE CASTRO; VON ZUBEN, 2002) e o opt-IMMALG (CUTELLO et al., 2006) os quais podem ser classificados como algoritmos evolucionários (CORTÉS; COELLO, 2005). Este interesse se justifica pela prova recente de que esta classe de algoritmos (sob certas condições) é capaz de encontrar o ótimo global de um problema de otimização (CUTELLO et al., 2007).

No sistema imune desenvolvido neste trabalho para resolver o problema de despacho econômico, cada anticorpo (ou célula B) é uma combinação válida das potências de saída de cada unidade geradora (codificadas como valores reais). A afinidade de um anticorpo

com o antígeno é dada por:

$$aff_i = (MaxCost_{gen} - Cost_i) / (MaxCost_{gen} - MinCost_{gen}) \quad (5.1)$$

onde $Cost_i$ é o custo total de combustível de uma determinada configuração (representada pelo i -ésimo anticorpo), $MinCost_{gen}$ e $MaxCost_{gen}$ são custo máximo e mínimo de combustível de um anticorpo na geração gen , respectivamente. Esta função associa maiores valores de afinidade a anticorpos com menor custo e ao mesmo tempo limita a afinidade no intervalo $[0,1]$.

Cada anticorpo é composto por n componentes (potência dos geradores) que são inicializadas na primeira geração segundo a Equação 5.2.

$$ant_{i,j} = P_j^{min} + U(0,1) * (P_j^{max} - P_j^{min}) \quad (5.2)$$

onde $ant_{i,j}$ é o j -ésimo componente do i -ésimo anticorpo (isto é, a potência de saída da j -ésima, $j = 1, \dots, n$, unidade geradora), $U(0,1)$ é uma variável aleatória amostrada por uma distribuição uniforme no intervalo $[0,1]$, P_j^{min} e P_j^{max} são, respectivamente, as potências de saída mínima e máxima da j -ésima unidade. Este procedimento de inicialização garante que nenhum anticorpo viole as restrições operacionais das unidades geradoras.

Os antígenos representam a função a ser otimizada e neste caso, cada anticorpo (solução do problema de otimização) possui uma afinidade diferente representando o seu grau de adequação ao problema.

Os passos do sistema imune implementado são apresentados no Algoritmo 3.

Algoritmo 3 Algoritmo imunológico proposto

1. Inicialize a População de Anticorpos;
 2. Avalie a População;
 - $gen \leftarrow 0$;
 3. Enquanto o Critério de Parada não for Satisfeito
 - 3.1. Clone a População;
 - 3.2. Aplique as Hipermutações nos Clones
 - 3.3. Aplique o Quase-Simplex;
 - 3.4. Aplique o Procedimento de Reparo;
 - 3.5. Avalie os Clones;
 - 3.6. Aplique o Operador de Aging na População e nos Clones;
 - 3.7. Aplique o Operador de Seleção ($\mu + \lambda$);
 - 3.8. $gen = gen + 1$;
- End While
-

Após a avaliação da população inicial, o Sistema Imune entra no seu laço principal (o qual representa o estágio de maturação da afinidade no algoritmo). Esse laço é repetido até que o critério de parada seja satisfeito. Nesse trabalho o critério de parada adotado foi um número máximo de gerações.

O processo de maturação da afinidade começa com a clonagem dos anticorpos da geração passada. Os operadores de clonagem mais comuns são: operador de clonagem estático (onde o número de clones de cada anticorpo é *dup*, independentemente de sua afinidade) e o operador de clonagem proporcional (onde o número de clones de um anticorpo é proporcional à sua afinidade) (CUTELLO et al., 2007). Neste trabalho, por simplicidade, é utilizado o operador de clonagem estático.

Na sequência, os operadores de hipermutação são aplicados. O algoritmo implementado faz uso de dois operadores de hipermutação adaptativos: Hipermutação Gaussiana Adaptativa (HGA) e Hipermutação de Cauchy Adaptativa (HCA). Os operadores de hipermutação propostos são ditos adaptativos porque a intensidade dos mesmo se adapta à afinidade do anticorpo. Estes operadores são dados pela Equação 5.3.

$$ant_{i,j} = ant_{i,j}^c + mult * (Cost_i / MinCost_{gen}) * R(0,1) * (P_j^{max} - P_j^{min}) \quad (5.3)$$

onde $ant_{i,j}^c$ é o j-ésimo componente do i-ésimo anticorpo após o processo de clonagem, $R(0,1)$ é um valor aleatório amostrado de uma distribuição Gaussiana ou de Cauchy dependendo do tipo de hipermutação aplicado (HGA ou HCA), em ambos os casos a distribuição possui média 0 e desvio padrão 1 e não é truncada. O termo *mult* representa um multiplicador adaptativo dado por:

$$mult = e^{-gen/\varphi} \quad (5.4)$$

onde φ é a velocidade de decréscimo do multiplicador. Os demais termos foram previamente definidos (Equações 5.1 e 5.2).

O termo $Cost_i / MinCost_{gen}$ na Equação 5.3 faz com que a mutação seja mais intensa em anticorpos com alto custo (baixa afinidade) e suave em anticorpos que possuem baixo custo (alta afinidade). No processo de hipermutação, os dois tipos (HGA ou HCA) são aplicados e o melhor anticorpo produzido é escolhido. De acordo com Cortés e Coello (2005), números aleatórios gerados de acordo com uma distribuição de Cauchy permitem uma maior exploração do espaço de busca (uma parte maior do espaço de busca pode ser atingida pela aplicação do operador) enquanto números aleatórios gerados de acordo com uma distribuição Gaussiana permitem uma maior intensificação da busca (o operador gera

valores na proximidade dos valores originais do anticorpo), o que em teoria deixa a busca balanceada. O multiplicador *mult*, definido na Equação 5.4 tenta fazer com que a busca seja intensificada no início da evolução e suavizada no final do processo evolutivo.

O número de mutações aplicado a cada anticorpo é dado pela Equação 5.5.

$$M(i) = \frac{e^{-aff_i}}{\rho} * n \quad (5.5)$$

onde $M(i)$ é o número de mutações aplicadas ao i -ésimo anticorpo, aff_i é a afinidade do i -ésimo anticorpo dada pela equação 5.19, ρ é um parâmetro que regula o número de mutações e n é a quantidade de geradores considerada.

Após a hipermutação um método quase-Simplex (DONG et al., 2008) é aplicado aos melhores anticorpos como um procedimento de busca local. O método quase-simplex é derivado do método simplex (SPENDLEY; HEXT; HIMSWORTH, 1962) e é implementado neste trabalho como descrito a seguir.

Primeiramente, os $n + 1$ melhores anticorpos da população são escolhidos (lembrando que n representa a dimensão da instância do problema de despacho de energia sendo resolvida, ou seja, o número de geradores). Então, o centróide X_c dos anticorpos é calculado pela Equação 5.6.

$$X_{c,j} = \left(\sum_{k=1}^{n+1} ant_{k,j} \right) / (n + 1) \quad (5.6)$$

onde $X_{c,j}$ representa o j -ésimo componente do centróide X_c e $ant_{k,j}$ o j -ésimo componente do k -ésimo anticorpo A_k selecionado.

A seguir algumas transformações são aplicadas ao conjunto de anticorpos tendo como base o centróide X_c calculado pela Equação 5.6, o pior anticorpo (A_p) e o melhor anticorpo (A_m) do conjunto de $(n + 1)$ anticorpos selecionados. Estas transformações têm como objetivo encontrar novos anticorpos melhores (com menor custo) do que os originais. Os principais passos destas transformações são adaptados de (DONG et al., 2008) e descritos a seguir.

- **Reflexão:** A reflexão do pior anticorpo em relação ao centróide é dada por

$$X_{p,j}^r = ant_{p,j} + (ant_{p,j} - X_{c,j}), \quad j = 1, \dots, n$$

onde $X_{p,j}^r$ representa o j -ésimo componente da reflexão X_p^r , do pior anticorpo A_p .

De forma similar, a reflexão do melhor anticorpo em relação ao centróide é dada por

$$X_{m,j}^r = ant_{m,j} + (X_{c,j} - ant_{m,j}), j = 1, \dots, n$$

com $X_{m,j}^r$ representando o j -ésimo componente da reflexão X_m^r , do melhor anticorpo A_m .

- **Expansão:** A expansão do pior anticorpo além do centróide é dada por

$$X_{p,j}^e = ant_{p,j} + (1 + \alpha)(ant_{p,j} - X_{c,j}), j = 1, \dots, n$$

onde $X_{p,j}^e$ representa o j -ésimo componente da expansão X_p^e , do pior anticorpo A_p , e α é um número aleatório sorteado no intervalo $[0, 1]$ para cada transformação.

A expansão do melhor anticorpo além do centróide é dada por

$$X_{m,j}^e = ant_{m,j} + (1 + \alpha)(X_{c,j} - ant_{m,j}), j = 1, \dots, n$$

ou

$$X_{m,j}^e = ant_{m,j} + \alpha(X_{c,j} - ant_{m,j}), j = 1, \dots, n$$

com $X_{m,j}^e$ representando o j -ésimo componente da expansão X_m^e , do melhor anticorpo A_m .

- **Compressão:** A compressão gera pontos entre o pior anticorpo e o centróide e o melhor anticorpo e o centróide. A compressão do pior anticorpo em relação ao centróide é dada por

$$X_{p,j}^c = ant_{p,j} - \alpha(ant_{p,j} - X_{c,j}), j = 1, \dots, n$$

onde $X_{p,j}^c$ representa o j -ésimo componente da compressão X_p^c , do pior anticorpo A_p .

A compressão do melhor anticorpo em relação ao centróide é dada por

$$X_{m,j}^c = ant_{m,j} - \alpha(X_{c,j} - ant_{m,j}), j = 1, \dots, n$$

com $X_{m,j}^c$ representando o j -ésimo componente da compressão X_m^c , do melhor anticorpo A_m .

Em cada iteração do método quase-simplex, o melhor anticorpo gerado pelas transformações descritas anteriormente, substitui o pior anticorpo A_p e os anticorpos são reordenados. Então o processo é repetido por um número fixo de vezes (50 nos experimentos realizados nesta tese). Após este laço, obtém-se um conjunto de anticorpos χ formados pelos anticorpos resultantes do processo de busca local aplicado ao conjunto dos $n + 1$ melhores anticorpos hipermutados. Na sequência, os sete melhores indivíduos do conjunto χ

são selecionados para substituírem os sete piores indivíduos da população de anticorpos hipermutados. O valor sete é baseado no número de transformações aplicáveis durante o processo.

Neste ponto as restrições podem ser violadas (visto que nenhuma medida especial foi tomada nos operadores ou na busca local para evitar que as restrições fossem violadas), podendo gerar anticorpos inactivíveis. Para evitar tal violação um processo de reparo é aplicado a cada anticorpo selecionado do processo de busca local para garantir que os anticorpos incorporados à população sejam factíveis. O procedimento de reparo implementado é descrito no Algoritmo 4.

Algoritmo 4 Procedimento de reparo - sem perdas

```

Repita para cada componente  $j$  de um anticorpo  $i$ 
  Se  $ant_{i,j} < P_j^{min}$ 
     $ant_{i,j} = P_j^{min}$ 
  Senão
    Se  $ant_{i,j} > P_j^{max}$ 
       $ant_{i,j} = P_j^{max}$ 
    Fim Se
  Fim Repita
Enquanto  $\sum_{i=1}^n P_i \neq PD$ 
  Selecione de acordo com uma distribuição uniforme um componente  $j$ 
  Se  $\sum_{i=1}^n P_i < PD$ 
    Some uma quantidade  $ant_{i,j}$  que não viole as
    restrições operacionais e que minimize a restrição de
    equilíbrio de demanda
  Senão
    Subtraia uma quantidade  $ant_{i,j}$  que não viole as
    restrições operacionais e que minimize a restrição de
    equilíbrio de demanda
  Fim enquanto

```

O procedimento de reparação do Algoritmo 4 é utilizado para as instâncias do problema do despacho onde a perda de energia é desconsiderada (ver Capítulo 4, Equação 4.7).

Outra contribuição original deste trabalho diz respeito ao procedimento de reparo para os problemas de despacho que consideram a perda de energia (ver Capítulo 4, Equação 4.6). Este procedimento é descrito no Algoritmo 5. Neste algoritmo, inicialmente garante-se que as restrições operacionais dos geradores sejam atendidas. Depois um gerador é sorteado de maneira aleatória utilizando uma distribuição uniforme. Então, utilizando a fórmula de Bhaskara, calcula-se qual deveria ser o valor produzido por aquele gerador para que a demanda seja atendida considerando-se as perdas de energia. Se o valor é factível (atende às restrições operacionais) o anticorpo é reparado fazendo com que o gerador

sorteado produza o valor calculado, caso contrário o valor do gerador é determinado de maneira aleatória utilizando uma distribuição uniforme no intervalo operacional do gerador e o processo é repetido até que a demanda seja atendida.

Outros algoritmos para reparação com perdas de energia foram propostos em Noman e Iba (2008) e Bhattacharya e Chattopadhyay (2010).

O procedimento proposto em Noman e Iba (2008) primeiro calcula $PowDiff = \sum_{i=1}^n P_i - (P_D + P_L)$, ou seja, a diferença entre a energia gerada e a energia demandada e perdida. Depois disto ele distribui da maneira mais uniforme possível entre os vários geradores. Esta abordagem não garante que o indivíduo será reparado e nem que após a reparação a restrição de balanceamento de potência será atendida, visto que seria necessário recalculer as perdas após a mudança das potências dos geradores. No caso de não conseguir reparar o indivíduo, um novo indivíduo gerado de maneira aleatória é criado. Este algoritmo é diferente do proposto.

Já o algoritmo proposto em Bhattacharya e Chattopadhyay (2010) é semelhante ao proposto, visto que ambos manipulam algebricamente as Equações 4.4 e 4.6 (ver Capítulo 4, Seção 4.1.1) e utilizam a fórmula de Bhaskara. A principal diferença é que o algoritmo proposto em Bhattacharya e Chattopadhyay (2010) utiliza apenas uma das raízes encontradas (aquela obtida por $-B + \sqrt{(2*A)}$) enquanto o algoritmo proposto considera as duas raízes obtidas. Além disto, não há informações de como eles tratam os casos onde não existem soluções para a equação ou a raiz encontrada está fora dos limites operacionais dos geradores, sendo que ambos os casos são tratados pelo procedimento proposto nesta tese - o qual garante sempre retornar uma solução factível.

Após a aplicação do procedimento de reparo a afinidade dada pela Equação 5.1 pode ser utilizada para avaliar a qualidade de um anticorpo.

O próximo passo no algoritmo implementado é a aplicação do operador de *aging*. Neste trabalho foi utilizado o operador de *aging* estático (CUTELLO et al., 2005a; CUTELLO et al., 2006). Este operador de *aging* elimina anticorpos antigos com a intenção de manter a diversidade da população e evitar a convergência prematura do algoritmo. Este operador permite que um anticorpo permaneça na população por no máximo τ_B gerações, depois deste período o algoritmo assume que este anticorpo é um ótimo local e deve ser eliminado da população. Um clone herda a idade de seu pai e tem sua idade zerada quando é hipermutado com sucesso (isto é, quando a hipermutação melhora a afinidade do anticorpo).

Finalmente, o último passo do processo de maturação da afinidade é a seleção dos anticorpos que irão fazer parte da próxima geração. O esquema utilizado é um operador de

Algoritmo 5 Procedimento de reparo - com perdas

```

Repita para cada componente  $j$  de um anticorpo  $x$ 
  Se  $ant_{x,j} < P_j^{min}$ 
     $ant_{x,j} = P_j^{min}$ 
  Senão
    Se  $ant_{x,j} > P_j^{max}$ 
       $ant_{x,j} = P_j^{max}$ 
    Fim Se
  Fim Repita
Reparou = Falso
Enquanto !Reparou
  Selecione de acordo com uma distribuição uniforme um componente  $k$ 
   $A = B_{kk}$ 
   $B = \sum_{i=1, i \neq k}^n B_{ik} * ant_{x,i} + \sum_{j=1, j \neq k}^n ant_{x,j} * B_{kj} + B_{0k} - 1$ 
   $C = P_d + \sum_{i=1, i \neq k}^n \sum_{j=1, j \neq k}^n ant_{x,i} * B_{ij} * ant_{x,j} + \sum_{t=1, t \neq k}^n B_{0t} * ant_{x,t} - \sum_{m=1, m \neq k}^n ant_{x,m} +$ 
 $B_{00}$ 
   $Raiz = B^2 - 4*A*C$ 
  Se  $Raiz \geq 0$ 
     $T_1 = (-B + \sqrt{Raiz})/2*A$ 
     $T_2 = (-B - \sqrt{Raiz})/2*A$ 
    Se  $T_1$  atende às restrições operacionais
       $ant_{x,k} = T_1$ 
    Senão
      Se  $T_2$  atende às restrições operacionais
         $ant_{x,k} = T_2$ 
      Senão
         $ant_{x,k} = P_k^{min} + U(0,1) * (P_k^{max} - P_k^{min})$ 
    Senão
       $ant_{x,k} = P_k^{min} + U(0,1) * (P_k^{max} - P_k^{min})$ 
  Fim enquanto

```

seleção ($\mu + \lambda$) (CUTELLO et al., 2006) que é aplicado aos pais e aos clones hipermutados que sobreviveram ao operador de *aging*. Ou seja, os μ anticorpos da população original são combinados com os clones hipermutados, melhorados com o processo de busca local, e os λ melhores anticorpos são selecionados para compor a próxima população.

5.2 O Sistema Imuno-cultural Proposto

O Algoritmo imuno-cultural proposto utiliza o sistema imune anteriormente descrito como o espaço populacional de um algoritmo cultural. O Algoritmo 6 sumariza os passos gerais do algoritmo proposto.

Algoritmo 6 Algoritmo imuno-cultural proposto

1. Inicialize a População de Anticorpos;
 2. Avalie a População;
 3. Inicialize o Espaço de Crenças;
 $gen \leftarrow 0$;
 4. Enquanto o Critério de Parada não for Satisfeito
 - 4.1. Clone a População;
 - 4.2. Aplique as Influências nas Hipermutações dos Clones
de acordo com a probabilidade de cada fonte de conhecimento;
 - 4.3. Aplique o Quase-Simplex;
 - 4.4. Aplique o Procedimento de Reparo;
 - 4.5. Avalie os Clones;
 - 4.6. Aplique o Operador de Aging na População e nos Clones;
 - 4.7. Aplique o Operador de Seleção ($\mu + \lambda$);
 - 4.8. Aplique a Função de Aceitação;
 - 4.9. Atualize o Espaço de Crenças;
 - 4.10. Aplique a Função de Influência Principal;
 - 4.11. $gen = gen + 1$;
- Fim Enquanto
-

Conforme salientado no Capítulo 1, pelo conhecimento do autor esta é a primeira vez que um sistema imunológico é utilizado como espaço populacional de um algoritmo cultural.

O sistema Imuno-Cultural pode ser considerado uma extensão do algoritmo anterior onde o espaço de crenças e os protocolos de comunicação são adicionados para melhorar o desempenho do sistema imune original. O espaço de crenças é utilizado para extrair informações da população de anticorpos e usa este conhecimento para guiar a geração de novos anticorpos durante os operadores de hipermutação através das funções de influência, isto é, os operadores de hipermutação são substituídos pelas funções de influência. Os protocolos de comunicação ditam quais anticorpos serão considerados durante a atualização do

espaço de crenças (função de aceitação) e a probabilidade do conhecimento armazenado no espaço de crenças influenciar o operador de hipermutação (função de influência principal).

5.2.1 Protocolos de Comunicação: Funções de Aceitação e de Influência

Neste trabalho foi utilizada a função de aceitação dinâmica (IACOBAN; REYNOLDS; BREWSTER, 2003c; REYNOLDS; PENG, 2005). Esta função de aceitação é dada por:

$$nAccepted = popSize * (accept_{perc} + \frac{accept_{perc}}{gen}) \quad (5.7)$$

onde $nAccepted$ é o número de anticorpos da população que será utilizado para atualizar as fontes de conhecimento, $popSize$ é o número de anticorpos na população, $accept_{perc}$ é um parâmetro que determina o percentual de anticorpos aceitos e gen é a geração atual. Esta função de aceitação permite que mais anticorpos contribuam para a atualização no início da evolução (quando há pouco conhecimento acumulado) e menos anticorpos no final da evolução (quando a maioria do conhecimento já foi adquirido).

A função de influência principal é responsável por selecionar (usando o método da roleta (BECERRA; COELLO, 2006a)) a fonte de conhecimento que irá influenciar os operadores de hipermutação. A função que está sendo proposta neste trabalho representa uma das contribuições originais da tese. No início, todas as fontes de conhecimento têm a mesma probabilidade de serem aplicadas (0,25). Durante o processo evolucionário, a probabilidade da k -ésima fonte de conhecimento influenciar o operador de mutação é dada pela Equação 5.8.

$$probKS_k = \zeta + (1,0 - NKS * \zeta) * (perc_{cost} * \frac{cost_{gbKS_k}}{popCost} + (1,0 - perc_{cost}) * \frac{ant_{gbKS_k}}{popSize}) \quad (5.8)$$

onde $probKS_k$ é a probabilidade de selecionar a k -ésima fonte de conhecimento para influenciar o operador de hipermutação, ζ é taxa mínima de aplicação de cada fonte de conhecimento, NKS é a quantidade de fontes de conhecimento utilizadas, $perc_{cost}$ é o percentual de atualização decorrente do custo dos anticorpos (dado por $exp^{-\frac{ger}{\theta}}$, onde ger é a geração atual e θ é um parâmetro que ajusta o decréscimo do percentual), $cost_{gbKS_k}$ é o custo médio dos anticorpos gerados pela k -ésima fonte de conhecimento, $popCost$ é a soma do custo médio de todas as fontes de conhecimento, ant_{gbKS_k} é o número de anticorpos que foram gerados com sucesso pela influência de k -ésima fonte de conhecimento e $popSize$ é o tamanho da população.

A Equação 5.8 favorece a aplicação de fontes de conhecimento que são capazes de gerar anticorpos com baixo custo (alta afinidade) e de manter seus anticorpos gerados na população. Além disso, ela garante que cada fonte de conhecimento tenha pelo menos $(100 * \zeta)\%$ de chance de ser aplicada, evitando que qualquer fonte de conhecimento seja descartada do processo.

Esta função de influência principal tenta balancear a função de influência principal aleatória baseada na taxa de sucesso de cada tipo de conhecimento (SALEEM, 2001; BECERRA; COELLO, 2005b; BECERRA; COELLO, 2006a) e a função de influência principal baseada no Teorema do Valor Marginal, a qual pode ser vista como um jogo do tipo predador/presa onde as fontes de conhecimento são os predadores e os ótimos são as presas (REYNOLDS; PENG, 2006b; REYNOLDS; ALI; ALOMARI, 2006; REYNOLDS; PENG; ALI, 2008).

Esta função de influência principal, além de representar uma importante inovação deste trabalho, permite uma maneira robusta e flexível de definir quando qual ou quais conhecimentos serão utilizados. Assim, com a implementação de todas as fontes de conhecimento, o método pode se ajustar ao problema, definindo qual conhecimento é mais importante para a aplicação e mesmo qual conhecimento é importante em uma determinada fase da evolução.

5.2.2 Fontes de Conhecimento

As seguintes fontes de conhecimento são utilizadas neste trabalho:

Conhecimento Situacional: o conhecimento situacional armazena, no espaço de crenças, os *pbest* melhores anticorpos encontrados durante o processo evolucionário (BECERRA; COELLO, 2006a) .

No processo de atualização, sempre que um novo anticorpo (com *fitness* melhor do que o pior anticorpo armazenado no espaço de crenças) é encontrado, este melhor do espaço populacional substitui aquele pior do espaço de crenças.

No conhecimento situacional, os anticorpos armazenados no espaço de crenças são utilizados como guias para influenciar os operadores de hipermutação. Assim, a influência do conhecimento situacional é similar aos operadores de hipermutação Gaussiana e de Cauchy (Equação 5.3), conforme mostra a Equação 5.9.

$$ant_{i,j} = ant_{k,j}^{best(s)} + mult * (Cost_i / MinCost_{gen}) * R(0,1) * (P_j^{max} - P_j^{min}) \quad (5.9)$$

onde o termo $ant_{i,j}$ representa o j -ésimo componente do anticorpo i o qual vai sofrer a influência do conhecimento situacional, $ant_{k,j}^{best(s)}$ é o j -ésimo componente do k -ésimo melhor anticorpo aceito no espaço de crenças pelo conhecimento situacional, e k é um índice selecionado aleatoriamente dentre os melhores anticorpos do espaço de crenças.

Conhecimento Normativo: o conhecimento normativo contém os intervalos das potências de saída das unidades geradoras onde boas soluções foram encontradas. Este conhecimento é usado para mover novas soluções para estes intervalos.

Os intervalos do conhecimento normativo são inicializados com os limites mínimos e máximos de potência de cada gerador. A atualização do conhecimento normativo pode reduzir ou expandir os intervalos armazenados no espaço de crenças. Uma expansão acontece quando indivíduos aceitos não se encaixam no intervalo, enquanto uma redução ocorre quando todos os indivíduos aceitos encontram-se dentro do intervalo e os extremos do intervalo são associados com indivíduos de melhor custo (BECERRA; COELLO, 2005b). Neste caso o procedimento é similar ao mostrado na Equação 2.1, apenas restrições são incorporadas para atender aos critérios operacionais dos geradores. A atualização mostrada na Equação 5.10 considera um problema de minimização.

$$l_j = \begin{cases} \max(ant_{k,j}, P_j^{min}) & \text{se } ant_{k,j} < l_j \text{ ou } f(A_k^{best}) < L_j \\ l_j & \text{caso contrário.} \end{cases} \quad (5.10)$$

$$u_j = \begin{cases} \min(ant_{k,j}, P_j^{max}) & \text{se } ant_{k,j} > u_j \text{ ou } f(A_k^{best}) < U_j \\ u_j & \text{caso contrário.} \end{cases}$$

onde l_j e u_j representam os intervalos inferior e superior, respectivamente, armazenados no espaço de crenças, considerando o j -ésimo gerador. Os termos L_j e U_j representam o *fitness* dos anticorpos que resultaram nos limites l_j e u_j , respectivamente. Já o termo $ant_{k,j}$ representa o valor do j -ésimo gerador do anticorpo A_k^{best} o qual acaba de ser aceito para o espaço de crenças. O termo $\max(ant_{k,j}, P_j^{min})$ é necessário para garantir as condições operacionais sempre que há modificação no limite inferior: seja uma expansão ($ant_{k,j} < l_j$) ou uma contração ($ant_{k,j} > l_j$ e $f(A_k^{best}) < L_j$) deste limite. Por outro lado, o termo $\min(ant_{k,j}, P_j^{max})$ é necessário para garantir as condições operacionais sempre que há modificação no limite superior: seja uma expansão ($ant_{k,j} > u_j$) ou uma contração ($ant_{k,j} < u_j$ e $f(A_k^{best}) < U_j$) deste limite.

Sempre que o intervalo é atualizado, o *fitness* associado também deve ser atualizado: $L_j = f(A_k^{best})$ no caso de alteração do limite inferior e $U_j = f(A_k^{best})$ no caso de alteração do limite superior.

A influência do conhecimento normativo acontece da seguinte forma:

$$ant_{i,j} = \begin{cases} ant_{i,j} + mult * Cost_i / MinCost_{gen} * Abs(R(0,1)) * (u_j - l_j), \\ \text{se } ant_{i,j} < l_j \\ ant_{i,j} - mult * Cost_i / MinCost_{gen} * Abs(R(0,1)) * (u_j - l_j), \\ \text{se } ant_{i,j} > u_j \\ ant_{i,j} + mult * Cost_i / MinCost_{gen} * R(0,1) * (u_j - l_j), \\ \text{caso contrário} \end{cases} \quad (5.11)$$

onde u_j e l_j são, respectivamente, os limites superiores e inferiores do intervalo normativo associado ao j -ésimo componente, $ant_{i,j}$ é o j -ésimo componente do anticorpo i o qual vai sofrer a influência do conhecimento normativo e os outros termos foram definidos na Equação 5.3. Esta função de influência é adaptativa: ela é intensiva quando o intervalo normativo é amplo (a incerteza em relação ao valor ótimo é grande) e suave quando o intervalo normativo é estreito (o intervalo normativo vai se aproximando do valor que otimiza a função).

Conhecimento Histórico: o conhecimento histórico foi introduzido nos algoritmos culturais como um meio de adaptação a mudanças ambientais (REYNOLDS; PENG, 2006c). Ele armazena uma lista dos melhores anticorpos encontrados antes da última mudança ambiental. Ele também armazena a direção média e a distância média (tamanho) de cada mudança para cada componente entre as mudanças ambientais.

Neste trabalho, como não existem mudanças ambientais, este conhecimento é adaptado e a atualização acontece quando o algoritmo encontra um mínimo local (ou seja, não há mudanças no melhor anticorpo encontrado há pelo menos $intervger_{minlocal}$ gerações).

A atualização do conhecimento funciona da seguinte forma: sempre que o algoritmo encontra um ótimo local (melhor indivíduo que não sofre melhoria há pelo menos $intervger_{minlocal}$ gerações), este é aceito para fazer parte do espaço de crenças. Esta aceitação de um novo elemento na lista representa o disparo (um “gatilho”) na lista de eventos do conhecimento histórico. Assim o conhecimento histórico vai armazenando a lista dos últimos mínimos locais encontrados. Este conhecimento também armazena informações em relação a distância e direção entre os pares de mínimos locais. Sempre que um novo mínimo local é incluído no espaço de crenças, os valores de médias de distância e direção (Equação 5.12) são atualizados, tal que

$$\begin{aligned} MedDist_j &= (\sum_{k=1}^{TJ-1} Abs(ant_{k+1,j}^{melhor(h)} - ant_{k,j}^{melhor})) / (TJ - 1) \\ MedDir_j &= (\sum_{k=1}^{TJ-1} Sinal(ant_{k+1,j}^{melhor(h)} - ant_{k,j}^{melhor})) \end{aligned} \quad (5.12)$$

onde TJ é o número de anticorpos armazenados no conhecimento histórico, $ant_{q,j}^{melhor(h)}$ é o j -ésimo componente do anticorpo armazenado no q -ésimo evento do conhecimento histórico, $Abs(.)$ é uma função que retorno o módulo do seu argumento e $Sinal(.)$ é uma função que retorna +1 se o argumento é positivo, -1 se o argumento é negativo e 0 caso contrário.

A função de influência do conhecimento histórico utilizada neste trabalho também é uma contribuição da tese e é dada pela Equação 5.13.

$$ant_{i,j} = \begin{cases} ant_{i,j} + mult * Cost_i / MinCost_{gen} * Abs(R(0,1)) * (MedDist_j), & \text{se } MedDir_j \geq 0 \\ ant_{i,j} - mult * Cost_i / MinCost_{gen} * Abs(R(0,1)) * (MedDist_j), & \text{se } MedDir_j < 0 \end{cases} \quad (5.13)$$

onde o termo $ant_{i,j}$ representa o j -ésimo componente do anticorpo i o qual vai sofrer a influência do conhecimento histórico e os outros termos foram previamente definidos.

Esta influência tenta incrementar o j -ésimo componente do anticorpo submetido à hipermutação se em média o j -ésimo componente do novo melhor anticorpo for maior ou igual ao j -ésimo componente do melhor anticorpo anterior e tenta decrementar este componente caso contrário. Em qualquer um dos casos a taxa de hipermutação é proporcional a distância observada entre as mudanças (então se novos melhores anticorpos são encontrados distantes dos anteriores a hipermutação será intensiva e ela será suave se eles são encontrados próximos dos anteriores).

A Figura 23 apresenta um exemplo do funcionamento do Conhecimento Histórico. É possível observar que de um evento para outro (G representa geração atual, então $G20$ representa a vigésima geração) são armazenadas a distância e a direção. No caso da ocorrência de um mínimo local, é possível saber quais os passos (distância e direção seguidas) indicados pelo conhecimento histórico para tentar escapar dele.

Conhecimento Topográfico: o conhecimento topográfico é utilizado para criar um mapa da superfície do *fitness* do problema durante o processo evolutivo. Este conhecimento consiste de um conjunto de regiões e os melhores indivíduos encontrados em cada região. Ele também armazena uma lista das *TotReg* melhores regiões (as quais são ordenadas de acordo com seus melhores anticorpos). Neste trabalho, uma região é representada como um nó em uma árvore binária e armazena os limites superiores e inferiores de cada componente, o ponto de corte (variável e valor onde haverá a subdivisão do nó em dois nós filhos) e o melhor anticorpo encontrado até o momento naquela região conforme ilustrado na Figura 24. Por exemplo, o nó B da figura indica que o intervalo para a variável x nesta

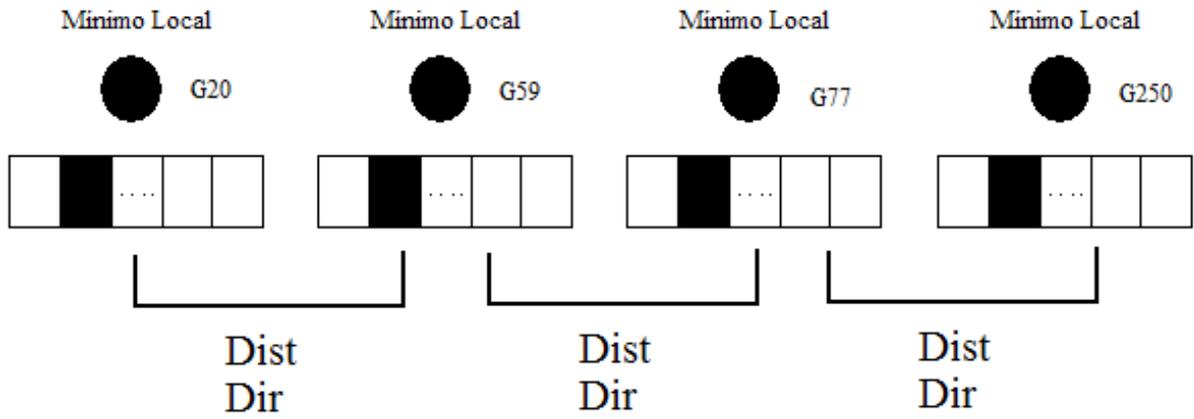


Figura 23: Representação do conhecimento histórico - Figura extraída de Kuk (2009)

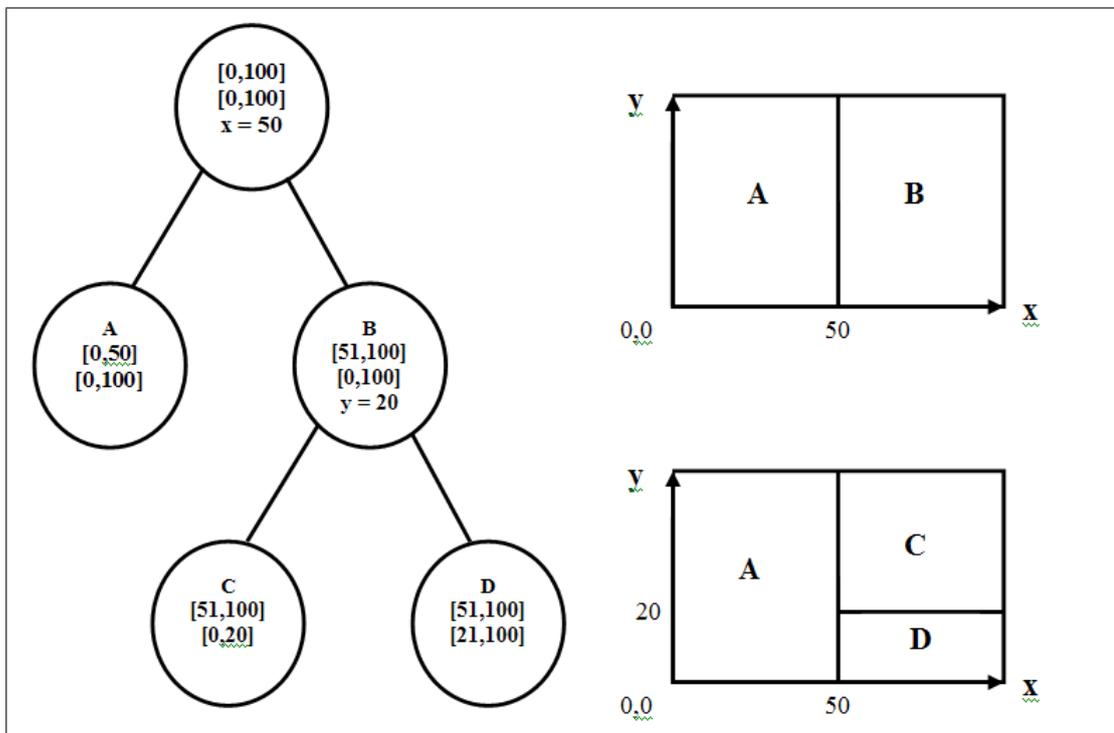


Figura 24: Árvore binária representando as regiões no conhecimento topográfico.

região vai de 51 a 100, para a variável y vai de 0 a 100 e o ponto de corte ocorre no valor 20 da variável y (o melhor indivíduo foi omitido da figura por questão de clareza).

A árvore binária é inicializada com um nó raiz que representa o espaço factível inteiro e tem o melhor anticorpo da primeira geração. Se durante a atualização do conhecimento topográfico um anticorpo com um melhor custo que o melhor anticorpo representado em uma região é aceito e este anticorpo pertence a esta região, o nó é dividido (a região é subdividida em duas). O componente onde a divisão é feita é o componente onde existe a maior diferença entre o melhor anticorpo anterior desta região e o novo melhor anticorpo (ver Figura 25).

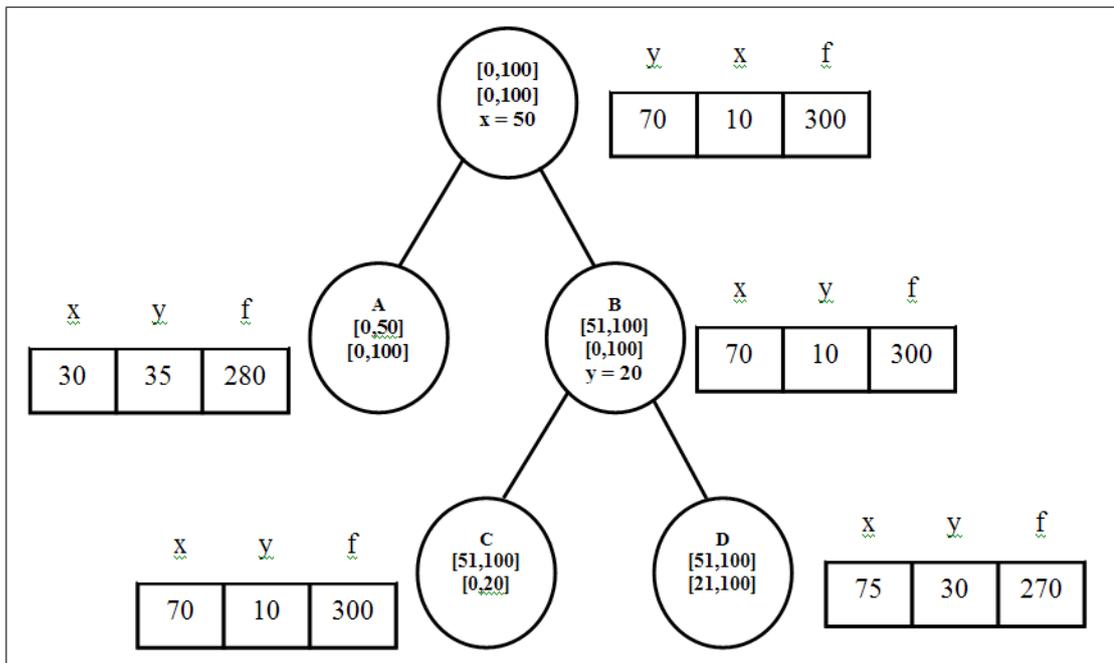


Figura 25: Representação do conhecimento topográfico após 2 divisões.

A divisão é feita no valor médio entre o valor deste componente no antigo melhor e o valor deste componente no novo melhor, conforme mostra a Equação 5.14.

$$PtDiv = \begin{cases} Novo_j + (Velho_j - Novo_j)/2, & \text{se } Velho_j \geq Novo_j \\ Velho_j + (Novo_j - Velho_j)/2, & \text{caso contrário} \end{cases} \quad (5.14)$$

onde $PtDiv$ é o valor do j -ésimo componente onde a região será dividida, $Velho_j$ é o j -ésimo componente do melhor anticorpo anterior da célula sendo dividida e $Novo_j$ é o j -ésimo componente do novo melhor anticorpo encontrado nesta célula. É importante notar que somente nós folha podem ser armazenados em uma lista ordenada ou serem divididos.

Cabe salientar que tanto a forma de armazenamento quando o procedimento de atualização do conhecimento topográfico adotados neste trabalhos constituem contribuições para a área.

A forma de armazenamento (árvore binária) difere da árvore k-d utilizada nos trabalhos de Coello e Becerra (COELLO; BECERRA, 2002b; BECERRA, 2002; BECERRA; COELLO, 2005b; BECERRA; COELLO, 2006a) e das hipergrades com uma árvore k-d por célula da hipergrade adotados nos trabalhos de Reynolds (REYNOLDS; PENG, 2006b; REYNOLDS; ALI; ALOMARI, 2006; REYNOLDS; ALI, 2008b; REYNOLDS; ALI, 2008a). Computacionalmente, o custo de armazenamento da árvore binária é inferior.

Além disto, a atualização proposta é mais simples tanto do ponto de vista da complexidade de implementação quanto da complexidade computacional de execução. As propostas anteriores reportadas na literatura necessitam buscar, em toda a população, os melhores indivíduos para povoar cada um dos novos nós gerados na árvore. Às vezes, há até mesmo a necessidade de gerar indivíduos de forma aleatória para preencher determinados nós da árvore (regiões do espaço topográfico) caso nenhum indivíduo pertença a região compreendida por aquele nó. Já o procedimento proposto sempre possui dois indivíduos após a subdivisão sendo que cada indivíduo pertence a um dos nós gerados, ou seja, nunca é necessário buscar o melhor indivíduo da população pertencente a uma nova região nem gerar um novo indivíduo aleatório que pertença àquela região, diminuindo o custo computacional do procedimento e tornando a sua implementação mais simples.

A função de influência do conhecimento topográfico é descrita na Equação 5.15.

$$ant_{i,j} = ant_{i,j} + mult * Cost_i / MinCost_{gen} * R(0,1) * (R_{k,j}^{sup} - R_{k,j}^{inf}) \quad (5.15)$$

onde $R_{k,j}^{sup}$ e $R_{k,j}^{inf}$ são os limites superiores e inferiores para o j-ésimo componente na k-ésima região a qual é escolhida com base na probabilidade p_k , o termo $ant_{i,j}$ representa o j-ésimo componente do anticorpo i o qual vai sofrer a influência do conhecimento topográfico e os demais termos foram previamente definidos.

A probabilidade de escolha da k-ésima região é dada por

$$p_k = \begin{cases} \frac{func(aff_{A_m^k})}{\sum_{i=1}^{TotReg} func(aff_{A_m^i})} & \text{se } u \in [0, 1] < pElite \\ \frac{1}{TotReg} & \text{caso contrário.} \end{cases} \quad (5.16)$$

onde $aff_{A_m^k}$ representa o *fitness* do melhor anticorpo A_m^k armazenado na região k , $TotReg$ é o total de regiões armazenadas no espaço de crenças pelo conhecimento topográfico, u é um número aleatório uniforme no intervalo $[0, 1]$ e $pElite$ é um fator que define a importância das melhores regiões na influência do conhecimento topográfico. Os outros termos da equação foram previamente definidos.

Assim, a região k tem probabilidade de escolha definida com base na afinidade do melhor anticorpo armazenado em k , se u é menor que $pElite$ (probabilidade das melhores regiões serem escolhidas com maior frequência) e selecionada aleatoriamente (com distribuição uniforme e independentemente da afinidade), caso contrário.

Esta função de influência tende a explorar boas regiões do espaço de busca e também representa uma contribuição deste trabalho.

5.3 O Sistema Imuno-Cultural Caótico Proposto

Técnicas de otimização que fazem uso de sequências caóticas (STROGATZ, 2000; ANISHCHENKO et al., 2007) têm sido aplicadas com sucesso em problemas de despacho econômico (COELHO; MARIANI, 2006; CAI et al., 2007; COELHO; LEE, 2008), motivando o uso deste tipo de sequência neste trabalho.

A diferença entre o Sistema Imune Cultural (SIC) e o Sistema Imune Cultural Caótico (SIC_c) reside na implementação do multiplicador (*mult*) utilizado nas influências da hipermutação. O SIC implementa este multiplicador como uma função decrescente com relação ao número da geração corrente (ver Equação 5.4), enquanto o SIC_c implementa o multiplicador como uma sequência caótica.

A sequência caótica utilizada neste trabalho é gerada por um mapa logístico, devido a sua simplicidade, (Equação 5.17).

$$mult_{gen} = r_c * mult_{gen-1} * (1,0 - mult_{gen-1}) \quad (5.17)$$

onde $mult_{gen}$ é o multiplicador utilizado na geração gen e r_c é uma constante que controla o comportamento do mapa logístico. Nem todos os valores de r_c produzem sequências caóticas, mas neste trabalho apenas valores que originam sequências caóticas foram considerados. Apesar de valores de r_c maiores do que 4 resultarem em sequências caóticas, estes valores não foram considerados por não gerarem sequências contidas no intervalo $[0, 1]$.

5.4 O Sistema Imuno-Cultural Multiobjetivo Proposto

Foram implementadas três versões multiobjetivo (SICMO) dos algoritmos imuno-culturais: uma versão escalarizada (SICMO_e), uma versão baseada em dominância de Pareto com armazenamento em grade adaptativa (SICMO_p) e uma versão caótica do algoritmo baseado em dominância de Pareto (SICMO_{pc}). Seguem detalhes sobre os algoritmos propostos.

5.4.1 Sistema Imuno-Cultural Multiobjetivo Escalarizado

O Sistema Imuno-Cultural Multiobjetivo Escalarizado (SICMO_e) é uma modificação simples do algoritmo imuno-cultural mono-objetivo (SIC) para trabalhar com um vetor de valores de escalarização. Nesta versão escalarizada, o espaço populacional e o espaço de crengas não sofrem nenhum tipo de alteração para lidar com problemas multiobjetivo. Apenas a função de *fitness* é modificada para avaliar tanto a questão econômica quanto a questão ambiental das instâncias do problema multiobjetivo. A Equação 5.18 mostra a modificação no cálculo da afinidade do anticorpo *i*.

$$Fitness_i = \lambda * Cost_i + (1 - \lambda) * \kappa * Emission_i \quad (5.18)$$

onde $Fitness_i$ é o *fitness* do *i*-ésimo anticorpo, λ é o fator de escalarização (o qual varia de 0,0 a 1,0 em intervalos regulares), κ é um termo para balancear as diferenças de grandezas entre os valores de custo e emissão (calculado pela divisão do custo obtido ao se considerar a otimização apenas da emissão pelo valor da emissão ao se considerar a otimização somente do objetivo custo), $Cost_i$ e $Emission_i$ são, respectivamente, o custo e a emissão do *i*-ésimo anticorpo.

Então a afinidade de um anticorpo com o antígeno é modificada para:

$$aff_i = (MaxFitness_{gen} - Fitness_i) / (MaxFitness_{gen} - MinFitness_{gen}) \quad (5.19)$$

onde $Fitness_i$ é o *fitness* do *i*-ésimo anticorpo, $MinFitness_{gen}$ e $MaxFitness_{gen}$ são o *fitness* máximo e mínimo de um anticorpo na geração *gen*, respectivamente.

A Figura 26 ilustra o processo de escalarização realizado pelo algoritmo proposto para o problema biobjetivo considerado neste trabalho, considerando o valor de κ igual a 1,0.

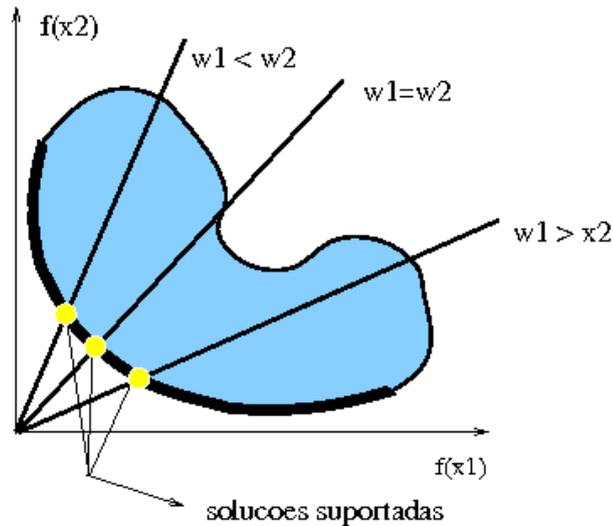


Figura 26: Representação do espaço multiobjetivo para uma versão escalarizada

Na figura 26, estão em destaque as soluções (suportadas) geradas para três elementos de escalarização, considerando-se $w_1 = \lambda$ e $w_2 = (1 - \lambda) * \kappa$, $f(x_1) = custo$ e $f(x_2) = emissão$. O objetivo do método proposto é gerar, em cada ciclo de combinação de valores (w_1, w_2) , soluções mais próximas possíveis das soluções suportadas.

O pseudocódigo desta abordagem é mostrado no Algoritmo 7. As mudanças para o algoritmo SIC residem nos passos 1., 3., 5.5. e 6. (destacados com fundo cinza no algoritmo), os demais passos correspondem exatamente aos passos do SIC, mas ao invés do custo a função de *fitness* é a descrita na Equação 5.18. O passo 1. faz com que o algoritmo processe as diferentes escalarizações enquanto o passo 6. armazena as soluções encontradas para montar a aproximação da fronteira de Pareto.

5.4.2 Sistema Imuno-Cultural Multiobjetivo Baseado em Dominância de Pareto

O Sistema Imuno-Cultural Multiobjetivo Baseado em Dominância de Pareto (SICMO_p) possui várias diferenças com relação a versão mono-objetivo (SIC). O Algoritmo 8 apresenta o pseudocódigo desta versão.

A primeira diferença para o algoritmo mono-objetivo é o uso da grade adaptativa para o arquivamento externo das soluções não dominadas (KNOWLES; CORNE, 2003) (ver Subseção 5.4.2.1). Esta grade é inicializada com os indivíduos não-dominados da população inicial.

No tocante ao espaço populacional utilizado pelo SICMO_p a principal alteração é que agora ao invés da seleção ocorrer por um algoritmo do tipo $(\mu + \lambda)$ (onde os melhores

Algoritmo 7 Algoritmo imuno-cultural multiobjetivo escalarizado proposto

1. Para cada elemento do Vetor de Escalarização
 2. Inicialize a População de Anticorpos;
 3. Avalie a População Considerando o elemento de Escalarização;
 4. Inicialize o Espaço de Crenças;
 $gen \leftarrow 0$;
 5. Enquanto o Critério de Parada não for Satisfeito
 - 5.1. Clone a População;
 - 5.2. Aplique as Influências nas Hipermutações dos Clones de acordo com a probabilidade de cada fonte de conhecimento;
 - 5.3. Aplique o Quase-Simplex;
 - 5.4. Aplique o Procedimento de Reparo;
 - 5.5. Avalie os Clones Considerando o elemento de Escalarização;
 - 5.6. Aplique o Operador de Aging na População e nos Clones;
 - 5.7. Aplique o Operador de Seleção ($\mu + \lambda$);
 - 5.8. Aplique a Função de Aceitação;
 - 5.9. Atualize o Espaço de Crenças;
 - 5.10. Aplique a Função de Influência Principal;
 - 5.11. $gen = gen + 1$;
 Fim Enquanto
 6. Armazene a Solução Encontrada na Aproximação do Conjunto de Pareto;
- Fim Para
-

Algoritmo 8 Algoritmo imuno-cultural multiobjetivo baseado em dominância de Pareto proposto

1. Inicialize a População de Anticorpos;
 2. Avalie a População;
 3. Inicialize a Grade Adaptativa
 4. Inicialize o Espaço de Crenças;
 $gen \leftarrow 0$;
 5. Enquanto o Critério de Parada não for Satisfeito
 - 5.1. Clone a População;
 - 5.2. Aplique as Influências nas Hipermutações dos Clones de acordo com a probabilidade de cada fonte de conhecimento;
 - 5.3. Aplique o Procedimento de Reparo;
 - 5.4. Avalie os Clones;
 - 5.5. Se algum Clone domina o Anticorpo Original ou não é dominado por ele
 - 5.5.1. Insira o Melhor Clone na Grade Adaptativa;
 - 5.5.2. Substitua o Anticorpo Original pelo Melhor Clone;
 - 5.6. Aplique o Operador de Aging na População e nos Clones;
 - 5.7. Aplique a Função de Aceitação;
 - 5.8. Atualize o Espaço de Crenças;
 - 5.9. Aplique a Função de Influência Principal;
 - 5.10. $gen = gen + 1$;
 Fim Enquanto
-

anticorpos entre os anticorpos originais e os clones hipermutados são selecionados para a próxima geração) é aplicado um torneio entre os anticorpos hipermutados e o anticorpo original que foi utilizado na clonagem e hipermutação. Este torneio é baseado em dominância de Pareto. Em caso de empate entre um anticorpo hipermutado e o anticorpo original, o anticorpo hipermutado vence. Se um clone vencer o torneio ele é inserido na grade adaptativa indicando que o processo de hipermutação foi capaz de gerar uma solução do conjunto de aproximação do conjunto pareto-ótimo. Cabe salientar que esta inserção pode falhar em dois casos: se o anticorpo hipermutado for dominado por algum anticorpo armazenado na grade ou se o anticorpo hipermutado pertencer a célula mais populosa da grade e a grade já estiver em sua capacidade máxima de armazenamento.

Outra alteração adotada no espaço populacional é a eliminação do procedimento quase-simplex. Esta eliminação se deve ao fato do método quase-simplex ser um método de busca-local para problemas mono-objetivo e o SICMO_p trabalhar com problemas multiobjetivo utilizando o conceito de dominância de Pareto.

As principais alterações se deram no espaço de crenças. A primeira diferença é que o espaço de crenças é dividido no número de objetivos do problema a ser resolvido. Ou seja, se o problema tiver dois objetivos o espaço de crenças terá dois tipos de conhecimento situacional, dois normativo e assim por diante, conforme ilustrado na Figura 27.

Por isto as funções de aceitação e influência principal tiveram que ser modificadas. A fórmula da função de aceitação é a mesma utilizada na versão mono-objetivo (Equação 5.7), mas para cada objetivo os anticorpos da população são ordenados de acordo com aquele objetivo e os melhores anticorpos são utilizados na atualização dos conhecimentos associados àquele objetivo. Ou seja, durante a aceitação do *i*-ésimo objetivo os anticorpos são ordenados de acordo com os seus valores para este objetivo e os *nAccepted* melhores são aceitos para influenciar os conhecimentos baseados neste conhecimento.

Agora a influência principal deve primeiramente selecionar qual objetivo será utilizado para influenciar as hipermutações. Esta escolha, no momento, é realizada de forma aleatória: todos os objetivos possuem a mesma probabilidade de serem escolhidos. Na sequência a Equação 5.20 é utilizada para determinar a probabilidade de aplicação da influência de cada tipo de conhecimento para o objetivo sorteado.

$$probKS_{o,k} = \zeta + (1,0 - NKS * \zeta) * (perc_{fitness_o} * \frac{fitness_{gbKS_{o,k}}}{popFitness_i} + (1,0 - perc_{fitness_o}) * \frac{ant_{gbKS_{o,k}}}{popSize}) \quad (5.20)$$

onde $probKS_{o,k}$ é a probabilidade de selecionar a *k*-ésima fonte de conhecimento do *o*-ésimo objetivo para influenciar o operador de hipermutação, ζ é a taxa mínima de

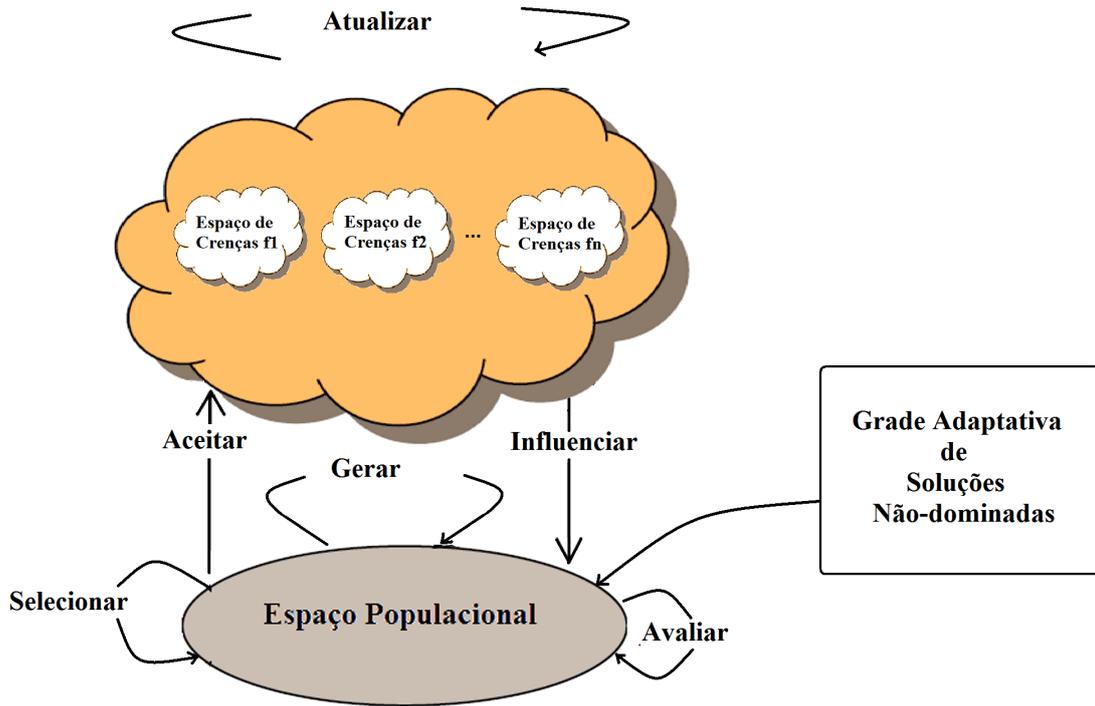


Figura 27: Divisão do espaço de crenças do algoritmo imuno-cultural multiobjetivo baseado em dominância de Pareto

aplicação de cada fonte de conhecimento, NKS é a quantidade de fontes de conhecimento utilizadas, $perc_{fitness_o}$ é o percentual de atualização decorrente do valor dos anticorpos para o o -ésimo objetivo (dado por $exp^{-\frac{ger}{\theta}}$, onde ger é a geração atual e θ é um parâmetro que ajusta o decréscimo do percentual), $fitness_{gbKS_{o,k}}$ é o $fitness$ médio dos anticorpos gerados pela k -ésima fonte de conhecimento para o o -ésimo objetivo, $popFitness_o$ é a soma dos $fitness$ médio de todas as fontes de conhecimento para o o -ésimo objetivo, $ant_{gbKS_{o,k}}$ é o número de anticorpos que foram gerados com sucesso pela influência de k -ésima fonte de conhecimento para o i -ésimo objetivo e $popSize$ é o tamanho da população.

Por exemplo, para o caso do despacho econômico/ambiental, a primeira parte consiste de escolher entre o objetivo custo ou emissão de poluentes. Suponha que o objetivo sorteado tenha sido a emissão. O segundo passo consiste em escolher qual conhecimento referente à emissão será utilizado, com base na probabilidade de cada fonte de conhecimento, probabilidade esta definida pela Equação 5.20. Assim, assumindo que o conhecimento topográfico seja o escolhido, a influência aplicada seria a influência do conhecimento topográfico do objetivo emissão, cuja função é definida pela Equação 5.24 descrita adiante.

As atualizações e as funções de influência dos conhecimentos da versão mono-objetivo (Equações 5.9 a 5.15) foram modificadas para levarem em consideração apenas o objetivo

ao qual o conhecimento está associado. Deste modo, para os conhecimentos baseados no custo as equações permanecem inalteradas e para os conhecimentos baseados na emissão as variáveis baseadas no custo são substituídas por variáveis baseadas na emissão. Por completude, as influências dos conhecimento do $SICMO_p$ são apresentadas nas Equações 5.21 a 5.24.

A influência do conhecimento situacional é dado por:

$$ant_{i,j} = ant_{k,j}^{best(s_o)} + mult * (Fitness_{i,o} / MinFitness_{o,gen}) * R(0,1) * (P_j^{max} - P_j^{min}) \quad (5.21)$$

onde o termo $ant_{i,j}$ representa o j-ésimo componente do anticorpo i o qual vai sofrer a influência do conhecimento situacional do o-ésimo objetivo, $ant_{k,j}^{best(s_o)}$ é o j-ésimo componente do k-ésimo melhor anticorpo aceito no espaço de crenças pelo conhecimento situacional do o-ésimo objetivo, $Fitness_{i,o}$ é o valor obtido pelo i-ésimo anticorpo para o-ésimo objetivo, $MinFitness_{o,gen}$ é o menor valor obtido por qualquer anticorpo da população da geração gen para o o-ésimo objetivo e k é um índice selecionado aleatoriamente dentre os melhores anticorpos do espaço de crenças.

No conhecimento normativo a influência acontece da seguinte forma:

$$ant_{i,j} = \begin{cases} ant_{i,j} + mult * (Fitness_{i,o} / MinFitness_{o,gen}) * Abs(R(0,1)) * (u_{j,o} - l_{j,o}), & \text{se } ant_{i,j} < l_{j,o} \\ ant_{i,j} - mult * (Fitness_{i,o} / MinFitness_{o,gen}) * Abs(R(0,1)) * (u_{j,o} - l_{j,o}), & \text{se } ant_{i,j} > u_{j,o} \\ ant_{i,j} + mult * (Fitness_{i,o} / MinFitness_{o,gen}) * R(0,1) * (u_{j,o} - l_{j,o}), & \text{caso contrário} \end{cases} \quad (5.22)$$

onde $u_{j,o}$ e $l_{j,o}$ são, respectivamente, os limites superiores e inferiores do intervalo normativo associado ao j-ésimo componente para o o-ésimo objetivo e os demais termos são como definidos anteriormente.

A função de influência do conhecimento histórico utilizada é dada pela Equação 5.23.

$$ant_{i,j} = \begin{cases} ant_{i,j} + mult * \frac{Fitness_{i,o}}{MinFitness_{o,gen}} * Abs(R(0,1)) * (MedDist_{j,o}), & \text{se } MedDir_{j,o} \geq 0 \\ ant_{i,j} - mult * \frac{Fitness_{i,o}}{MinFitness_{o,gen}} * Abs(R(0,1)) * (MedDist_{j,o}), & \text{se } MedDir_{j,o} < 0 \end{cases} \quad (5.23)$$

onde o termo $MedDist_{j,o}$ é a distância média entre os j -ésimos componentes dos anticorpos armazenados no conhecimento histórico do o -ésimo objetivo, $MedDir_{j,o}$ é a direção média armazenada no conhecimento histórico para o o -ésimo objetivo para o j -ésimo componente e os outros termos foram previamente definidos.

A função de influência do conhecimento topográfico é descrita na Equação 5.24.

$$ant_{i,j} = ant_{i,j} + mult * (Fitness_{i,o} / MinFitness_{o,gen}) * R(0,1) * (R_{k,j,o}^{sup} - R_{k,j,o}^{inf}) \quad (5.24)$$

onde $R_{k,j,o}^{sup}$ e $R_{k,j,o}^{inf}$ são os limites superiores e inferiores para o j -ésimo componente na k -ésima região do o -ésimo objetivo a qual é escolhida com base na probabilidade p_k e os demais termos foram previamente definidos.

A probabilidade de escolha da k -ésima região é dada por

$$p_k = \begin{cases} \frac{Fitness_{A_{m,o}^k}}{\sum_{i=1}^{TotReg_o} Fitness_{A_{m,o}^i}} & \text{se } u \in [0,1] < pElite \\ \frac{1}{TotReg_o} & \text{caso contrário.} \end{cases}$$

onde $Fitness_{A_{m,o}^k}$ representa o valor do melhor anticorpo $A_{m,o}^k$ armazenado na região k para o o -ésimo objetivo, $TotReg_o$ é o total de regiões armazenadas no espaço de crenças pelo conhecimento topográfico para o o -ésimo objetivo, u é um número aleatório uniforme no intervalo $[0,1]$ e $pElite$ é um fator que define a importância das melhores regiões na influência do conhecimento topográfico.

Outra alteração importante ocorreu no “gatilho” de evento do conhecimento histórico. Antes o “gatilho” era disparado sempre que um número pré-determinado de gerações sem melhorias ocorria, agora o “gatilho” é disparado sempre que houver uma modificação dos extremos da grade adaptativa, ou seja, sempre que um novo mínimo ou máximo seja encontrado para alguma das funções objetivo.

5.4.2.1 Grade Adaptativa

O algoritmo AGA (*Adaptive Grid Archiving*) (KNOWLES; CORNE, 2003) é baseado no método de arquivamento utilizado pelo algoritmo PAES (Pareto Archived Evolution Strategy) (KNOWLES; CORNE, 2000) e constitui uma forma de arquivo externo de soluções não-dominadas. Este algoritmo implementa estratégias para armazenar os pontos encontrados por um algoritmo evolucionário multiobjetivo de uma maneira bem distribuída e diversificada.

No algoritmo AGA o espaço objetivo é dividido em regiões, formando uma grade no espaço k -dimensional, onde k corresponde ao número de funções objetivo. O tamanho e a alocação das regiões da grade são adaptadas conforme os pontos são gerados ao longo da evolução do AE multiobjetivo. Sempre que novos extremos para as funções objetivo forem encontrados a grade é expandida e as regiões são realocadas, mantendo fixo o número de regiões (o qual é um parâmetro do algoritmo).

A grade é utilizada na seleção de quais pontos devem ser removidos do arquivo quando ele chega ao limite de sua capacidade de armazenamento (outro parâmetro do algoritmo). Quando isto acontece, um ponto que pertence à uma das regiões mais populosas da grade e que não represente um ponto extremo é escolhido para ser removido do arquivo.

O Algoritmo 9 apresenta os passos do AGA. O algoritmo considera M_t o arquivo de soluções/vetores não-dominados, M_{t-1} o arquivo de vetores não-dominados antes da sua última atualização, z_t o vetor objetivo gerado no tempo t (o qual deseja-se arquivar) e $z^{c,t}$ um vetor em M_{t-1} , selecionado aleatoriamente de um conjunto de regiões mais populosas.

Algoritmo 9 Algoritmo da arquivo de grade adaptativa

```

1: foreach ( $k \in 1..K$ )
2:   Recalcular os limites de todas as regiões da grade para a dimensão  $k$ 
3: Se ( $z_t > M_{t-1}$ )
4:    $M_t \leftarrow M_{t-1}$ 
5: Senão
6:   Se ( $z_t < M_{t-1}$ )
7:      $M_t \leftarrow M_{t-1} \cup \{z_t\} \setminus \{z \in M_{t-1} \mid z_t < z\}$ 
8:   Senão Se ( $|M_{t-1}| = \text{tamanhoArquivo}$ )
9:     Se ( $z$  está fora dos limites antigos da grade)
10:       $M_t \leftarrow M_{t-1} \cup \{z_t\} \setminus \{z^{c,t}\}$ 
11:     Senão Se ( $z$  não está na região mais populosa)
12:       $M_t \leftarrow M_{t-1} \cup \{z_t\} \setminus \{z^{c,t}\}$ 
13:   Senão
14:      $M_t \leftarrow M_{t-1}$ 
15:   Senão
16:      $M_t \leftarrow M_{t-1} \cup z_t$ 
17: retorne ( $M_t$ )

```

Inicialmente o algoritmo atualiza os limites da grade adaptativa (passo 2). A seguir, no passos 3 e 4, o ponto z_t é descartado, caso ele seja dominado por algum ponto do arquivo. Nos passos 6 e 7, o ponto z_t é aceito caso ele domine o arquivo e com isso todos os pontos dominados são removidos do arquivo. Caso o ponto z_t seja não-dominado com relação ao arquivo e o arquivo não tenha alcançado sua capacidade máxima ele é aceito no passo 16. Se o arquivo estiver com a sua capacidade máxima de pontos existem duas condições para este ponto não-dominado (z_t) ser aceito no arquivo: z_t estar fora dos limites

da grade (aumentando sua extensão em algum objetivo) e z_t localizar-se em uma região da grade menos populosa do que algum ponto no arquivo, neste caso um ponto localizado em uma das regiões populosas da grade é substituído por z_t .

Estes passos encorajam a progressão dos pontos para a fronteira de Pareto, mantendo os intervalos dos valores para cada objetivo extensos e tendendo a uma distribuição uniforme dos pontos no espaço objetivo.

5.4.3 Sistema Imuno-Cultural Caótico Multiobjetivo Baseado em Dominância de Pareto

A única diferença entre o $SICMO_p$ e a versão caótica ($SICMO_{pc}$) reside na definição de qual conjunto de crenças (conhecimentos) será escolhido pela influência principal. No $SICMO_p$ a escolha é aleatória enquanto no $SICMO_{pc}$ ela se dá por um mapa logístico. A Equação 5.25 apresenta o cálculo das probabilidades de aplicação das influências dos conhecimentos baseados no custo e na emissão.

$$\begin{aligned} probCost_{gen} &= r_c * probCost_{gen-1} * (1.0 - probCost_{gen-1}) \\ probEmission_{gen} &= 1.0 - probCost_{gen} \end{aligned} \quad (5.25)$$

onde $probCost_{gen}$ é a probabilidade de aplicar as influências dos conhecimentos baseados em custo durante a geração gen , $probEmission_{gen}$ é a probabilidade de aplicar as influências dos conhecimentos baseados em emissão durante a geração gen e r_c é uma constante que controla o comportamento do mapa logístico.

5.4.4 Comparação com a Literatura

A seguir as características dos algoritmos multiobjetivo propostos serão comparadas com aquelas associadas a alguns algoritmos evolucionários multiobjetivo da literatura. A comparação foca-se mais em outras abordagens culturais (ver Capítulo 3, Subseção 3.4.4.2) e imunológicos (ver Capítulo 3, Subseção 3.4.4.1) e no NSGA-II, dada a sua importância na literatura da área e no seu uso como algoritmo de comparação no Capítulo 6.

5.4.4.1 Algoritmos não Baseados em Dominância de Pareto

O $SICMO_e$ possui semelhanças com o algoritmo $\epsilon CCDE$ (BECERRA; COELLO, 2006c). Ambos transformam os problemas multiobjetivo em vários sub-problemas mono-objetivo. A principal diferença neste sentido é a forma dos sub-problemas: o $SICMO_e$

modifica a função objetivo enquanto o $\epsilon CCDE$ mantém como objetivo um dos objetivos originais e introduz uma restrição por objetivo adicional.

O uso de $\epsilon - constraints$ permite encontrar soluções suportadas e não-suportadas, o que permite que o $\epsilon CCDE$ possa encontrar soluções que o $SICMO_e$ não seria capaz. Mas em contra-partida o $\epsilon CCDE$ precisa de uma fase de pré-processamento (que dependendo do número de objetivos pode ser custosa) para encontrar um conjunto eficiente de ϵ 's, enquanto a fase de pré-processamento do $SICMO_e$ (geração dos vetores de escalarização) é simples.

5.4.4.2 Algoritmos Baseados em Dominância de Pareto

Assim como o algoritmo CPSA (GAO et al., 2007) e o Cultural MOPSO (DANESH-YARI; YEN, 2008), o algoritmo $SICMO_p$ faz uso de um arquivo externo para armazenar as soluções não-dominadas - o que os diferencia do MOCDE (QIN et al., 2010) que aproxima a fronteira de Pareto na própria população.

Com relação a função de aceitação também há diferenças significativas: o $SICMO_p$ usa uma função de aceitação dinâmica (no começo, quando os conhecimentos são poucos, escolhe-se muitos indivíduos e este valor é decrementado ao longo do processo evolutivo) para cada um dos objetivos, o CPSA utiliza uma função de aceitação estática para cada uma das sub-populações enquanto o Cultural MOPSO e o MOCDE aceitam uma certa quantidade dos indivíduos não-dominados.

As funções de influência dos algoritmos também são distintas. O motivo óbvio para esta distinção seria os diferentes espaços populacionais adotados, mas além disto eles aplicam conceitos diferentes. O CPSA, o Cultural MOPSO e o MOCDE usam soluções não-dominadas como guias para a influência do conhecimento situacional, já o $SICMO_p$ usa boas soluções para cada objetivo. Na influência do conhecimento normativo o $SICMO_p$ tenta mover as soluções para dentro do intervalo normativo, enquanto o MOCDE move soluções fora do intervalo normativo para cima da fronteira do intervalo normativo e o Cultural MOPSO utiliza o conhecimento normativo para aumentar ou diminuir a contribuição da posição atual na atualização da posição da partícula. Já o CPSA não faz uso de conhecimento normativo.

O CPSA também não utiliza conhecimento histórico, assim como o Cultural MOPSO. O $SICMO_p$ utiliza informações sobre a expansão da grade adaptativa para influenciar os indivíduos. Já no MOCDE o conhecimento histórico serve para medir se há evolução entre populações subsequentes e se há diversidade na população, caso nenhuma destas condições seja satisfeita um operador para aumentar a diversidade é aplicado. O grande problema

do conhecimento histórico do MOCDE é a necessidade de se ter uma boa aproximação para a fronteira de Pareto.

Apenas o SICMO_p e o Cultural MOPSO fazem uso de conhecimento topográfico. O SICMO_p representa o conhecimento por uma árvore binária e armazena as regiões onde os melhores valores para cada um dos objetivos foram encontrados e sua influência visa explorar boas regiões do espaço de busca. Já o Cultural MOPSO armazena uma grade adaptativa e utiliza o número de elementos em cada célula da grade como informação. O objetivo é guiar as partículas para as células menos povoadas.

A principal diferença entre o algoritmo proposto e os sistemas imunológicos multiobjetivo descritos na literatura é a forma de eliminação de anticorpos: no SICMO_p ela ocorre através do operador de *aging* enquanto nos demais algoritmos geralmente é utilizado um operador de supressão (na maioria das vezes baseados na distância entre os anticorpos).

Assim como a maioria dos SIAs multiobjetivo, o SICMO_p aplica hipermutações proporcionais à afinidade dos anticorpos. A diferença é que no SICMO_p cada anticorpo possui uma afinidade diferente para cada objetivo e as hipermutações são influenciadas pelos conhecimentos adquiridos ao longo do processo evolutivo.

Quando comparado ao NSGA-II, o SICMO_p apresenta inúmeras diferenças, tais como o uso de arquivo externo para armazenamento de soluções não-dominadas (grade adaptativa), operadores adaptativos (hipermutações influenciadas), extração de informações em tempo de evolução (atualização dos conhecimentos), eliminação de indivíduos que não estão evoluindo da população (*aging*) e seleção dos indivíduos para a próxima população (ordenação baseada em *ranks* e *crowding* no NSGA-II *versus* torneio binário baseado em dominância no SICMO_p).

6 *Simulações e Resultados Obtidos*

Para validar os algoritmos imuno-culturais propostos foram utilizadas algumas instâncias conhecidas do problema do despacho de energia elétrica (PDE) com efeito de ponto de válvula. O conjunto de instâncias foi dividido em duas partes: instâncias do problema mono-objetivo (otimização do custo de combustível dos geradores) e instâncias do problema biobjetivo (otimização do custo e da emissão de poluentes). Tanto no caso mono, quanto no caso biobjetivo, foram analisadas as situações considerando o caso ideal (sem perda de energia) e o caso real (com perda de energia).

No primeiro grupo de instâncias, os algoritmos propostos para solucionar problemas mono-objetivo (SI, SIC e SIC_c) foram aplicados a instâncias envolvendo 13 geradores (dois casos com demandas diferentes) e 40 geradores, todos para a situação ideal (sem perdas), e 20 geradores para a situação real (com perdas), totalizando quatro estudos de caso. No segundo grupo, os algoritmos multiobjetivo (SICMO_e, SICMO_p e SICMO_{pc}) foram aplicados a instâncias envolvendo 6 geradores para o caso sem perdas, 6 geradores (três casos com demandas ou coeficientes de perdas diferentes) 14 geradores (dois casos com demandas diferentes) e 20 geradores, todos para a situação real (com perdas), totalizando 6 estudos de caso.

Assim, o conjunto de experimentos realizados neste trabalho envolve 10 estudos de caso e pode ser resumido da seguinte forma:

- problemas mono-objetivo (Abordagens SI, SIC e SIC_c)
 - Sem perdas
 - * caso 1: 13 geradores e demanda de 1800 MW;
 - * caso 2: 13 geradores e demanda de 2520 MW;
 - * caso 3: 40 geradores e demanda de 10500 MW;
 - Com perdas

- * caso 4: 20 geradores e demanda de 2500 MW.;
- problemas biobjetivo (Abordagens $SICMO_e$, $SICMO_p$ e $SICMO_{pc}$)
 - Sem perdas
 - * caso 1: 6 geradores e demanda de 2,834 MW
 - Com perdas
 - * caso 2: 6 geradores e demanda de 2,834 MW;
 - * caso 3: 6 geradores e demanda de 2,834 MW (diferença para o caso 2 nos parâmetros do cálculo das perdas);
 - * caso 4: 6 geradores e demanda de 900 MW;
 - * caso 5: 14 geradores e demanda de 3668 MW;
 - * caso 6: 14 geradores e demanda de 4242 MW;

Em cada estudo de caso, os algoritmos propostos foram executados 50 vezes de maneira independente¹. Inicialmente as abordagens propostas foram comparadas entre si e a abordagem capaz de produzir os melhores resultados foi escolhida para ser comparada com abordagens recentes da literatura. Entre as abordagens de comparação estão trabalhos publicados nos últimos anos e que produziram resultados importantes para cada uma das instâncias consideradas. Entre as abordagens comparadas destacam-se:

- DE - abordagem baseada em Evolução Diferencial (do inglês *Differential Evolution* (DE)) proposta por Noman e Iba (2008);
- SDE - abordagem baseada em Evolução Diferencial proposta por Balamurugan e Subramanian (2007);
- IHS - abordagem baseada em Harmony Search proposta por Geem (2009).
- NSGAI - abordagem baseada em Algoritmos Genéticos proposta por Abido (2003b) para solucionar problemas multiobjetivo. Esta abordagem é bastante utilizada em trabalhos que consideram problemas de otimização multi-critério (COELLO; LAM-ONT; VELDHUIZEN, 2007).
- MOPSO - abordagem baseada em Exame de Partículas (do inglês *Particle Swarm Optimization* (PSO)) proposta por Abido (2009) para solucionar problemas multi-objetivo.

¹Em cada caso uma semente diferente do gerador de números aleatórios foi usada na geração da população inicial.

6.1 Ajustes de Parâmetros

Os experimentos iniciais foram realizados para se determinar um bom conjunto de parâmetros². Durante estes experimentos foi utilizado o algoritmo SIC e o primeiro estudo de casos. Partiu-se do pressuposto que os parâmetros que mais afetam o desempenho são: o tamanho da população, a taxa de aceitação, o número de melhores indivíduos no conhecimento situacional, a idade máxima dos clones e o número de clones. Os demais parâmetros foram mantidos fixos e os valores iniciais para os parâmetros analisados foram: tamanho da população igual a 50, taxa de aceitação igual a 0.2 (20%), número de melhores indivíduos igual a 10, idade máxima igual a 100 e número de clones igual a 4. A escolha destes valores se deve ao fato de que eles foram utilizados em experimentos preliminares durante o desenvolvimento da tese e demonstraram um bom desempenho. Durante os experimentos foi utilizado o teste de Kruskal-Wallis para determinar se houve diferença estatística entre os resultados encontrados para as diferentes configurações. Todos os testes utilizaram uma confiabilidade de 95%. As várias versões foram ajustadas para realizar o mesmo número de avaliações.

6.1.1 Ajuste do Tamanho da População

Para o tamanho da população foram analisados os valores 30, 50, 70 e 90. A Tabela 2 mostra que as médias obtidas pelas diferentes configurações foram próximas e que a configuração com 50 anticorpos mostrou-se melhor nos valores de custo mínimo, médio e máximo.

Tabela 2: Resultados para o ajuste do tamanho da população. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
Tamanho da População	Custo Min	Custo Médio	Custo Max	σ
30	17968,36	17969,36	17969,56	0,26
50	17960,36	17968,59	17969,50	2,44
70	17960,5	17969,44	17976,48	1,93
90	17968,36	17969,29	17969,78	0,38

O teste de Kruskal-Wallis foi aplicado para ver se as diferenças observadas eram estatisticamente significativas. Pode-se notar na Figura 28 que existe diferença entre o tamanho de população igual a 50 e as demais configurações, mas que não existe diferença significativa entre as demais. Portanto, nos demais experimentos o tamanho da população utilizado foi sempre 50.

²Neste trabalho um bom conjunto de parâmetro é aquele que quando utilizado permite que o algoritmo encontre valores próximos aos relatados na literatura.

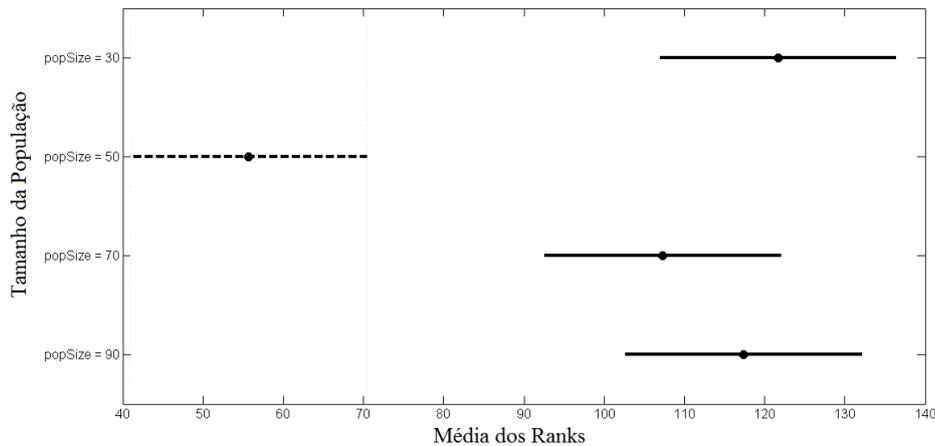


Figura 28: Médias dos *ranks* para diferentes configurações de tamanho de população.

6.1.2 Ajuste da Taxa Aceitação - $accept_{perc}$

Foram testadas configurações com taxas de aceitação de 10%, 20% e 30%. Os resultados obtidos estão na Tabela 3. Percebeu-se uma maior variabilidade das médias (sendo a configuração com taxa de 20% a que teve menor média) e que duas configurações foram capazes de encontrar o mesmo valor para o custo mínimo.

Tabela 3: Resultados para o ajuste da taxa aceitação - $accept_{perc}$. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
$accept_{perc}$	Custo Min	Custo Médio	Custo Max	σ
10%	17960,36	17969,63	17976,46	2,16
20%	17960,36	17968,59	17969,50	2,44
30%	17968,36	17970,91	18035,62	9,45

A Figura 29 mostra que a configuração com taxa de aceitação de 20% possui diferença estatística para as demais e por isto será adotado nos demais experimentos.

6.1.3 Ajuste do Número de Melhores - $pbest$

Para o número de melhores indivíduos mantidos no conhecimento situacional foram testados os valores 5, 10 e 15. Na Tabela 4 observa-se que a configuração com 10 indivíduos apresentou melhores valores para o custo mínimo, médio e máximo.

Pela Figura 30 pode-se concluir que a diferença entre as médias é estatisticamente significativa com 95% de confiabilidade. Por isto, este valor será adotado nos experimentos seguintes.

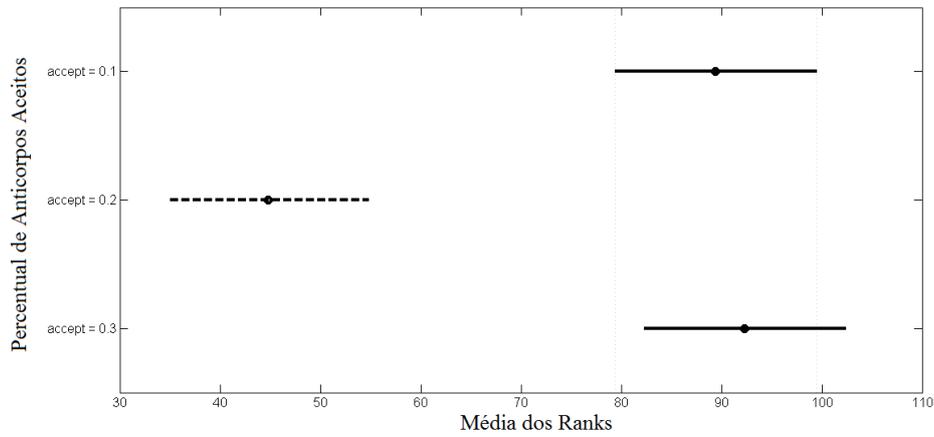


Figura 29: Médias dos *ranks* para diferentes configurações de taxa de aceitação.

Tabela 4: Resultados para o ajuste número de melhores - *pbest*. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
<i>pbest</i>	Custo Min	Custo Médio	Custo Max	σ
5	17969,35	17969,86	17976,45	1,68
10	17960,36	17968,59	17969,50	2,44
15	17968,37	17969,35	17969,57	0,29

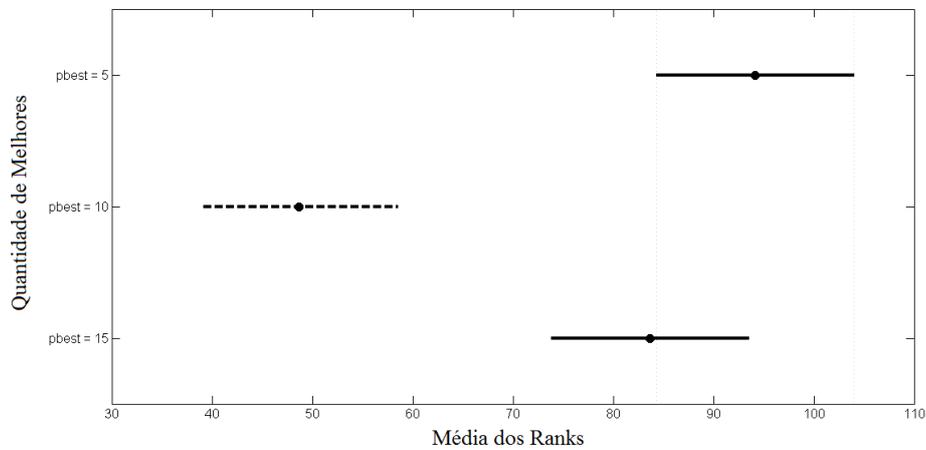


Figura 30: Médias dos *ranks* para diferentes configurações de número de anticorpos mantidos no conhecimento situacional.

6.1.4 Ajuste da Idade Máxima dos Anticorpos - τ_B

Para a idade máxima dos anticorpos foram analisados os valores 50, 100 e 150. A Tabela 5 mostra que os resultados obtidos para o valor 100 foram melhores.

Tabela 5: Resultados para o ajuste da idade máxima dos anticorpos - τ_B . Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
τ_B	Custo Min	Custo Médio	Custo Max	σ
50	17968,36	17969,34	17969,64	0,3
100	17960,36	17968,59	17969,50	2,44
150	17968,36	17969,51	17976,37	1,03

A Figura 31 permite observar que a diferença entre a configuração com idade máxima igual a 100 é significativamente diferente das demais e, portanto, será adotada nos próximos experimentos.

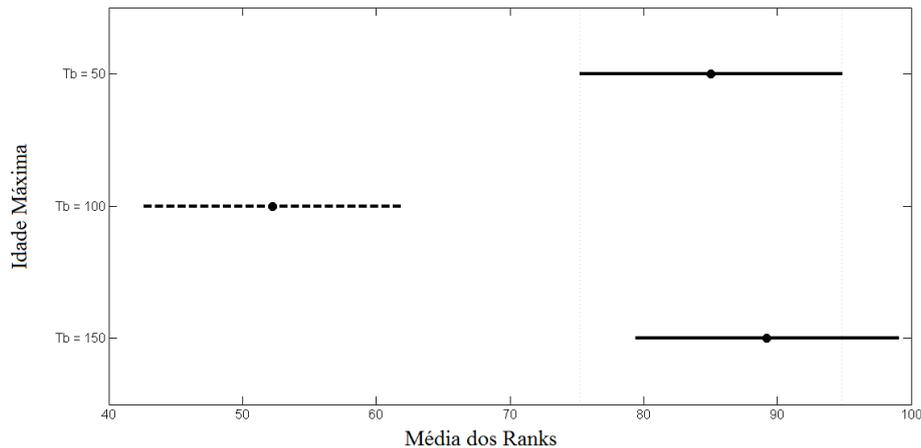


Figura 31: Médias dos *ranks* para diferentes configurações de idade máxima dos anticorpos.

6.1.5 Ajuste do Número de Clones

Foram analisadas configurações com 2, 4 e 6 clones para cada anticorpo. Os resultados estão ilustrados na Tabela 6 e na Figura 32.

Tabela 6: Resultados para o ajuste do número de clones - *dup*. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
<i>dup</i>	Custo Min	Custo Médio	Custo Max	σ
2	17968,36	17969,72	17977,23	1,82
4	17960,36	17968,59	17969,50	2,44
6	17968,36	17969,76	17976,52	1,72

De acordo com a Tabela 6 a configuração com 4 clones apresentou melhores valores para o custo mínimo, médio e máximo.

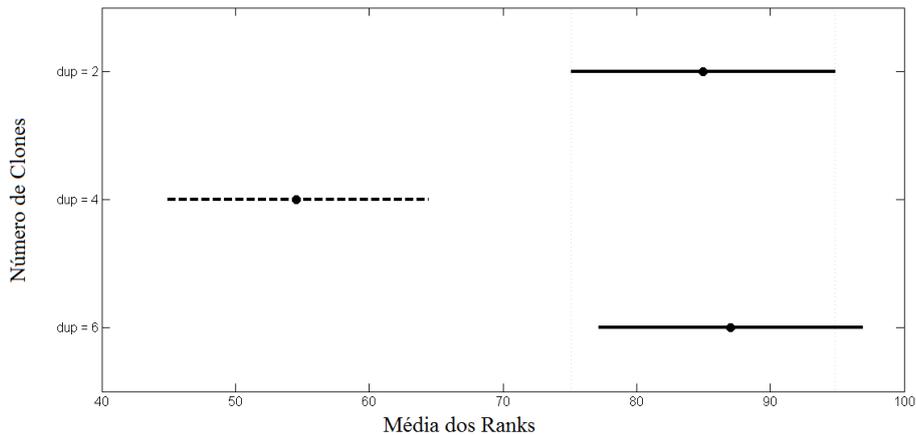


Figura 32: Médias dos *ranks* para diferentes configurações de número de clones por anticorpo.

Na Figura 32 nota-se a diferença estatística entre a configuração com 4 clones e as demais. Por este motivo adota-se este valor para o número de clones de um anticorpo nos experimentos a seguir.

6.1.6 Parâmetros Utilizados nos Experimentos

Os algoritmos propostos foram executados utilizando o conjunto de parâmetros apresentado na Tabela 7.

É importante lembrar que esse conjunto de parâmetros foi encontrado de maneira empírica para a primeira instância e foi mantido constante para todas as instâncias.

6.2 Resultados para o Despacho Econômico de Energia

Conforme salientado no início do capítulo, no grupo que considera o problema do despacho econômico de energia (problema mono-objetivo de otimização do custo), foram utilizadas quatro instâncias de teste: uma com 13 geradores e uma demanda de 1800 MW, outra com os mesmos 13 geradores mas com uma demanda de 2520 MW, uma com 40 geradores e demanda de 10500 MW e uma com 20 geradores, que considera as perdas na geração de energia, e demanda de 3668 MW. Os resultados para os quatro casos de teste são apresentados nas subseções 6.2.1, 6.2.3, 6.2.4 e 6.2.5. Os algoritmos foram implementados em Java e, nesta primeira etapa dos experimentos, foram executados em um notebook rodando Windows VistaTM com CPU Core 2 DuoTM da Intel com frequência de 1.6 GHz.

Tabela 7: Parâmetros utilizados nos algoritmos propostos.

Parâmetro	Valor
$popSize$ - tamanho da população	50
dup - número de clones gerados para cada anticorpo	4
critério de parada (número máximo de gerações)	3000
φ - taxa de decréscimo do multiplicador não-caótico	40
ρ - regula o número de mutações durante a hipermutação	4
τ_B - idade máxima que um anticorpo pode atingir antes de ser morto	100
θ - controla o balanço entre a taxa de sucesso e o $fitness$ médio de cada tipo de conhecimento na atualização da influência principal	400
$pbest$ - quantidade de melhores indivíduos armazenados no conhecimento situacional	10
TJ - número de eventos armazenados no conhecimento histórico	10
$pElite$ - percentual de vezes que as melhores regiões serão escolhidas no conhecimento topográfico	50%
$TotReg$ - quantidade de regiões consideradas entre as melhores no conhecimento topográfico	30
$intervger_{minlocal}$ - número de gerações sem melhorias no melhor indivíduo antes que o algoritmo seja considerado preso em um ótimo local, ou seja, quando um evento histórico é disparado	200
r_c - controla o comportamento do multiplicador caótico	0,99
$accept_{perc}$ - percentual de anticorpos aceitos para atualizar o espaço de crenças	0,2

6.2.1 Mono-objetivo - Estudo de Caso 1

Este estudo de caso aborda o problema descrito no Capítulo 4 Seção 4.1 e considera 13 geradores com efeito de ponto de válvula conforme mostra a Tabela 64 do Apêndice A.1. Neste caso, a demanda esperada é definida como $P_D = 1800$ MW e o objetivo é minimizar o custo de combustível dos geradores. Este problema possui muitos mínimos locais, e o ótimo global é difícil de ser determinado (COELHO; SOUZA; MARIANI, 2009).

6.2.1.1 Comparação entre as abordagens propostas

Os resultados obtidos pelos algoritmos propostos para este estudo de caso são mostrados na Tabela 8.

Os dados mostram que para este problema, o algoritmo SIC encontrou melhores valores para os custos mínimo, médio e máximo. O tempo médio de execução para cada um

Tabela 8: Resultados das abordagens propostas para o estudo de caso 1. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
Método	Custo Min	Custo Médio	Custo Máx	σ
SI	17960,41	17983,03	18074,41	24,96
SIC	17960,36	17968,59	17969,50	2,44
SIC_c	17969,34	17969,55	1976,36	0,98

dos algoritmos foi: SI = 2,14s, SIC = 2,45s e SIC_c = 2,06s.

Um teste de inferência de Kruskal-Wallis foi aplicado para determinar se a diferença nos valores das médias é significativa (GIBBONS, 1985). Conforme discutido no Capítulo 3, Seção 3.2.1, esta inferência é baseada na média do rank das soluções encontradas por cada algoritmo. Como o p -valor foi nulo, conclui-se que há pelo menos uma média significativamente diferente das demais. Para descobrir quais valores são diferentes dos demais do ponto de vista estatístico, será utilizado um gráfico com os intervalos de confiança para a média dos ranks das soluções de cada algoritmo, conforme ilustrado na Figura 33.

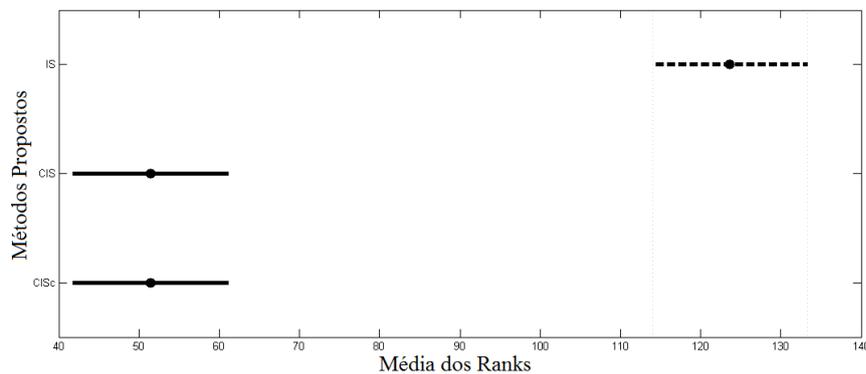


Figura 33: Resultados estatísticos para o teste Kruskal-Wallis para o caso com 13 geradores e demanda de 1800 MW, mostrando que o SIC e o SIC_c obtiveram melhores resultados do que o SI.

Na análise da Figura 33, se não há intersecção entre os valores de confiança de dois algoritmos, suas médias de custo são consideradas significativamente diferentes, caso contrário (quando há qualquer intersecção entre os intervalos), suas médias de custo são consideradas indistinguíveis.

Portanto, pela análise da Figura 33, é possível concluir que a introdução do algoritmo cultural trouxe benefícios à abordagem uma vez que tanto o SIC quanto o SIC_c possuem desempenho superior ao do SI. Outra conclusão é que a introdução do componente caótico não trouxe nenhum benefício, uma vez que não houve diferença estatística entre os custos médios da versão caótica (SIC_c) e da versão sem caos (SIC).

Com o objetivo de entender melhor o comportamento dos métodos propostos, foi calculado o intervalo com 95% de confiança para o custo médio usando o método BC_a

bootstrap (BRADLEY; TIBSHIRANI, 1994) com 10^5 reamostragens (*re-samples*). O intervalo encontrado pelo *bootstrap* para este caso foi:

- SI = [17978,269760 - 17990,287700];
- SIC = [17967,858862 - 17969,033812];
- SIC_c = [17969,416555 - 17969,983659].

Como a diferença entre duas médias é considerada significativa se não há intersecção entre os intervalos de confiança para as médias (WOOD, 2005), conclui-se que há diferença estatística entre todos os algoritmos e o SIC é superior aos outros dois algoritmos (SI e SIC_c) com respeito à média de custo de combustível. Assim, com base no método BC_a *bootstrap*, pode-se dizer inclusive que, neste estudo de caso, houve certa degradação do desempenho com a introdução do componente caótico uma vez que além da média de custo, o mínimo obtido pela versão caótica (SIC_c) foi inferior àquele obtido pela versão não-caótica (SIC).

Portanto, com base no teste anterior, conclui-se que o melhor algoritmo para este estudo de caso foi o SIC. O melhor resultado obtido pelo SIC (ou seja, o melhor indivíduo obtido ao final da evolução de uma entre as 50 simulações executadas) é dado pelo vetor $P_i, i = 1, \dots, 13$ com um custo mínimo de 17960.36 US\$, conforme mostrado na Tabela 9.

Tabela 9: Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 13 geradores e demanda de 1800 MW.

Gerador	Potência (MW)	Gerador	Potência (MW)
P_1	628,3185	P_8	109,8666
P_2	149,5997	P_9	109,8666
P_3	222,7491	P_{10}	40
P_4	60	P_{11}	40
P_5	109,8666	P_{12}	55
P_6	109,8666	P_{13}	55
P_7	109,8666	$\sum_{i=1}^{13} P_i$	17960,36

6.2.2 Análise do Algoritmo SIC

O comportamento dos conhecimentos, a influência da busca local quase-simplex e da função de influência principal proposta nesta tese são analisados a seguir.

6.2.2.1 Comportamento dos Conhecimentos no Algoritmo SIC

Para melhor compreender o funcionamento do algoritmo SIC, foram analisadas as probabilidades de aplicação da função de influência de cada conhecimento e a forma como os conhecimentos afetam a convergência do algoritmo. Pela Figura 34-a) é possível observar que os conhecimentos topográfico e situacional possuem baixa probabilidade de serem aplicados (bem próximas dos valores mínimos) enquanto os conhecimentos histórico e normativo possuem grande probabilidade de serem aplicados. Apesar das baixas probabilidades das influências dos conhecimentos situacional e topográfico é importante salientar que eles muitas vezes são capazes de melhorar o menor valor do custo encontrado.

Na Figura 34-b) observa-se que os valores de custo mínimo e médio de combustível mantêm-se próximos ao longo de todo o processo evolutivo. Isto indica duas coisas: que os conhecimentos adquiridos se espalham rapidamente e, por conseguinte, a diversidade se mantém relativamente baixa. Por um lado isto é bom porque novos conhecimentos são utilizados na melhoria da maioria da população e por outro isto é ruim porque os conhecimentos se propagam tão rapidamente que a diversidade da população se mantém relativamente baixa, mesmo com a inserção de novos anticorpos após a morte de anticorpos antigos (que não são melhorados há muitas gerações).

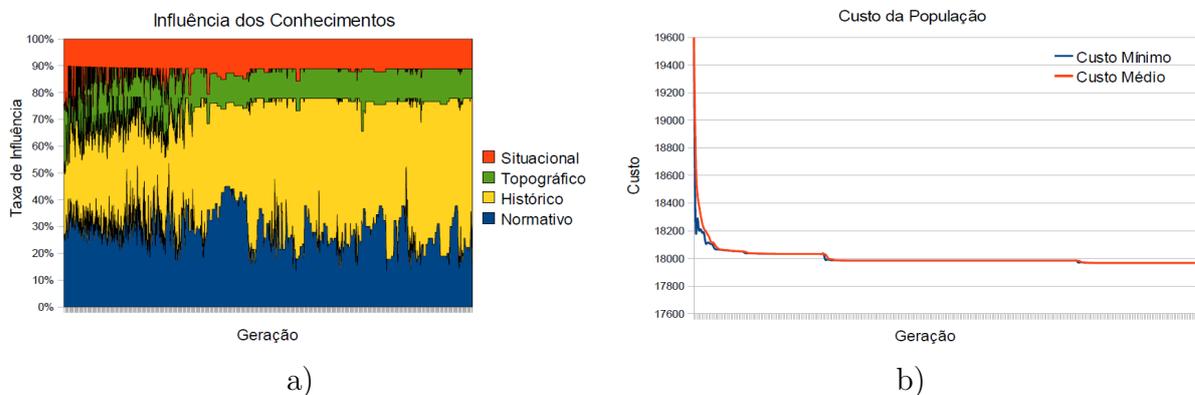


Figura 34: Comportamento do algoritmo SIC a) Probabilidade de aplicação da influência de cada tipo de conhecimento por geração b) Custo ótimo e médio por geração

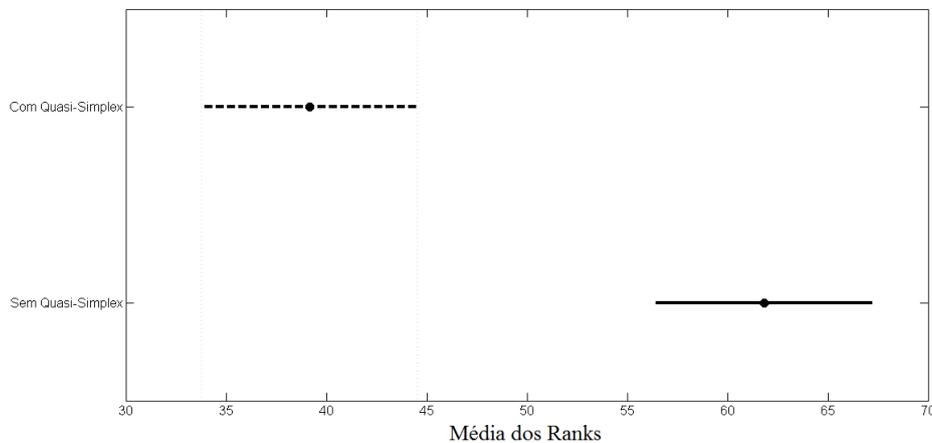
6.2.2.2 Efeito da Busca Local Quase-Simplex no Algoritmo SIC

O efeito da busca local quase-simplex no desempenho do algoritmo SIC foi analisado para este estudo de caso. Na Tabela 10 observam-se menores valores de custo mínimo, médio e máximo para a configuração com a busca local.

Tabela 10: Efeito da busca local Quase-Simplex. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
	Custo Min	Custo Médio	Custo Max	σ
Com Quase-Simplex	17960,36	17968,59	17969,50	2,44
Sem Quase-Simplex	17968,36	17969,49	17976,37	1,03

A Figura 35 permite afirmar com 95% de confiabilidade que os resultados da configuração com busca local são superiores aos resultados da configuração sem a busca local. Portanto, a busca local foi considerada importante para a qualidade dos resultados encontrados.

Figura 35: Médias dos *ranks* para as configurações com e sem a busca local.

6.2.2.3 Efeito da Função de Influência Principal Proposta no Algoritmo SIC

Uma das contribuições desta tese é a proposta de uma nova função de influência principal. Para aferir se a nova função é ou não vantajosa, seu efeito foi avaliado para este estudo de caso. Na Tabela 11 observa-se que os menores valores de custo mínimo foram obtidos com a função de influência principal proposta e com a aleatória, mas que menores valores de custo médio e máximo foram obtidos somente com a função de influência principal proposta.

Tabela 11: Efeito da função de influência principal proposta. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 1800 MW				
	Custo Min	Custo Médio	Custo Max	σ
Aleatória	17960,36	17969,37	17976,50	1,97
MVT	17968,45	18005,44	18078,64	35,33
Proposta	17960,36	17968,59	17969,50	2,44

Para se obter conclusões estatísticas sobre as diferenças de comportamento das configurações foi aplicado o teste de Kruskal-Wallis. A Figura 36 apresenta os resultados.

Pode-se notar que a função proposta é melhor que as demais e que a função aleatória foi melhor que a função baseada na Teoria do Valor Marginal (função MVT). Portanto pode-se afirmar que a função de influência principal proposta contribuiu para a obtenção de bons resultados.

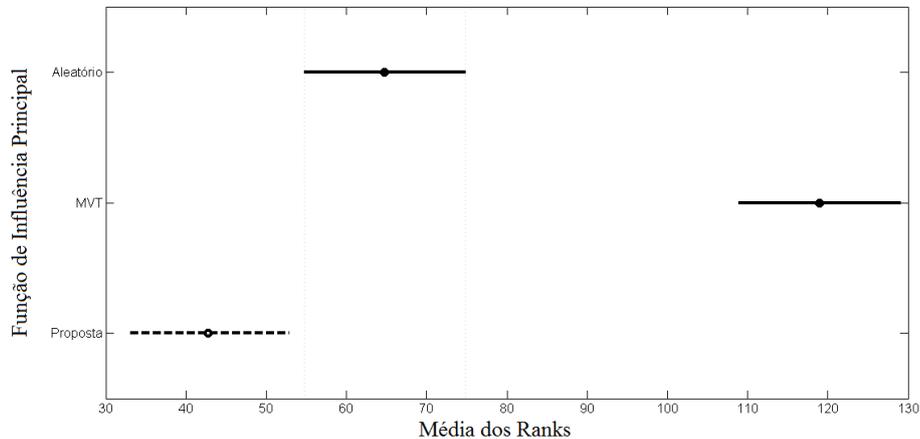


Figura 36: Médias dos *ranks* para diferentes configurações de função de influência principal.

6.2.2.4 Comparação com as abordagens da literatura

De forma a comparar o desempenho da proposta com outras abordagens apresentadas na literatura, o melhor algoritmo para este estudo de caso (SIC) foi confrontado com outras abordagens recentes da literatura, conforme mostra a Tabela 12.

Neste caso é possível observar que o algoritmo proposto (SIC) obteve melhores valores para o custo mínimo (17960.36, juntamente com IHS) e custo máximo (17969.50) do que as abordagens de comparação, e foi superado no quesito custo médio pela abordagem baseada em Busca Harmônica (Algoritmo IHS) (COELHO; MARIANI, 2009). Neste caso, não foi possível aplicar nenhum teste estatístico uma vez que os dados para obtenção da média das outras abordagens não estão disponíveis.

Portanto, para o estudo de caso 1, a Tabela 12 mostra que do ponto de vista de obtenção do menor custo, a abordagem proposta (SIC) foi superior às outras abordagens recentes publicadas na literatura, com exceção da IHS.

6.2.3 Mono-objetivo - Estudo de Caso 2

Assim como no estudo de caso anterior, esta seção aborda o problema descrito no Capítulo 4 Seção 4.1 e considera o problema de otimização de custo de combustível sem perdas para 13 geradores com efeito de ponto de válvula conforme mostra a Tabela 64 do Apêndice A.1. Só que neste caso, a demanda é definida como $P_D = 2520$ MW.

Tabela 12: Resultados da comparação com a literatura para o estudo de caso 1. Os custos são dados em US\$

Caso com 13 geradores e demanda de 1800 MW				
Método	Custo Min	Custo Médio	Custo Max	σ
SIC	17960,36	17968,59	17969,50	2,44
IGA ^a	18063,58	18096,40	18293,47	45,79
CEP ^b	18048,21	18190,32	18404,04	-
FEP ^c	18018,00	18200,79	18453,82	-
MFEP ^d	18028,09	18192,00	18416,89	-
IFEP ^e	17994,07	18127,06	18267,42	-
PSO ^f	18030,72	18205,78	-	-
PSO-SQP ^g	17969,93	18029,99	-	-
MPSO ^h	17973,34	-	-	-
DEC-SQP ⁱ	17963,94	17973,13	17984,81	1,97
HQPSO ^j	17963,95	18273,86	18633,04	123,22
HDE ^k	17975,73	18134,80	-	-
ST-HDE ^l	17963,89	18046,38	-	-
DE ^m	17963,83	17965,48	17975,36	-
CDEMD ⁿ	17961,94	17974,68	18061,41	20,30
IHS ^o	17960,36	17965,41	17971,65	16,95
PSO-TVAC ^p	17963,87	18154,56	18358,31	-
ICA-PSO ^q	17960,37	17967,94	17978,14	-
QPSO ^r	17969,01	18075,11	-	-

^a (LING et al., 2003)

^b (CHAKRABARTI; CHATTOPADHYAY, 2003)

^c (CHAKRABARTI; CHATTOPADHYAY, 2003)

^d (CHAKRABARTI; CHATTOPADHYAY, 2003)

^e (CHAKRABARTI; CHATTOPADHYAY, 2003)

^f (VICTOIRE; JEYAKUMAR, 2004)

^g (VICTOIRE; JEYAKUMAR, 2004)

^h (HOU et al., 2005)

ⁱ (COELHO; MARIANI, 2006)

^j (COELHO; MARIANI, 2008)

^k (WANG; CHIOU; LIU, 2007)

^l (WANG; CHIOU; LIU, 2007)

^m (NOMAN; IBA, 2008)

ⁿ (COELHO; SOUZA; MARIANI, 2009)

^o (COELHO; MARIANI, 2009)

^p (CHATURVEDI; PANDIT; SRIVASTAVA, 2009)

^q (VLACHOGIANNIS; LEE, 2009)

^r (MENG et al., 2010)

6.2.3.1 Comparação entre as abordagens propostas

Os resultados obtidos pelos algoritmos propostos para este estudo de caso são mostrados na Tabela 13.

Tabela 13: Resultados das abordagens propostas para o estudo de caso 2. Os custos são dados em US\$

Caso com 13 geradores e demanda esperada de 2520 MW				
Método	Custo Min	Custo Médio	Custo Max	σ
SI	24164,80	24223,22	24427,29	52,74
SIC	24164,05	24164,05	24164,13	0,01
SIC _c	24164,05	24164,05	24164,05	0,00

Os resultados da Tabela 13 mostram que tanto o algoritmo SIC quanto o SIC_c obtiveram melhores resultados do que o SI para todos os indicadores. Entre as duas abordagens culturais, as diferenças se resumem ao valor máximo e ao desvio padrão (onde o SIC_c é ligeiramente superior). Neste estudo de caso o SI leva em média 1.95s para obter estes resultados enquanto que o SIC e o SIC_c levam 2.16s and 1.83s, respectivamente.

O teste de inferência de Kruskal-Wallis foi aplicado para determinar se as diferenças nas médias são significativas. O p -valor foi novamente nulo indicando que ao menos um valor de média é diferente dos outros. A Figura 37 mostra que o SI é inferior às versões culturais (SIC e SIC_c). Com isso, mostra-se mais uma vez a vantagem da introdução dos algoritmos culturais na proposta. Somente que, diferente do estudo de caso anterior, neste caso a versão caótica SIC_c se mostrou superior à versão sem caos com respeito ao valor de custo médio, o que representa um resultado de certa forma surpreendente dada a pequena diferença entre os valores de média obtidos.

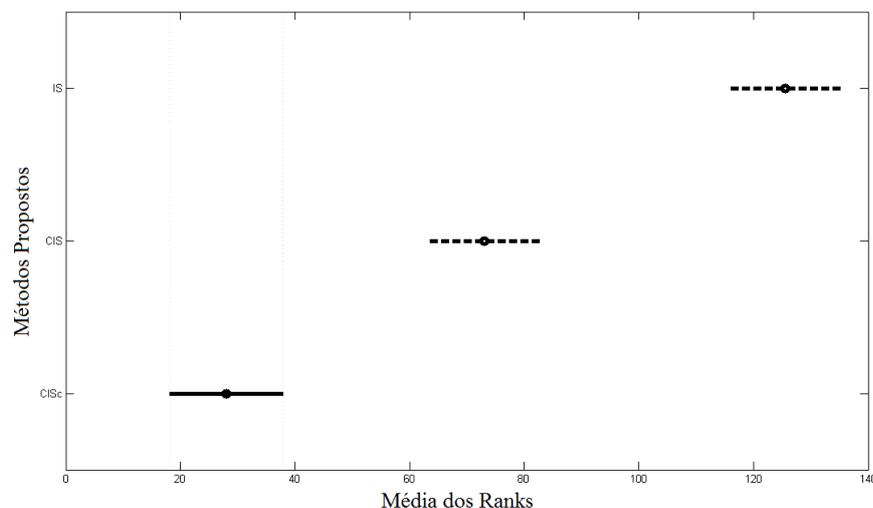


Figura 37: Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 13 geradores e demanda de 2520 MW, mostrando que o SIC_c é melhor do que o SIC e o SI.

A aplicação do método *bootstrap* (BRADLEY; TIBSHIRANI, 1994), com 10^5 reamostragens, ao caso de 13 geradores com demanda de 2520 MW resultou num intervalo de confiança de 95% para a média do custo de combustível dada por

- SI = [24213,441020 - 24238,073320];
- SIC = [24164,057312 - 24164,066267];
- SIC_c = [24164,050830 - 24164,050830].

Estes valores confirmam as conclusões obtidas pelo teste de Kruskal-Wallis, ou seja, que as versões culturais dos algoritmos propostos (SIC and SIC_c) são melhores do que a versão não-cultural (SI). Neste teste, assim como no teste de Kruskal-Wallis, a superioridade da versão caótica em relação à versão sem caos é confirmada.

A Tabela 14 mostra a melhor solução (codificada no vetor $P_i, i = 1, \dots, 13$) encontrada pelo algoritmo (SIC) que resultou num custo mínimo de 24164.05 US\$.

Tabela 14: Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 13 geradores e demanda de 2520 MW.

Gerador	Potência (MW)	Gerador	Potência (MW)
P_1	628,3185	P_8	159,7331
P_2	299,1993	P_9	159,7331
P_3	294,4839	P_{10}	77,3999
P_4	159,7331	P_{11}	77,3999
P_5	159,7331	P_{12}	92,3999
P_6	159,7331	P_{13}	92,3999
P_7	159,7331	$\sum_{i=1}^{13} P_i$	24164,05

6.2.3.2 Comparação com as abordagens da literatura

Novamente foi feita uma comparação da melhor abordagem proposta com algumas abordagens recentes da literatura e os resultados estão na Tabela 15.

Tabela 15: Resultados da comparação com a literatura para o estudo de caso 2. Os custos são dados em US\$

Caso com 13 geradores e demanda de 2520 MW				
Método	Custo Min	Custo Médio	Custo Máx	σ
SIC _c	24164.05	24164.05	24164.05	0.00
SDE (BALAMURUGAN; SUBRAMANIAN, 2007)	24164.05	24168,28	24200,05	-
DTSA (KHAMSAWANG; POTHIIYA; BOONSENG, 2004)	24169,95	-	-	-
DE (NOMAN; IBA, 2008)	24169,91	24169,91	24169,91	-

Para este caso, a abordagem proposta SIC_c obteve os melhores valores para os três indicadores de custo: mínimo (21164,05), médio (21164,05) e máximo (21164,05) com desvio padrão nulo (0,00), indicando que todas as rodadas conseguiram atingir o valor mínimo.

Portanto, para o estudo de caso de 13 geradores com demanda de 2520 *MW*, a Tabela 15 mostra que a versão proposta SIC_c obteve resultados comparativamente menores do que os recentes estudos publicados na literatura. Cabe salientar que os valores obtidos pelo algoritmo SIC também são melhores do que os reportados na literatura.

6.2.4 Mono-objetivo - Estudo de Caso 3

Da mesma forma dos dois casos anteriores, esta seção considera o problema da otimização do custo de combustível do problema sem perdas com efeito de ponto de válvula (ver Capítulo 4, Seção 4.1). Somente que neste caso a dimensão do problema foi aumentada para 40 geradores conforme mostra a Tabela 65 do Apêndice A.1. Neste caso, a demanda esperada é definida como $P_D = 10500$ *MW*.

6.2.4.1 Comparação entre as abordagens propostas

Os resultados obtidos pelos algoritmos propostos para este estudo de caso são mostrados na Tabela 16.

Tabela 16: Resultados das abordagens propostas para o estudo de caso 3. Os custos são dados em US\$

Caso com 40 geradores e demanda esperada de 10500 <i>MW</i>				
Método	Custo Min	Custo Médio	Custo Máx	σ
SI	121500,43	121880,13	122283,55	237,08
SIC	121411,97	121584,64	121751,75	108,13
SIC_c	121452,90	121662,05	121833,63	119,36

Os resultados da Tabela 16 mostram que o SIC obteve melhores valores do que o SI e o SIC_c para todos os indicadores e que novamente, as versões culturais são melhores do que a versão pura (sistema imune SI). A média do tempo de execução dos algoritmos é dada por SI = 40,36s, SIC = 43,74s and SIC_c = 35,72s.

O teste de Kruskal-Wallis resultou num p -valor de $1,78 * 10e - 11$ indicando que ao menos uma das médias é estatisticamente diferente das demais. A Figura 38 mostra que a versão imune pura SI é pior do que as duas culturais e que o SIC e o SIC_c são estatisticamente equivalentes com respeito ao custo médio.

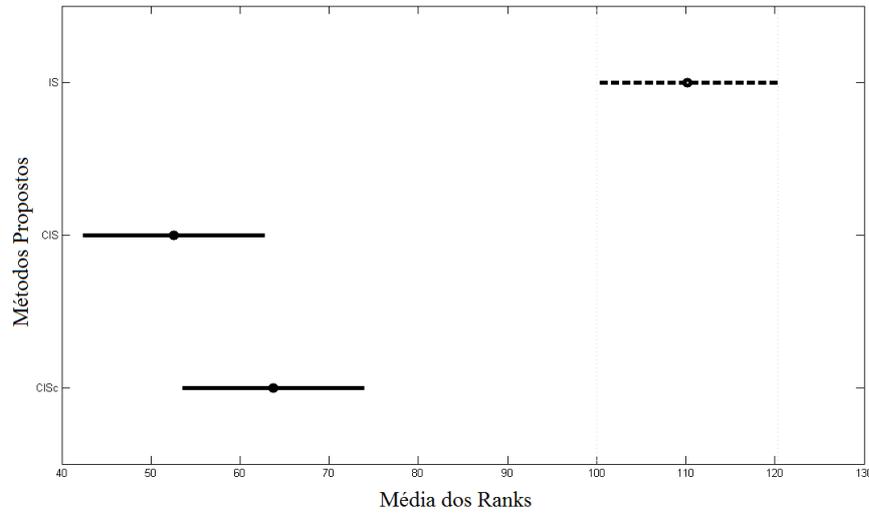


Figura 38: Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 40 geradores e demanda de 10500 MW, mostrando que o SIC e o SIC_c obtiveram melhores resultados do que o SI.

Neste caso também foi aplicado o intervalo com 95% de confiança para o custo médio usando o método BC_a *bootstrap* (BRADLEY; TIBSHIRANI, 1994) com 10^5 reamostragens. Os intervalos encontrados para o caso de 40 geradores e demanda de 10500 MW foram:

- SI = [121824,057860 - 121934,583120];
- SIC = [121561,444054 - 121610,877282];
- SIC_c = [121634,953296 - 121689,208582].

Estes valores confirmam que as versões culturais são capazes, em média, de alcançar melhores resultados do que a versão imune pura (SI). É possível concluir também que o SIC é melhor do que o SIC_c (uma vez que não há intersecção entre os intervalos de confiança associados às duas abordagens).

O melhor resultado obtido pelo SIC resultou num custo mínimo de 121411.97 US\$ e é codificado pelo vetor $(P_i, i = 1, \dots, 40)$ conforme mostra a Tabela 17.

6.2.4.2 Comparação com as abordagens da literatura

Novamente, o melhor algoritmo para este estudo de caso (SIC) foi confrontado com outras abordagens recentes da literatura, conforme mostra a Tabela 18.

Tabela 17: Melhores resultados (entre 50 rodadas) obtido pelo SIC no caso de 40 geradores e demanda de 10500 MW.

Gerador	Potência (MW)	Gerador	Potência (MW)
P_1	110,8939	P_{21}	523,2793
P_2	110,8052	P_{22}	523,2793
P_3	97,4040	P_{23}	523,2793
P_4	179,7337	P_{24}	523,2811
P_5	97	P_{25}	523,2793
P_6	140	P_{26}	523,2793
P_7	259,6034	P_{27}	10
P_8	284,6037	P_{28}	10
P_9	284,6021	P_{29}	10
P_{10}	130	P_{30}	87,8430
P_{11}	168,7998	P_{31}	190
P_{12}	94	P_{32}	190
P_{13}	214,7597	P_{33}	190
P_{14}	394,2767	P_{34}	164,7998
P_{15}	304,5194	P_{35}	200
P_{16}	394,2793	P_{36}	200
P_{17}	489,2793	P_{37}	110
P_{18}	489,2793	P_{38}	110
P_{19}	511,2793	P_{39}	110
P_{20}	511,2793	P_{40}	511,2800
		$\sum_{i=1}^{40} P_i$	121411,97

Tabela 18: Resultados da comparação com a literatura para o estudo de caso 3. Os custos são dados em US\$

Caso com 40 geradores e demanda de 10500 MW				
Método	Custo Min	Custo Médio	Custo Max	σ
SIC	121411,97	121584,64	121751,75	108,13
DEC-SQP ^a	121741,97	122295,12	122839,29	386,18
NPSO-LRS ^b	121664,43	122209,31	122981,59	-
CEP-PSO ^c	123670,00	124145,60	124900,00	-
CEP ^d	123488,29	124793,48	126902,89	-
FEP ^d	122679,71	124119,37	127245,59	-
MFEP ^d	122647,57	123489,74	124356,47	-
IFEP ^d	122624,35	123382,00	125740,63	-
PSO ^e	123930,45	124154,49	-	-
PSO-SQP ^e	122094,67	122245,25	-	-
DEvol ^f	121412,91	121430,00	121464,40	-
HDE ^g	121813,26	122705,66	-	-
ST-HDE ^g	121698,51	122304,30	-	-
DE ^h	121416,29	121422,72	121431,47	-
CDEMD ⁱ	121423,40	121526,73	121696,98	54,86
CSO ^j	121461,67	121936,19	122844,53	-
BBO ^k	121426,95	121508,03	121688,66	-
ICA-PSO ^l	121413,20	121428,14	121453,56	-
QPSO ^m	121448,21	122225,07	-	-

^a (COELHO; MARIANI, 2006)

^b (SELVAKUMAR; THANUSHKODI, 2007)

^c (SINHA; PURKAYASTHA, 2004)

^d (CHAKRABARTI; CHATTOPADHYAY, 2003)

^e (VICTOIRE; JEYAKUMAR, 2004)

^f (PEREZ-GUERRERO; CEDENIO-MALDONADO, 2005)

^g (WANG; CHIOU; LIU, 2007)

^h (NOMAN; IBA, 2008)

ⁱ (COELHO; SOUZA; MARIANI, 2009)

^j (SELVAKUMAR; THANUSHKODI, 2009)

^k (BHATTACHARYA; CHATTOPADHYAY, 2010)

^l (VLACHOGIANNIS; LEE, 2009)

^m (MENG et al., 2010)

A Tabela 18 mostra que o SIC obteve o melhor valor para o custo mínimo de combustível (121411,97), enquanto que a abordagem baseada em Evolução Diferencial DE (NOMAN; IBA, 2008) obteve os melhores valores de custo médio e máximo (121422,72) e (121431,47), respectivamente. Cabe destacar que a abordagem DE (NOMAN; IBA, 2008), assim como as abordagens propostas, também faz uso de reparação dos indivíduos ineficazes, enquanto a maioria dos algoritmos comparados utiliza alguma forma de penalização.

Portanto, para o estudo de caso mono-objetivo sem perdas com 40 geradores e demanda de 10500 MW a Tabela 18 mostra a superioridade da melhor abordagem proposta para este caso (SIC) em termos de custo mínimo quando comparada com abordagens recentes publicadas na literatura.

6.2.5 Mono-objetivo - Estudo de Caso 4

Este estudo de caso considera 20 unidades geradoras de energia. Este sistema supre uma demanda total de $P_D = 2500$ MW. Diferente dos 3 estudos de casos anteriores, aqui as perdas de energia são consideradas aproximando mais o problema da realidade. Os dados para esta instância estão disponíveis na Tabela 66 no Apêndice A.1. A matriz dos coeficientes das perdas nas linhas de transmissão são apresentados na Tabela 67 também no Apêndice A.1.

6.2.5.1 Comparação entre as abordagens propostas

Os valores de custo mínimo, médio, máximo e o desvio padrão encontrados pelos métodos propostos são apresentados na Tabela 19.

Tabela 19: Mono-objetivo - Estudo de Caso 4 - Custos em US\$

Caso com 20 geradores e Demanda de 2500MW				
Método	Custo Mínimo	Custo Médio	Custo Máximo	σ
SI	62456,6350	62456,6391	62456,6482	0,0025
SIC	62456,6330	62456,6330	62456,6330	0,0000
SIC _c	62456,6345	62456,6486	62456,7003	0,0138

Os resultados mostram que SIC obteve melhores resultados que SI e SIC_c e que SI é melhor que SIC_c. Os tempos médios para a execução de cada algoritmo são os seguintes: SI = 14,58s, SIC = 12,93s e SIC_c = 5,56s.

O teste de inferência Kruskal-Wallis resultou em um p-valor = 0,000 indicando que existe ao menos um valor de média diferente dos outros. A Figura 39 mostra que SIC é melhor que SI e SIC_c com relação ao valor de custo médio.

Para entender melhor o comportamento dos métodos propostos foram calculados os intervalos de confiança para as médias dos custos usando o método *BC_a bootstrap* (BRADLEY; TIBSHIRANI, 1994) com 10^5 reamostragens e confiabilidade de 95%. Os intervalos encontrados para o caso com 20 geradores e demanda de 2500 MW, considerando as perdas na geração de energia foram:

- SI = [62456,638 - 62456,639];
- SIC = [62456,633 - 62456,633];
- SIC_c = [62456,645 - 62456,652].

Os valores mostram que para este caso de teste o algoritmo SIC apresenta resultados estatisticamente melhores do que os demais, enquanto o algoritmo SI é estatisticamente melhor que o SIC_c.

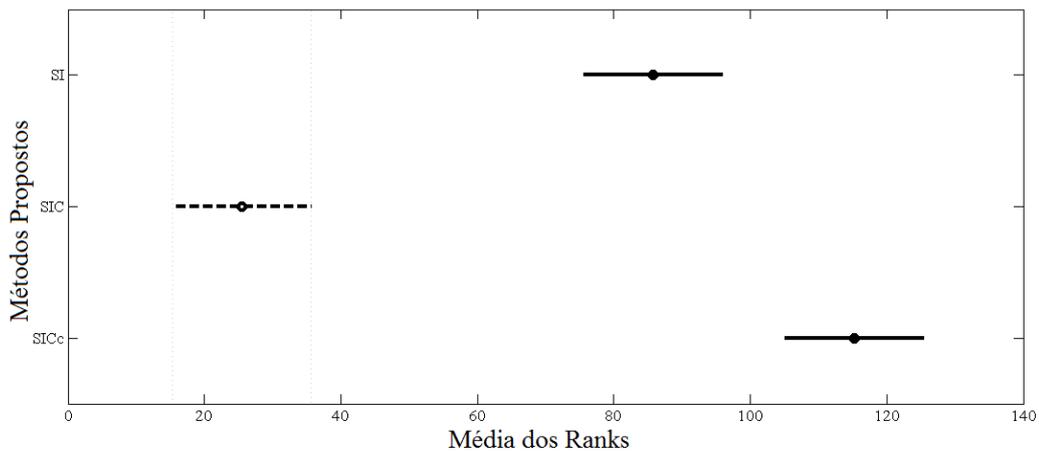


Figura 39: Resultados Estatísticos para o teste Kruskal-Wallis para o caso com 20 geradores e demanda de 2500 MW, mostrando que o SIC obteve melhores resultados.

O melhor resultado foi obtido pelo algoritmo SIC o qual encontrou um custo mínimo igual a 62456,6330. Este resultado é apresentado na Tabela 20 codificado por um vetor $(P_i, i = 1, \dots, 20)$.

6.2.5.2 Comparação com as abordagens da literatura

Para esta instância o algoritmo proposto que apresentou melhor resultado foi o SIC. Portanto, a Tabela 21 apresenta os resultados desse algoritmo comparados com aqueles

Tabela 20: Melhor Resultado para o caso com 20 geradores, considerando perdas, e demanda de 2500 MW usando o SIC.

Gerador	Potência (MW)	Gerador	Potência (MW)
P_1	512,7818	P_{11}	150,2385
P_2	169,1015	P_{12}	292,7657
P_3	126,8907	P_{13}	119,1141
P_4	102,8672	P_{14}	30,8314
P_5	113,6829	P_{15}	115,8058
P_6	73,5720	P_{16}	36,2544
P_7	115,2900	P_{17}	66,8592
P_8	116,3999	P_{18}	87,9712
P_9	100,4050	P_{19}	100,8027
P_{10}	106,0273	P_{20}	54,3048
Perdas	91,9666	$\sum_{j=1}^{20} P_i$	2591,9666

reportados na literatura. Os algoritmos utilizados para comparação são HNN (SU; LIN, 2000), λ -iteration (WOOD; WOLLENBERG, 1996) e BBO (BHATTACHARYA; CHATTOPADHYAY, 2010). Conforme pode-se observar SIC é capaz de encontrar valores para o custo mínimo inferiores aos demais algoritmos.

Tabela 21: Mono-objetivo - Estudo de Caso 4 - Comparação com a Literatura

Caso com 20 geradores e Demanda de 2500 MW				
Método	Custo Mínimo	Custo Médio	Custo Máximo	σ
SIC	62456,6330	62456,6330	62456,6330	0,0000
HNN	62456,6341	-	-	-
λ -iteration	62456,6391	-	-	-
BBO	62456,7926	62456,7928	62456,7935	-

6.2.6 Conclusões sobre os Estudos de Caso Mono-objetivo

A combinação entre algoritmos culturais e sistemas imunológicos artificiais mostrou-se eficiente na resolução dos estudos de caso do problema do despacho econômico de energia considerados. Em todos os estudos de caso que não consideram perdas de energia observou-se a supremacia dos resultados das versões culturais (SIC e SIC_c) sobre a versão puramente imune (SI). Para o estudo de caso onde as perdas de energia foram consideradas, a versão SIC mostrou-se melhor que o algoritmo SI, mas este obteve melhores resultados que o algoritmos SIC_c para as médias. Estas observações estão de acordo com os testes estatísticos realizados e com os intervalos de confiança obtidos para as médias dos custos. Além disto, os intervalos de confiança podem ser futuramente utilizados para a comparação de outros métodos com os métodos propostos, devido a equivalência entre intervalos de confiança e testes estatísticos.

Observou-se que, para o primeiro estudo de caso, os conhecimentos histórico e normativo possuem maior probabilidade de serem aplicados e que os conhecimentos se propagam muito rapidamente, reduzindo a diversidade da população. Para este estudo de caso, pode-se verificar que a busca local quase-simplex contribuiu para os resultados obtidos e que a função de influência principal proposta mostrou-se mais eficiente que duas das funções reportadas na literatura.

Outra verificação importante dos experimentos realizados, é que os métodos de reparação adotados mostraram-se eficientes, sendo sempre capazes de reparar as soluções ineficazes e permitindo que boas soluções fossem encontradas.

Na comparação com a literatura, o algoritmo SIC obteve os melhores valores para o custo mínimo em todos os estudos de caso e obteve a melhor média no estudo de caso 2 e menor valor máximo de custo no estudo de caso 1. O algoritmo SIC_c obteve os melhores valores em todos os indicadores para o estudo de caso 2. Por estes resultados, pode-se considerar o algoritmo SIC entre os algoritmos do estado-da-arte para o problema do despacho econômico de energia.

6.3 Resultados para o Despacho Econômico/Ambiental de Energia

Para validar os algoritmos Imuno-Culturais (SICMO_e, SICMO_p e SICMO_{pc}) propostos para o problema multiobjetivo do Despacho Econômico/Ambiental foram utilizadas seis instâncias de teste. Os resultados para os seis casos de teste são apresentados nas Subseções 6.3.1 a 6.3.6. Além disso, para comparação das fronteiras obtidas pelos algoritmos propostos foi utilizado o algoritmo NSGA-II implementado na plataforma jMetal (DURILLO et al., 2006), disponível no endereço <http://jmetal.sourceforge.net/>.

Os parâmetros básicos dos algoritmos Imuno-Culturais para este problema foram os mesmos utilizados no Despacho Econômico de Energia, os quais foram listados na Tabela 7. A versão escalarizada (SICMO_e) foi executada com 100 fatores de escalarização entre 0 e 1 com espaçamento uniforme e critério de parada de 50 gerações por fator de escalarização. As versões que fazem uso do arquivamento por grade adaptativa (SICMO_p e SICMO_{pc}) para armazenamento das soluções não-dominadas utilizam os seguintes parâmetros: número máximo de soluções armazenadas = 100, número de bissecções por objetivo = 10 e critério de parada = 300 gerações. A versão caótica (SICMO_{pc}) utiliza o valor 3,6 como parâmetro do mapa logístico.

Para o NSGA-II foram utilizados os seguintes parâmetros: tamanho da população igual a 100, *crossover* do tipo SBX (*Simulated Binary Crossover*) com probabilidade

de 0,90 e índice de distribuição igual a 20, mutação polinomial com probabilidade de 1,0/número de geradores e índice de distribuição igual a 20, seleção por torneio binário. Estes parâmetros foram baseados nos parâmetros utilizados em King, Rughooputh e Deb (2005). O critério de parada foi definido como sendo igual a 1000000 avaliações para garantir que o NSGA-II realize mais avaliações que os algoritmos propostos neste trabalho.

Os algoritmos foram implementados em linguagem Java e, nesta segunda etapa dos experimentos, foram executados em um desktop rodando o sistema operacional Ubuntu com CPU Core 2 DuoTM da Intel com frequência de 1.6 GHz. As execuções foram repetidas 50 vezes de maneira independente.

A metodologia de teste adotada foi inicialmente comparar os resultados obtidos pelos algoritmos Imune Culturais propostos com os resultados obtidos pelo NSGA-II utilizando a implementação do jMetal. Nesta comparação inicial, primeiro são comparados os valores extremos para os objetivos custo e emissão e depois são executados um teste de ranking de dominância e três indicadores: hipervolume, épsilon unário e R_2 . Para concluir se há diferença estatística entre os valores encontrados pelos diferentes algoritmos nos diferentes indicadores é utilizado o teste não-paramétrico de Kruskal-Wallis (p-valores inferiores à 0.05 indicam uma diferença estatística com confiabilidade de 95%). A seguir, o melhor dos algoritmos propostos é comparado com os algoritmos da literatura. Nestes casos só é possível comparar os valores extremos encontrados com os valores extremos reportados na literatura, porque as fronteiras de Pareto não são publicadas.

6.3.1 Multiobjetivo - Estudo de Caso 1

As simulações foram realizadas utilizando a instância com 6 geradores do padrão IEEE e uma demanda de 2,834 MW (KING; RUGHOOPTH; DEB, 2005). Os coeficientes de custo de combustível e emissão de NO_x são apresentados nas Tabelas 68 e 69, respectivamente, do Apêndice A.2.

A Tabela 22 apresenta os melhores resultados obtidos pelos algoritmos propostos e pelo NSGA-II para o objetivo custo. Pode-se observar que para este objetivo o algoritmo SICMO_e obteve o melhor resultado, dominando os valores do NSGA-II e do SICMO_{pc} .

Tabela 22: Multiobjetivo - Estudo de Caso 1 - Solução com o melhor valor para o custo de combustível

	SICMO_e	SICMO_p	SICMO_{pc}	NSGA-II
Custo	600,111	600,135	600,123	600,119
Emissão	0,22214	0,22158	0,22295	0,22251

A Tabela 23 apresenta os melhores resultados obtidos pelos algoritmos propostos e pelo NSGA-II para o objetivo emissão. Para este objetivo o algoritmo NSGA-II obteve o melhor resultado.

Tabela 23: Multiobjetivo - Estudo de Caso 1 - Solução com o melhor valor para a emissão de poluentes

	SICMO _e	SICMO _p	SICMO _{pc}	NSGA-II
Emissão	0,19420	0,19431	0,19435	0,19417
Custo	638,273	640,888	636,846	650,856

O rank de dominância foi inconclusivo com relação a qual dos quatro algoritmo é melhor (nenhum algoritmo dominou os demais). Os resultados (p-valores) obtidos pelo teste Kruskal-Wallis para os três indicadores de qualidade são apresentados nas Tabelas 24, 25 e 26. Valores menores que 0,05 indicam que o algoritmo da linha é significativamente melhor do que o algoritmo da coluna.

Tabela 24: Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	SICMO _{pc}	NSGA-II
SICMO _e	-	1,9e⁻⁶⁶	6,4e⁻⁷³	1,2e⁻²²
SICMO _p	1,0	-	4,4e⁻³	1,0
SICMO _{pc}	1,0	0,99	-	1,0
NSGA-II	1,0	2,1e⁻³⁵	2,5e⁻⁴³	-

Tabela 25: Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	SICMO _{pc}	NSGA-II
SICMO _e	-	6,4e⁻⁶⁰	1,0e⁻⁶⁶	1,5e⁻²¹
SICMO _p	1,0	-	4,0e⁻³	1,0
SICMO _{pc}	1,0	0,99	-	1,0
NSGA-II	1,0	7,5e⁻²⁹	7,8e⁻³⁷	-

Tabela 26: Multiobjetivo - Estudo de Caso 1 - P-valores para o indicador R₂.

	SICMO _e	SICMO _p	SICMO _{pc}	NSGA-II
SICMO _e	-	9,2e⁻⁶⁷	1,4e⁻⁷¹	2,2e⁻²²
SICMO _p	1,0	-	2,6e⁻³	1,0
SICMO _{pc}	1,0	0,97	-	1,0
NSGA-II	1,0	4,7e⁻³⁶	6,6e⁻⁴²	-

Com o resultado dos testes estatísticos pode-se concluir que em termos de qualidade da fronteira de Pareto obtida o algoritmo SICMO_e é superior ao NSGA-II que é melhor que o SICMO_p que por sua vez é melhor que o SICMO_{pc} (isto para todos os indicadores de qualidade). Aqui cabe salientar que o tempo médio de execução do algoritmo SICMO_e é 102,91s, do NSGA-II é 54,20s, do SICMO_p é 15,08s e do SICMO_{pc} é 24,55s. O SICMO_p mostrou-se mais rápido que o SICMO_{pc}, o qual é duas vezes mais rápido que o NSGA-II

e o $SICMO_e$ é o algoritmo mais lento. As simulações realizadas nas instâncias seguintes não irão levar em consideração o algoritmo $SICMO_{pc}$, devido ao seu desempenho ser estatisticamente inferior aos demais algoritmos.

A Figura 40 apresenta um exemplo de fronteira de Pareto obtida por cada um dos algoritmos propostos juntamente com a fronteira obtida pelo NSGA-II, onde pode-se observar a supremacia do algoritmo $SICMO_e$ sobre os demais no quesito qualidade das soluções geradas.

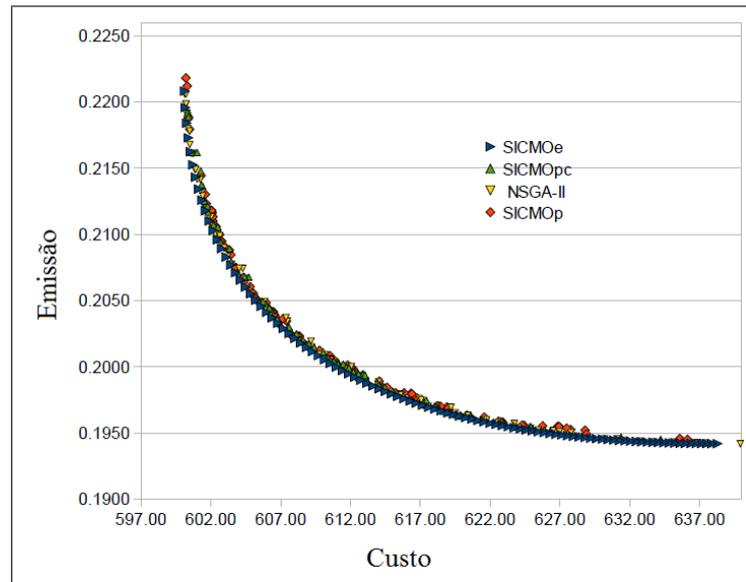


Figura 40: Exemplos de conjuntos de aproximação das fronteiras de Pareto obtidos pelos algoritmos para a instância com 6 geradores e demanda de 2,834 MW.

6.3.1.1 Influência dos Conhecimentos nos Algoritmos Imuno-Culturais

Primeiramente foi analisada a influência dos conhecimentos no algoritmo $SICMO_e$. A Figura 41 mostra as probabilidades de aplicação de cada influência ao longo do processo evolutivo³. Observa-se que as probabilidades são próximas para todos os conhecimentos, diferentemente do que ocorreu para o caso mono-objetivo. Isto provavelmente se deve as diferentes características de cada instância e não aos diferentes valores de escalarização, visto que as probabilidades mantiveram-se quase constantes ao longo de todo o processo evolutivo.

Depois foi analisada a influência dos conhecimentos no algoritmo $SICMO_p$. A Figura 42-a) apresenta as probabilidades de aplicação da influência de cada tipo de conhecimento para o objetivo custo. Pode-se observar que as probabilidades são praticamente as mesmas, com leves flutuações; ou seja, ao contrário do que ocorria no caso mono-objetivo,

³O vetor de escalarização é trocado a cada 50 gerações.

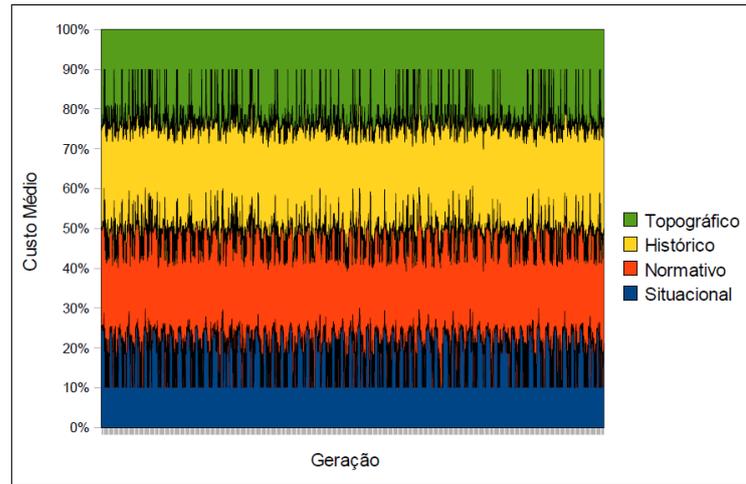


Figura 41: Probabilidade da aplicação das funções de influência dos diferentes conhecimentos no SICMO_e.

na versão multiobjetivo todos os conhecimentos possuem quase a mesma importância na resolução do problema. Comportamento semelhante é visto na Figura 42-b) para a aplicação das influências dos conhecimentos baseados no objetivo emissão. Uma explicação plausível para as probabilidades parecidas para cada tipo de conhecimento é que em um problema de otimização multiobjetivo várias regiões do espaço de busca devem ser exploradas. Outro fator que pode ter influenciado este comportamento é o fato de que o método de seleção do algoritmo SICMO_p favorece a seleção dos clones hipermutados.

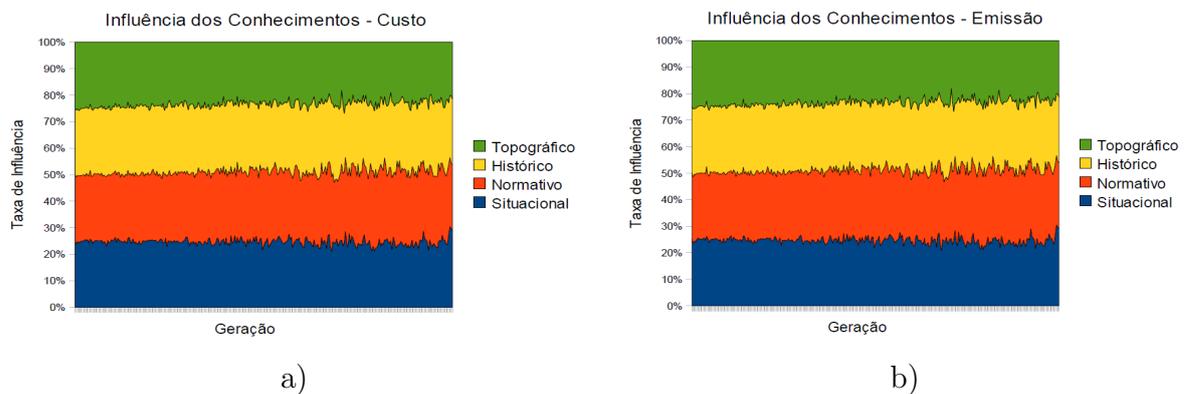


Figura 42: Probabilidade da aplicação das funções de influência dos diferentes conhecimentos no SICMO_p. a) Conhecimentos baseados em custo. b) Conhecimentos baseados em emissão.

Contudo, como em ambos os algoritmos a influência dos conhecimentos é parecida, acredita-se que a diferença observada entre o comportamento dos conhecimentos no problema do despacho econômico de energia e o comportamento dos conhecimentos no problema do despacho econômico/ambiental de energia se deva às diferentes características das instâncias, ou seja, acredita-se que a função de influência principal proposta seja capaz de se adaptar às diferentes características de cada instância.

Um fato importante que foi observado na Figura 43 é que os valores mínimos tanto para o objetivo custo quanto para o objetivo emissão foram encontrados no início do processo evolutivo. Isto indica que o algoritmo possui dificuldades em melhorar os valores mínimos para cada um dos objetivos. Outro fato que pode ser observado é a grande variabilidade das médias para os objetivos o que, de certo modo, indica um bom nível de diversidade da população.

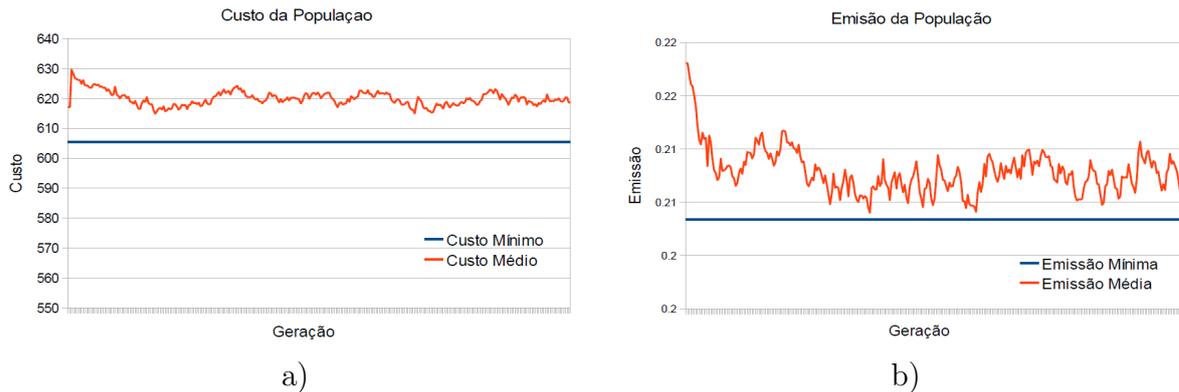


Figura 43: Valores mínimos e médios para os objetivos custo e emissão. a)Objetivo custo. b)Objetivo emissão.

6.3.1.2 Comparação com os Algoritmos da Literatura

Os algoritmos utilizados na comparação com os resultados da literatura são: LP (FARAG; AL-BAIYAT; CHENG, 1995), MOSST (DAS; PATVARDHAN, 1998), NSGA (ABIDO, 2003b), NPGA (ABIDO, 2003a), SPEA (ABIDO, 2003), NSGA-II (KING; RUGHOOPUTH; DEB, 2005), MOPSO (ABIDO, 2009) e PCEED (CETINKAYA, 2009).

A Tabela 27 apresenta os resultados obtidos pelo melhor algoritmo proposto ($SICMO_e$) e pelos algoritmos descritos na literatura para o objetivo custo. Pode-se observar que para este objetivo o algoritmo $SICMO_e$ obteve o melhor resultado.

A Tabela 28 apresenta os resultados obtidos pelo melhor algoritmo proposto e pelos algoritmos descritos na literatura para o objetivo emissão. Para este objetivo o algoritmo MOSST obteve o melhores resultado.

Cabe salientar que não foi possível realizar testes estatísticos comparando as fronteiras de Pareto obtidas pelos algoritmos da literatura porque as mesmas não se encontram disponíveis. Isso ocorreu em todos os casos de teste do Despacho Econômico/Ambiental e é um dos motivos da utilização do NSGA-II nas comparações entre os algoritmos propostos.

Tabela 27: Multiobjetivo - Estudo de Caso 1: Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	LP	MOSST	NSGA	NPGA	SPEA	NSGA-II	MOPSO	PCEED
G ₁	0,1097	0,1500	0,1125	0,1567	0,1080	0,1062	0,1059	0,1183	0,1098
G ₂	0,2997	0,3000	0,3020	0,2870	0,3284	0,2897	0,3177	0,3019	0,2998
G ₃	0,5243	0,5500	0,5311	0,4671	0,5386	0,5289	0,5216	0,5224	0,5244
G ₄	1,0161	1,0500	1,0208	1,0467	1,0067	1,0025	1,0146	1,0116	1,0160
G ₅	0,5243	0,4600	0,5311	0,5037	0,4949	0,5402	0,5159	0,5254	0,5240
G ₆	0,3597	0,3500	0,3625	0,3729	0,3574	0,3664	0,3583	0,3544	0,3598
Custo	600,111	606,314	605,889	600,572	600,259	600,15	600,155	600,12	600,18
Emissão	0,22214	0,22330	0,22220	0,22282	0,22116	0,2215	0,22188	0,2216	0,2537

Tabela 28: Multiobjetivo - Estudo de Caso 1: Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	LP	MOSST	NSGA	NPGA	SPEA	NSGA-II	MOPSO	PCEED
G ₁	0,4060	0,4000	0,4095	0,4394	0,4002	0,4116	0,4074	0,4015	0,3918
G ₂	0,4590	0,4500	0,4626	0,4511	0,4474	0,4532	0,4577	0,4594	0,4603
G ₃	0,5379	0,5500	0,5426	0,5105	0,5166	0,5329	0,5389	0,5511	0,5252
G ₄	0,3829	0,4000	0,3884	0,3871	0,3688	0,3832	0,3837	0,3919	0,3810
G ₅	0,5379	0,5500	0,5427	0,5553	0,5751	0,5383	0,5352	0,5413	0,5467
G ₆	0,5100	0,5000	0,5142	0,4905	0,5259	0,5148	0,5110	0,5111	0,5528
Emissão	0,19420	0,19424	0,19418	0,19436	0,19433	0,1942	0,19420	0,19420	0,2125
Custo	638,273	639,600	644,112	639,231	639,182	638,51	638,269	637,42	644,40

6.3.2 Multiobjetivo - Estudo de Caso 2

Estas simulações foram realizadas utilizando a mesma instância com 6 geradores e demanda de 2,834 MW do estudo de caso 1, mas foram considerados os coeficientes de perdas de energia (GUERRERO, 2004), tornando a restrição de demanda não linear - o que torna o problema mais complexo. Os coeficientes de custo combustível e emissão de NO_x são os mesmos apresentados para o estudo de caso 1 (Tabelas 68 e 69, respectivamente). Os coeficientes de perda podem ser vistos na Tabela 70 no Apêndice A.2. Esta instância pode ser encontrada também em Guerrero (2004).

A Tabela 29 apresenta os melhores resultados obtidos pelos algoritmos propostos juntamente com o NSGA-II para o objetivo custo. Pode-se observar que para este objetivo o algoritmo NSGA-II obteve o melhor resultado, dominando os valores dos demais algoritmos.

A Tabela 30 apresenta os melhores resultados obtidos pelos algoritmos propostos juntamente com o NSGA-II para o objetivo emissão. Para este objetivo o algoritmo SICMO_e obteve o melhor valor. Cabe destacar que o extremo do algoritmo SICMO_p para este objetivo foi dominado tanto pelo NSGA-II quanto pelo SICMO_e.

O rank de dominância foi inconclusivo com relação a qual dos três algoritmos é melhor

Tabela 29: Multiobjetivo - Estudo de Caso 2: Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	SICMO _p	NSGA-II
Custo	603,372	603,390	603,332
Emissão	0,20420	0,20426	0,20281
Perda	0,01276	0,01245	0,01228

Tabela 30: Multiobjetivo - Estudo de Caso 2: Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	SICMO _p	NSGA-II
Emissão	0,18715	0,18723	0,18719
Custo	636,129	634,335	633,732
Perda	0,01214	0,01168	0,01152

(nenhum algoritmo dominou os demais). Os resultados (p-valores) obtidos pelo teste Kruskal-Wallis para os três indicadores de qualidade são apresentados nas Tabelas 31, 32 e 33. Os valores menores que 0,05 indicam que o algoritmo da linha é significativamente melhor do que o algoritmo da coluna.

Tabela 31: Multiobjetivo - Estudo de Caso 2: P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	4,2x10⁻³¹	8,6x10⁻⁶⁹
SICMO _p	1,0	-	1,5x10⁻³³
NSGA-II	1,0	1,0	-

Tabela 32: Multiobjetivo - Estudo de Caso 2: P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	8,5x10⁻³⁰	5,5x10⁻⁶⁵
SICMO _p	1,0	-	1,1x10⁻²⁹
NSGA-II	1,0	1,0	-

Tabela 33: Multiobjetivo - Estudo de Caso 2: P-valores para o indicador R₂.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	8,8x10⁻³¹	6,6x10⁻⁶⁸
SICMO _p	1,0	-	1,2x10⁻³²
NSGA-II	1,0	1,0	-

Houve superioridade estatística do algoritmo SICMO_e frente aos demais algoritmos em todos os indicadores de qualidade. Além disso, pode-se observar que ambas versões culturais apresentam melhores valores de indicadores de qualidade do que o NSGA-II. Os tempos médios de execução dos algoritmos são: SICMO_e = 88,36s, SICMO_p = 4,08s e

NSGA-II = 33,10s. Pode-se observar que a versão cultural baseada em dominância de Pareto é a mais rápida e a escalarizada é a mais lenta dentre as três versões.

A Figura 44 apresenta um exemplo de conjunto de aproximação da fronteira de Pareto obtido por cada um dos algoritmos propostos juntamente com o NSGA-II, onde pode-se observar a supremacia do algoritmo SICMO_e sobre os demais no quesito qualidade das soluções geradas.

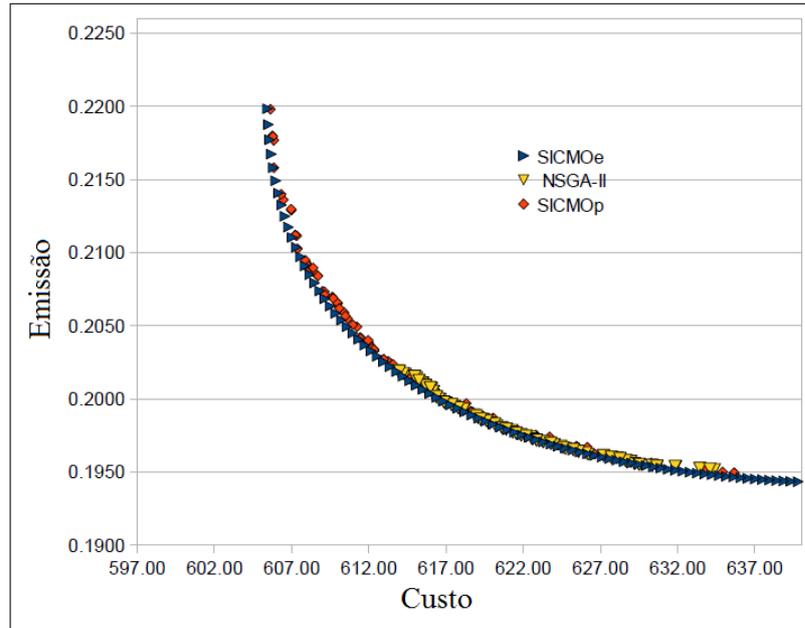


Figura 44: Exemplos de conjuntos de aproximação das fronteiras de Pareto obtidos pelos algoritmos para a instância com 6 geradores e demanda de 2,834 MW.

Para este estudo de caso, a literatura apresenta os algoritmos MEED e ECED (GUERRERO, 2004).

Tabela 34: Multiobjetivo - Estudo de Caso 2 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	MEED	ECED
G ₁	0,15094	-	-
G ₂	0,28423	-	-
G ₃	0,57958	-	-
G ₄	0,98126	-	-
G ₅	0,50499	-	-
G ₆	0,34577	-	-
Custo	603,3728	606,0	606,0
Emissão	0,20420	0,22173	0,22173

A Tabela 34 apresenta os melhores resultados obtidos pelo algoritmo proposto (SICMO_e) e pelos algoritmos descritos na literatura para o objetivo custo. Pode-se observar que para

este objetivo o algoritmo SICMO_e obteve o melhor resultado dominando os valores reportados na literatura.

A Tabela 35 apresenta os melhores resultados obtidos pelo algoritmo proposto e pelos algoritmos descritos na literatura para o objetivo emissão. Para este objetivo o algoritmo SICMO_e obteve novamente os melhores valores, mais uma vez dominando os valores obtidos pelos outros algoritmos.

Tabela 35: Multiobjetivo - Estudo de Caso 2 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	MEED	ECED
G ₁	0,39256	-	-
G ₂	0,49521	-	-
G ₃	0,50376	-	-
G ₄	0,45519	-	-
G ₅	0,50473	-	-
G ₆	0,49470	-	-
Custo	636,129	646,2	645,6
Emissão	0,18715	0,19518	0,19518

6.3.3 Multiobjetivo - Estudo de Caso 3

As simulações foram realizadas utilizando a instância com 6 geradores do padrão IEEE e uma demanda de 2,834 MW (WANG; SINGH, 2008). Os coeficientes de custo combustível e emissão de NO_x são apresentados nas Tabelas 68 e 69, respectivamente, e são os mesmos adotados nos estudos de caso 1 e 2. Esta instância, assim como a segunda instância, leva em consideração as perdas na geração de energia, porém possui menos parâmetros para o cálculo dessas perdas. Os coeficientes de perda podem ser vistos na Tabela 71 do Apêndice A.2. Estes dados podem ser encontrados também em Wang e Singh (2008).

A Tabela 36 apresenta os melhores resultados obtidos pelos algoritmos propostos juntamente com o NSGA-II para o objetivo custo. Pode-se observar que para este objetivo o algoritmo NSGA-II obteve o melhor resultado. Destaca-se que o extremo do algoritmo SICMO_p foi dominado pelos extremos do NSGA-II e do SICMO_e.

A Tabela 37 apresenta os melhores resultados obtidos pelos algoritmos propostos juntamente com o NSGA-II para o objetivo emissão. Para este objetivo o algoritmo NSGA-II também obteve os melhores resultados. Mais uma vez os valores obtidos pelo SICMO_p foram dominados pelos valores dos pontos extremos dos demais algoritmos propostos.

Tabela 36: Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	SICMO _p	NSGA-II
Custo	605,425	605,473	605,178
Emissão	0,22099	0,22191	0,22189
Perda	0,02320	0,02301	0,02233

Tabela 37: Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	SICMO _p	NSGA-II
Emissão	0,19417	0,19428	0,19418
Custo	646,104	649,071	645,555
Perda	0,03486	0,03518	0,03452

Como o rank de dominância foi inconclusivo com relação a qual algoritmo é melhor (nenhum algoritmo dominou os demais), foi aplicado o teste não-paramétrico de Kruskal-Wallis sobre três indicadores de qualidade das soluções para determinar qual algoritmo é superior em termos de fronteira de Pareto. Os indicadores utilizados foram o hipervolume, o épsilon unário e o R_2 . Os resultados (p-valores) são apresentados nas Tabelas 38, 39 e 40. Valores menores que 0,05 indicam que o algoritmo da linha é significativamente melhor do que o algoritmo da coluna.

Tabela 38: Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	1,0x10⁻³⁴	4,4x10⁻⁶⁷
SICMO _p	1,0	-	6,0x10⁻³²
NSGA-II	1,0	1,0	-

Tabela 39: Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	2,2x10⁻³⁴	2,4x10⁻⁶⁶
SICMO _p	1,0	-	3,3x10⁻³¹
NSGA-II	1,0	1,0	-

Tabela 40: Multiobjetivo - Estudo de Caso 3 - P-valores para o indicador R_2 .

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	1,5x10⁻³⁵	6,8x10⁻⁶⁹
SICMO _p	1,0	-	9,4x10⁻³⁴
NSGA-II	1,0	1,0	-

Mais uma vez é possível observar a superioridade dos algoritmos propostos frente ao NSGA-II e da versão escalarizada ($SICMO_e$) frente à versão com dominância de Pareto ($SICMO_p$). Mais uma vez o algoritmo $SICMO_p$ foi mais rápido que os demais com tempo médio igual a 11,65s, seguido pelo NSGA-II com tempo médio de 30,99s e o $SICMO_e$ com tempo médio de 74,55s.

A Figura 45 apresenta um exemplo de conjunto de aproximação da fronteira de Pareto obtido por cada um dos algoritmos propostos juntamente com o NSGA-II, onde pode-se observar a supremacia do algoritmo $SICMO_e$ sobre os demais no quesito qualidade das soluções geradas.

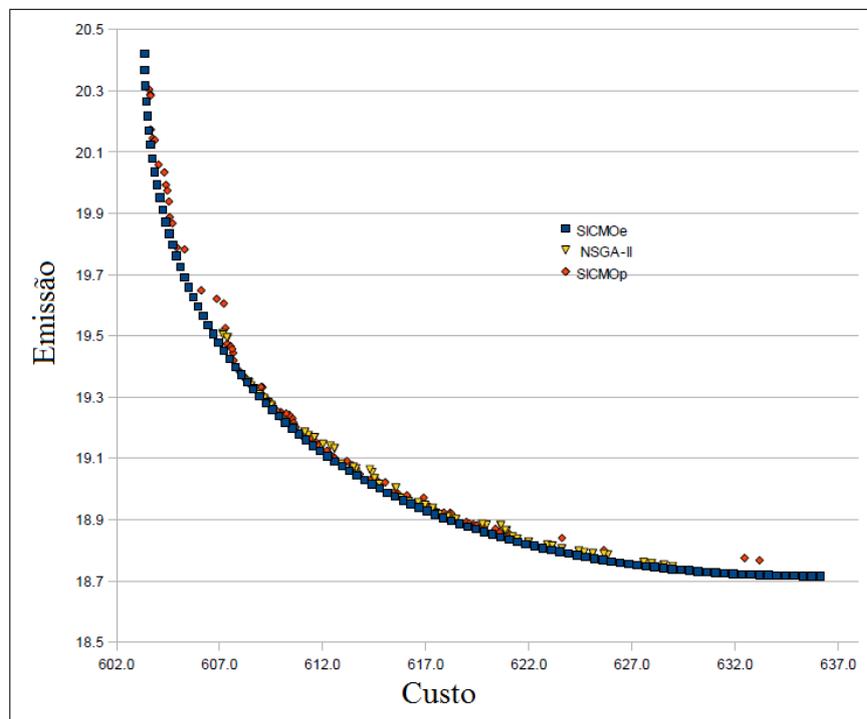


Figura 45: Exemplos de conjuntos de aproximação das fronteiras de Pareto obtidas pelos algoritmos para a instância com 6 geradores e demanda de 2,834 MW.

Tabela 41: Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para o custo de combustível

	$SICMO_e$	MOPSO	NSGA	NPGA	SPEA	NSGA-II
G_1	0,11215	0,1153	0,1168	0,1245	0,1086	0,1182
G_2	0,29107	0,3062	0,3165	0,2792	0,3056	0,3148
G_3	0,57993	0,5962	0,5441	0,6284	0,5818	0,5910
G_4	0,99444	0,9803	0,9447	1,0264	0,9846	0,9710
G_5	0,52452	0,5141	0,5498	0,4693	0,5288	0,5172
G_6	0,35510	0,3550	0,3964	0,3993	0,3584	0,3548
Custo	605,425	607,78	608,245	608,147	607,807	607,801
Emissão	0,22099	0,2198	0,21664	0,22364	0,22015	0,21891

Os resultados obtidos pelo SICMO_e foram comparados com os seguintes algoritmos da literatura: MOPSO (ABIDO, 2009), NSGA (ABIDO, 2003b), NPGA (ABIDO, 2003a), SPEA (ABIDO, 2003) e NSGA-II (KING; RUGHOOPTH; DEB, 2005). As comparações podem ser vistas nas Tabelas 41 e 42. Tanto para o custo quanto para a emissão os melhores valores foram encontrados pelo algoritmo imune cultural escalarizado.

Tabela 42: Multiobjetivo - Estudo de Caso 3 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	MOPSO	NSGA	NPGA	SPEA	NSGA-II
G ₁	0,41090	0,4104	0,4113	0,3923	0,4043	0,4141
G ₂	0,46359	0,4629	0,4591	0,4700	0,4525	0,4602
G ₃	0,54434	0,5436	0,5117	0,5565	0,5525	0,5429
G ₄	0,39027	0,3896	0,3724	0,3695	0,4079	0,4011
G ₅	0,54437	0,5437	0,5810	0,5599	0,5468	0,5422
G ₆	0,51540	0,5149	0,5304	0,5163	0,5005	0,5045
Emissão	0,19417	0,1942	0,19432	0,19424	0,19422	0,19419
Custo	646,104	645,23	647,251	645,984	642,603	644,133

6.3.4 Multiobjetivo - Estudo de Caso 4

As simulações foram realizadas utilizando a instância com 6 geradores do padrão IEEE e uma demanda de 900 MW (CAIA et al., 2009). Os valores mínimos e máximos para cada gerador juntamente com os coeficientes de custo para esta instância são apresentados na Tabela 72. Os coeficientes de emissão de NO_x estão descritos na Tabela 73. Já a Tabela 74 apresenta os coeficientes de perda. Estas tabelas estão localizadas no Apêndice A.2. Estes dados também podem ser encontrados em Caia et al. (2009).

A Tabela 43 apresenta os melhores resultados obtidos pelos algoritmos propostos para o objetivo custo. Pode-se observar que para este objetivo o algoritmo SICMO_e obteve o melhor resultado.

Tabela 43: Multiobjetivo - Estudo de Caso 4: Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	SICMO _p	NSGA-II
Custo	47328,744	47329,355	48080,166
Emissão	863,23435	860,96318	812,20576
Perda	38,32286	38,32162	50,13147

A Tabela 44 apresenta os melhores resultados obtidos pelos algoritmos propostos juntamente com o NSGA-II. Para este objetivo o algoritmo SICMO_e também obteve o melhor resultado.

Tabela 44: Multiobjetivo - Estudo de Caso 4: Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	SICMO _p	NSGA-II
Emissão	701,45611	701,88329	705,55548
Custo	50262,757	50104,012	49971,792
Perda	41,12074	41,07349	42,81095

O rank de dominância determinou que as versões imune culturais dominam o NSGA-II, mas foi inconclusivo com relação a comparação entre o SICMO_e e o SICMO_p. Portanto, ainda foram aplicados os indicadores de qualidade. Os resultados (p-valores) obtidos pelo teste Kruskal-Wallis para os três indicadores de qualidade são apresentados nas Tabelas 45, 46 e 47. Valores menores que 0,05 indicam que o algoritmo da linha é significativamente melhor do que o algoritmo da coluna.

Tabela 45: Multiobjetivo - Estudo de Caso 4: P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	3,3x10⁻³⁷	2,4x10⁻⁷²
SICMO _p	1,0	-	3,3x10⁻³⁷
NSGA-II	1,0	1,0	-

Tabela 46: Multiobjetivo - Estudo de Caso 4: P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	3,3x10⁻³⁷	2,4x10⁻⁷²
SICMO _p	1,0	-	3,3x10⁻³⁷
NSGA-II	1,0	1,0	-

Tabela 47: Multiobjetivo - Estudo de Caso 4: P-valores para o indicador R₂.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	3,3x10⁻³⁷	2,4x10⁻⁷²
SICMO _p	1,0	-	3,3x10⁻³⁷
NSGA-II	1,0	1,0	-

Novamente pode-se observar a seguinte ordem com relação a qualidade estatística dos indicadores: SICMO_e melhor que SICMO_p melhor que NSGA-II. O tempo médio de execução do algoritmo SICMO_e é 82,97s, do SICMO_p é 5,87s e do NSGA-II é 28,55s. Portanto, o algoritmo imune cultural baseado em dominância de Pareto é mais rápido que os demais e o escalarizado é o algoritmo mais lento.

A Figura 46 apresenta um exemplo de conjunto de aproximação da fronteira de Pareto obtido por cada um dos algoritmos propostos, onde pode-se observar a supremacia do

algoritmo $SICMO_e$ sobre os demais no quesito qualidade das soluções geradas.

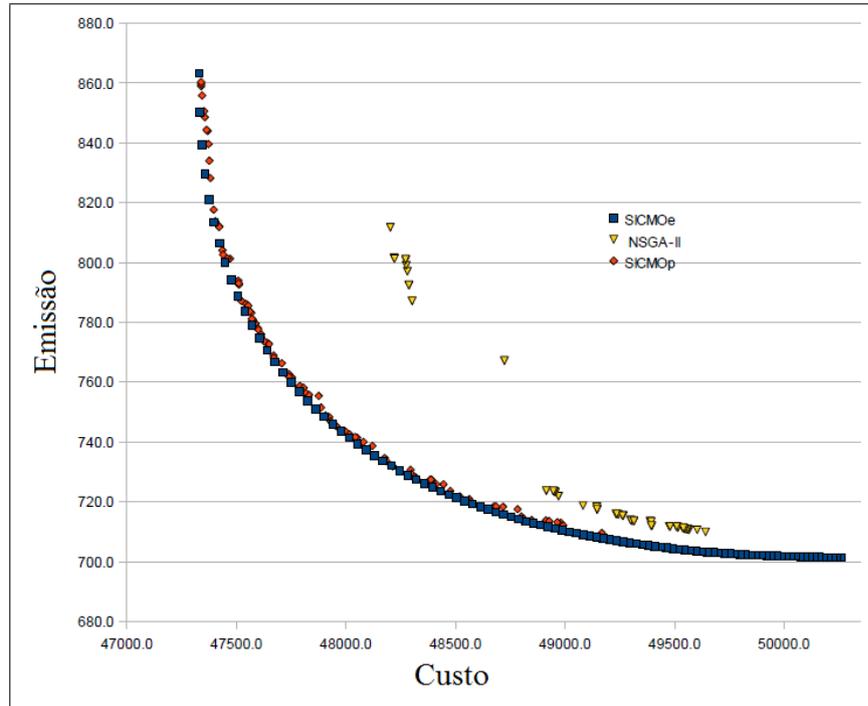


Figura 46: Exemplos de conjuntos de aproximação das fronteiras de pareto obtidas pelos algoritmos para a instância com 6 geradores e demanda de 900 MW.

Os resultados do $SICMO_e$ foram comparados com os algoritmos MOCPSO (CAIA et al., 2009) e ADS (PALANICHAMY; BABU, 2008) e se encontram nas Tabelas 48 e 49.

Tabela 48: Multiobjetivo - Estudo de Caso 4 - Solução que apresenta o melhor valor para o custo de combustível

	$SICMO_e$	MOCPSO	ADS
G_1	33,99430	33,71	33,77
G_2	12,97261	12,65	12,65
G_3	151,79127	150,56	150,56
G_4	147,27623	148,50	148,50
G_5	294,23216	296,3	296,29
G_6	298,05630	296,72	293,68
Custo	47328,744	47187,41	47188,29
Emissão	863,23435	857,81	857,74
Perda Ideal	-	35,44	35,45
Perda Real	38,32286	38,10	38,10

As tabelas mostram que os resultados apresentados na literatura são superiores aos encontrados pelo $SICMO_e$, contudo as perdas reportadas (perdas ideais) não correspondem à aplicação da matriz de coeficientes de perdas (Tabela 74) aos valores encontrados para os geradores. Aparentemente a discrepância ocorre porque no artigo (PALANICHAMY;

Tabela 49: Multiobjetivo - Estudo de Caso 4 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	MOCPSO	ADS
G ₁	122,74627	124,51	124,51
G ₂	122,74442	124,51	124,51
G ₃	139,22425	140,31	140,31
G ₄	141,98691	140,31	140,31
G ₅	206,64043	204,14	204,15
G ₆	207,77847	204,14	204,15
Emissão	701,45611	696,92	696,99
Custo	50262,757	50217,56	50217,62
Perdas Ideais	-	37,91	37,92
Perdas Reais	41,12074	41,01	41,01

BABU, 2008) existem fórmulas para se encontrar os valores ideais de perdas para uma determinada demanda (uma fórmula para as perdas do custo ótimo e outra fórmula para as perdas da emissão ótima).

Outro ponto a ser destacado é que o algoritmo ADS é um algoritmo exato, mas que só pode ser aplicado caso a equação do custo e da emissão sejam diferenciáveis (PALANICHAMY; BABU, 2008). Com isso ele não poderia ser aplicado a nenhum dos estudos de caso anteriores.

6.3.5 Multiobjetivo - Estudo de Caso 5

Estas simulações foram realizadas utilizando a instância com 14 geradores do padrão IEEE e uma demanda de 3668 MW (GUERRERO, 2004). Os coeficientes de custo de combustível e emissão de NO_x são apresentados nas Tabelas 75 e 76, respectivamente. Esta instância leva em consideração as perdas na geração de energia. Os coeficientes de perda podem ser vistos na Tabela 77. Estes dados podem ser encontrados também em Guerrero (2004) e em Venkatesh, Gnanadass e Padhy (2003).

A Tabela 50 apresenta os melhores resultados obtidos pelos algoritmos desenvolvidos neste trabalho e pelo NSGA-II para o objetivo custo. Pode-se observar que para este objetivo o algoritmo SICMO_e obteve o melhor resultado, dominando os extremos dos demais algoritmos.

A Tabela 51 apresenta os melhores resultados obtidos pelos algoritmos desenvolvidos neste trabalho e pelo NSGA-II para o objetivo emissão de poluentes. Para este objetivo o algoritmo SICMO_e também obteve os melhores resultados.

Tabela 50: Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	SICMO _p	NSGA-II
Custo	16657,456	16758,036	16689,021
Emissão	15226,72464	14621,76740	14507,41385
Perda	0,00016	6,12832	0,45426

Tabela 51: Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	SICMO _p	NSGA-II
Emissão	10976,59940	11184,78731	12145,68250
Custo	17969,388	17993,817	17609,247
Perda	0,77250	0,01734	0,14867

O tempo médio de execução do algoritmo SICMO_e é 117,15s, do SICMO_p é 4,39s e do NSGA-II é 47,28s. Portanto, o algoritmo escalarizado é mais lento que os demais e o algoritmo cultural baseado em dominância de Pareto é o que possui o menor tempo computacional.

Através do teste rank de dominância pode-se concluir que o conjunto de aproximação da fronteira de Pareto encontrado pelo algoritmo SICMO_e domina os conjuntos encontrados pelo algoritmo SICMO_p e pelo NSGA-II. Por este mesmo teste conclui-se que o conjunto obtido pelo SICMO_p domina o conjunto obtido pelo NSGA-II. Portanto, o algoritmo SICMO_e apresenta melhor desempenho que os demais algoritmos para este estudo de caso, enquanto o algoritmo SICMO_p é superior ao NSGA-II.

Apenas por completude da análise, as Tabelas 52, 53 e 54 apresentam os p-valores para o teste Kruskal-Wallis sobre os indicadores de qualidade hipervolume, o épsilon unário e o R₂, respectivamente. Valores menores que 0,05 indicam que o algoritmo da linha é significativamente melhor do que o algoritmo da coluna.

Tabela 52: Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	9,5e⁻³⁷	1,2e⁻⁷¹
SICMO _p	1,0	-	1,3e⁻³⁶
NSGA-II	1,0	1,0	-

A Figura 47 apresenta um exemplo de conjunto de aproximação da fronteira de Pareto obtido por cada um dos algoritmos propostos, onde pode-se observar a supremacia do algoritmo SICMO_e sobre os demais no quesito qualidade das soluções geradas.

Tabela 53: Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	$9,3e^{-21}$	$1,0e^{-55}$
SICMO _p	1,0	-	$2,2e^{-30}$
NSGA-II	1,0	1,0	-

Tabela 54: Multiobjetivo - Estudo de Caso 5 - P-valores para o indicador R₂.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	$2,6e^{-36}$	$3,1e^{-71}$
SICMO _p	1,0	-	$1,8e^{-36}$
NSGA-II	1,0	1,0	-

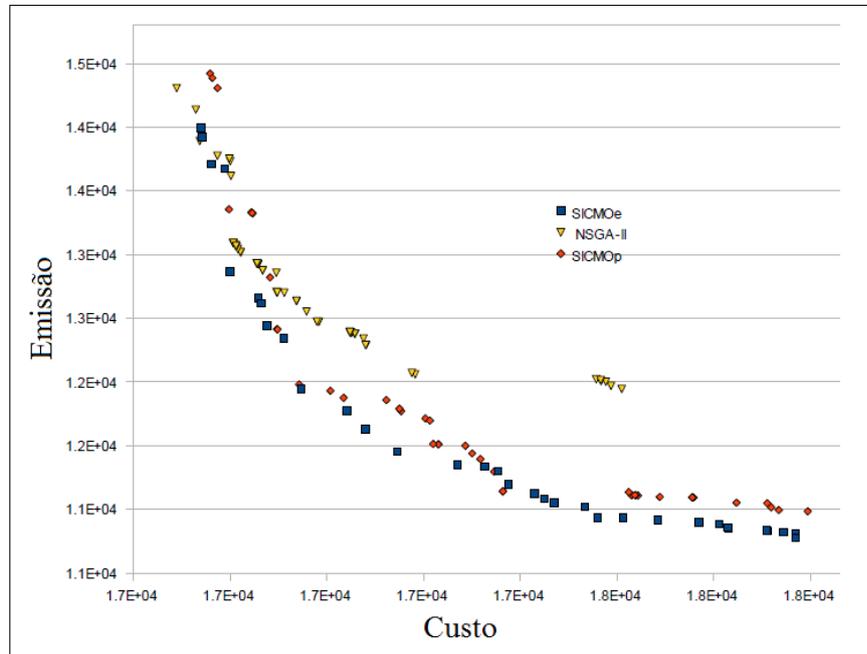


Figura 47: Exemplos de conjuntos de aproximação das fronteiras de Pareto obtidas pelos algoritmos para a instância com 14 geradores e demanda de 3668 MW.

Pode-se concluir que o algoritmo escalarizado (SICMO_e) foi o que apresentou melhor desempenho dentre os três algoritmos que participaram das simulações (SICMO_e, SICMO_p e NSGA-II) para este estudo de caso. Portanto, os resultados obtidos por este algoritmo serão comparados com os resultados reportados na literatura.

A Tabela 55 apresenta os melhores resultados obtidos pelo algoritmo proposto (SICMO_e) e pelo algoritmo EP (VENKATESH; GNANADASS; PADHY, 2003) para o objetivo custo. Pode-se observar que para este objetivo o algoritmo SICMO_e obteve o melhor resultado, dominando a solução apresentada na literatura.

Já a Tabela 56 apresenta os melhores resultados obtidos pelo melhor algoritmo pro-

Tabela 55: Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	EP
G ₁	337,2100	-
G ₂	266,6770	-
G ₃	171,0151	-
G ₄	244,0732	-
G ₅	291,6092	-
G ₆	289,1739	-
G ₇	175,2398	-
G ₈	131,9961	-
G ₉	309,3537	-
G ₁₀	302,2976	-
G ₁₁	289,5251	-
G ₁₂	553,7834	-
G ₁₃	197,1128	-
G ₁₄	108,9331	-
Custo	16657,456	18475,649
Emissão	15226,72464	15772,914

posto e pelo algoritmo EP para o objetivo emissão. Para este objetivo o algoritmo SICMO_e obteve novamente o melhor resultado, também dominando o extremo reportado pelo algoritmo EP.

Tabela 56: Multiobjetivo - Estudo de Caso 5 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	EP
G ₁	265,9367	-
G ₂	146,3080	-
G ₃	302,1005	-
G ₄	370,3701	-
G ₅	217,3118	-
G ₆	424,3345	-
G ₇	293,8395	-
G ₈	254,5088	-
G ₉	221,6574	-
G ₁₀	382,3772	-
G ₁₁	121,8969	-
G ₁₂	275,6027	-
G ₁₃	247,1418	-
G ₁₄	145,3875	-
Emissão	10976,59940	15472,959
Custo	17969,388	18498,355

6.3.6 Multiobjetivo - Estudo de Caso 6

Estas simulações foram realizadas utilizando a instância com 14 geradores do padrão IEEE e uma demanda de 4242 MW (GUERRERO, 2004). Os coeficientes de custo de combustível e emissão de NO_x são apresentados nas Tabelas 75 e 78, respectivamente. Esta instância leva em consideração as perdas na geração de energia. Os coeficientes de perda podem ser vistos na Tabela 77. Estes dados podem ser encontrados também em Guerrero (2004). Esta instância é parecida com a instância do estudo de caso 5, mas a demanda e os coeficientes de emissão são diferentes.

Os melhores resultados obtidos pelos algoritmos desenvolvidos neste trabalho e pelo NSGA-II para o objetivo custo encontram-se nas Tabelas 57 e 58. Por estes resultados, pode-se concluir que o algoritmo SICMO_e mostrou-se melhor tanto para o objetivo custo quanto para o objetivo emissão.

Tabela 57: Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO_e	SICMO_p	NSGA-II
Custo	20090,245	20133,323	20103,144
Emissão	346691,77944	348711,04567	341958,78417
Perda	0.39444	1,38511	0,00001

Tabela 58: Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO_e	SICMO_p	NSGA-II
Emissão	305593,40200	306879,01640	327125,01825
Custo	20973,693	21051,772	20447,372
Perda	0,00000	1,01098	0,00151

O tempo médio de execução do algoritmo SICMO_e é 352,54s, do SICMO_p é 1,61s e do NSGA-II é 17,36s. Portanto, o algoritmo escalarizado é mais lento que os demais e o algoritmo cultural baseado em dominância de Pareto é o que possui o menor tempo computacional.

Através do teste rank de dominância pode-se concluir que o conjunto de aproximação da fronteira de Pareto encontrado pelo algoritmo SICMO_e domina os conjuntos encontrados pelo algoritmo SICMO_p e pelo NSGA-II. Por este mesmo teste conclui-se que o conjunto obtido pelo SICMO_p domina o conjunto obtido pelo NSGA-II. Portanto, o algoritmo SICMO_e apresenta melhor desempenho que os demais algoritmos para este estudo de caso, enquanto o algoritmo o NSGA-II é inferior aos demais.

Apenas por completude da análise, as Tabelas 59, 60 e 61 apresentam os p-valores para o teste Kruskal-Wallis sobre os indicadores de qualidade hipervolume, o épsilon unário e o R_2 , respectivamente. Os dados apresentados para os indicadores reforçam os resultados do rank de dominância.

Tabela 59: Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador hipervolume.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	2,1e⁻³³	1,5e⁻⁶⁸
SICMO _p	1,0	-	2,2e⁻³⁵
NSGA-II	1,0	1,0	-

Tabela 60: Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador épsilon unário.

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	7,1e⁻³³	5,1e⁻⁶⁸
SICMO _p	1,0	-	3,9e⁻³⁵
NSGA-II	1,0	1,0	-

Tabela 61: Multiobjetivo - Estudo de Caso 6 - P-valores para o indicador R_2 .

	SICMO _e	SICMO _p	NSGA-II
SICMO _e	-	2,7e⁻³¹	2,0e⁻⁶⁶
SICMO _p	1,0	-	2,0e⁻³⁴
NSGA-II	1,0	1,0	-

A Figura 48 apresenta um exemplo de conjunto de aproximação da fronteira de Pareto obtido por cada um dos algoritmos propostos, onde pode-se observar a supremacia do algoritmo SICMO_e sobre os demais no quesito qualidade das soluções geradas.

Pode-se concluir que o algoritmo escalarizado (SICMO_e) foi o que apresentou melhor desempenho dentre os três algoritmos que participaram das simulações (SICMO_e, SICMO_p e NSGA-II) para este estudo de caso. Portanto, os resultados obtidos por este algoritmo serão comparados com os resultados reportados na literatura.

A Tabela 62 apresenta os melhores resultados obtidos pelo algoritmo proposto (SICMO_e) e pelo algoritmo MEED (GUERRERO, 2004) para o objetivo custo. Pode-se observar que para este objetivo o algoritmo SICMO_e obteve o melhor resultado, dominando o extremo encontrado pelo algoritmo MEED.

Já a Tabela 63 apresenta os melhores resultados obtidos pelo melhor algoritmo proposto e pelo algoritmo MEED para o objetivo emissão. Para este objetivo o algoritmo SICMO_e obteve novamente o melhor resultado, dominando a solução reportada na literatura.

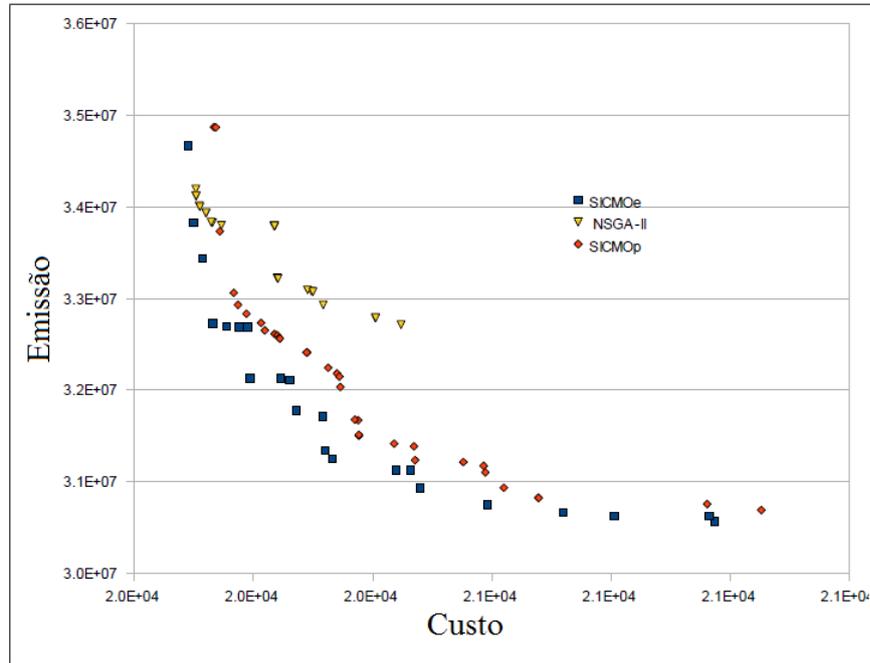


Figura 48: Exemplos de conjuntos de aproximação das fronteiras de Pareto obtidas pelos algoritmos para a instância com 14 geradores e demanda de 4242 MW.

Tabela 62: Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para o custo de combustível

	SICMO _e	MEED
G ₁	284,2266	-
G ₂	339,0450	-
G ₃	241,9360	-
G ₄	323,7859	-
G ₅	372,9674	-
G ₆	263,1863	-
G ₇	152,3293	-
G ₈	179,7235	-
G ₉	366,8545	-
G ₁₀	403,9445	-
G ₁₁	356,5215	-
G ₁₂	576,9668	-
G ₁₃	156,2573	-
G ₁₄	224,6498	-
Custo	20090,245	20939
Emissão	346691,77944	362260

Tabela 63: Multiobjetivo - Estudo de Caso 6 - Solução que apresenta o melhor valor para a emissão de poluentes

	SICMO _e	EP
G ₁	264,6333	-
G ₂	281,9664	-
G ₃	313,1989	-
G ₄	288,7254	-
G ₅	382,6414	-
G ₆	379,3915	-
G ₇	287,4402	-
G ₈	300,7538	-
G ₉	288,0135	-
G ₁₀	280,5607	-
G ₁₁	251,0587	-
G ₁₂	311,1201	-
G ₁₃	316,5834	-
G ₁₄	295,9128	-
Emissão	305593,40200	328610
Custo	20973,693	21771

6.3.7 Conclusões sobre os Estudos de Caso Multiobjetivo

Os resultados mostram que com relação aos indicadores de qualidade hipervolume, épsilon unário e R_2 , as versões imune culturais propostas são superiores ao NSGA-II (com exceção do primeiro estudo de caso onde o NSGA-II mostrou-se melhor que o SICMO_p e SICMO_{pc}). Para os três últimos estudos de caso, os conjuntos de aproximação encontrados pelos algoritmos propostos dominam os conjuntos de aproximação encontrados pelo NSGA-II, mostrando a eficiência dos algoritmos propostos na solução do problema. Os conjuntos de aproximação do SICMO_e também dominaram os conjuntos de aproximação do SICMO_p nos estudos de caso 5 e 6. Com isto, é possível concluir que em termos de qualidade dos conjuntos de aproximação gerados o SICMO_e apresenta o melhor desempenho seguido pelo SICMO_p.

No tocante ao tempo computacional, para todos os estudos de caso observou-se que o algoritmo SICMO_p foi o mais rápido enquanto o SICMO_e foi mais lento que os demais.

Na comparação entre o melhor algoritmo proposto e aqueles reportados na literatura seria melhor comparar os conjuntos de aproximação gerados pelos algoritmos propostos com aqueles gerados pelos algoritmos da literatura. Porém nenhum artigo apresenta os conjuntos de aproximação de maneira que esta comparação não foi viável. Deste modo compararam-se os pontos extremos dos conjuntos de aproximação. No ponto extremo equivalente à minimização do objetivo custo o SICMO_e apresentou os melhores resultados

em todos os estudos de caso com exceção do estudo de caso 4. Já para o ponto extremo correspondente à minimização do objetivo emissão de poluentes o SICMO_e mostrou-se melhor nos estudos de caso 2, 3, 5 e 6. No estudo de caso 1 quatro algoritmos envolvidos na comparação obtiveram a mesma emissão (inclusive o SICMO_e), mas o algoritmo MOPSO obteve melhor valor para o objetivo custo associado, dominando as demais soluções.

Observou-se que, para o algoritmo SICMO_p no primeiro estudo de caso, todos os conhecimentos possuem probabilidades semelhantes de serem aplicados e que o algoritmo apresenta dificuldades em reduzir os valores mínimos dos objetivos.

Cabe salientar que para todos os estudos de caso onde as perdas de energia foram consideradas o algoritmo de reparação mostrou-se eficiente, pois baixos valores de perda foram encontrados e sempre foi possível reparar as soluções infactíveis.

7 *Conclusões e Trabalhos Futuros*

Nesta tese foram propostos algoritmos imuno-culturais para o problema do despacho econômico de energia e para o problema do despacho econômico/ambiental de energia.

No problema do despacho econômico de energia observou-se que a versão imuno-cultural (SIC) apresentou os melhores resultados. Para todos os estudos de caso ela encontrou um valor de custo mínimo de combustível igual ou inferior aos reportados na literatura recente. A versão imuno-cultural com multiplicador baseado em sequência caótica também apresentou bom desempenho, particularmente no segundo estudo de caso para este problema. Intervalos de confiança e testes estatísticos não-paramétricos foram utilizados para determinar a significância estatística dos resultados.

No primeiro estudo de caso, o comportamento do algoritmo imuno-cultural foi melhor analisado. Pode-se notar então uma maior probabilidade de uso das influências dos conhecimentos histórico e normativo. Além disto, pode-se observar que a aplicação da busca local quase-simplex é capaz de melhorar a qualidade das soluções encontradas pelo algoritmo. Percebeu-se também que a função de influência principal proposta foi capaz de obter resultados melhores que as funções de influência principal aleatória (BECERRA; COELLO, 2006a) e baseada na teoria do valor marginal (REYNOLDS; ALI; ALOMARI, 2006). A diferença é estatisticamente significativa entre a abordagem proposta e aquelas da literatura para o primeiro estudo de caso.

No quarto estudo de casos para o problema do despacho econômico de energia, observou-se que o método de reparação para problemas que consideram perdas de energia foi eficiente: sempre que uma solução ineficaz foi gerada ele conseguiu repará-la e os resultados obtidos pelo algoritmo SIC foram superiores àqueles reportados na literatura para os valores de custo mínimo, médio e máximo.

Devido aos bons resultados obtidos pelo SIC em todos os estudos de casos do problema do despacho econômico de energia utilizados nesta tese é possível afirmar que ele se encontra entre os algoritmos do estado-da-arte para o problema.

No problema do despacho econômico/ambiental de energia os resultados obtidos pelos algoritmos imuno-culturais também se mostraram satisfatórios. Quando comparados com

um algoritmo do estado-da-arte para otimização multiobjetivo, o NSGA-II, os resultados obtidos pelas versões imuno-culturais para os indicadores de qualidade hipervolume, épsilon unário e R_2 mostraram-se superiores. O algoritmo SICMO_e obteve melhores valores que o NSGA-II em todos os estudos de caso enquanto o SICMO_p obteve valor inferior somente no primeiro estudo de caso, sendo superior em todos os demais.

Para estudos de caso 4, 5 e 6 os conjuntos de aproximação encontrados pelos algoritmos propostos dominaram (pelo *rank* de dominância) os conjuntos de aproximação encontrados pelo NSGA-II, demonstrando a eficiência dos algoritmos propostos na solução do problema.

Os conjuntos de aproximação do SICMO_e também dominaram os conjuntos de aproximação do SICMO_p nos estudos de caso 5 e 6. Com isto, é possível concluir-se que, em termos de qualidade dos conjuntos de aproximação gerados, o SICMO_e apresenta o melhor desempenho seguido pelo SICMO_p.

Já com relação ao tempo computacional dos algoritmos ficou claro que o SICMO_p é mais eficiente que o SICMO_e e o NSGA-II, constituindo-se como uma boa opção de compromisso entre qualidade das soluções geradas e o tempo gasto para encontrá-las.

Nos estudos de caso onde as perdas de energia foram consideradas no problema do despacho econômico ambiental (casos de 2 a 6), percebeu-se a eficiência do procedimento reparo: todas as soluções infactíveis geradas foram reparadas, valores pequenos para as perdas foram encontradas e os algoritmos imuno-culturais foram capazes de encontrar boas soluções, ou seja, o procedimento não prejudicou a capacidade de otimização dos algoritmos propostos.

Além disto, o primeiro estudo de caso do problema de despacho econômico/ambiental foi utilizado para aferir o comportamento dos conhecimentos no algoritmo. Observou-se que todas as funções de influência dos conhecimentos possuem probabilidades semelhantes de serem aplicadas tanto para o algoritmo SICMO_p quanto para o algoritmo SICMO_e, o que é diferente do que foi observado no algoritmo SIC para o problema do despacho econômico de energia. Esta diferença de comportamento deve ser devido às características diferentes dos problemas e à capacidade dos algoritmos imuno-culturais de se adaptarem a estas características. Neste estudo de caso pode-se concluir também que o algoritmo SICMO_p apresenta dificuldades em encontrar os extremos dos valores dos objetivos.

Na comparação com os algoritmos da literatura para o problema do despacho econômico/ambiental, o ideal seria comparar os conjuntos de aproximação gerados pelos algoritmos propostos com aqueles gerados pelos algoritmos da literatura. Porém nenhum artigo apresenta os conjuntos de aproximação de maneira que esta comparação não foi

viável. Por isto, os pontos extremos dos conjuntos de aproximação foram comparados. No ponto extremo equivalente à minimização do objetivo custo o SICMO_e apresentou os melhores resultados em todos os estudos de caso com exceção do estudo de caso 4. Já para o ponto extremo correspondente à minimização do objetivo emissão de poluentes o SICMO_e mostrou-se melhor nos estudos de caso 2, 3, 5 e 6. No primeiro estudo de caso quatro algoritmos envolvidos na comparação obtiveram a mesma emissão (inclusive o SICMO_e), mas o algoritmo MOPSO obteve melhor valor para o objetivo custo associado, dominando as demais soluções.

Pelos resultados obtidos pode-se afirmar que o algoritmo SICMO_e está entre os melhores para o problema do despacho econômico/ambiental de energia.

Os bons resultados encontrados pelos algoritmos propostos são encorajadores para trabalhos futuros. Algumas possibilidades são:

- Utilização de outros espaços populacionais como, por exemplo, enxame de partículas ou evolução diferencial, visto que estes algoritmos apresentam bons resultados para o despacho econômico de energia;
- Paralelização dos algoritmos imuno-culturais;
- Aplicação dos algoritmos propostos em outros problemas de otimização mono e multiobjetivo;
- Aprimoramento da versão multiobjetivo baseada em Pareto para aplicações em problemas com soluções não suportadas;
- Investigação mais aprofundada das influências dos conhecimentos através de testes estatísticos;
- Estudo de outras formas de conhecimento;
- Uso do tecido social como função de influência;
- Desenvolvimento de um algoritmo imuno-cultural baseado em indicadores;
- Utilização de particionamento de domínio nos algoritmos multiobjetivo propostos;
- Modificações na forma de arquivamento dos anticorpos não-dominados.

Referências Bibliográficas

- ABIDO, M. A niched pareto genetic algorithm for multiobjective environmental/economic dispatch. *Electric Power Systems*, v. 25, n. 2, p. 97–105, 2003.
- ABIDO, M. A novel multiobjective evolutionary algorithm for environmental/economic power dispatch. *Electric Power Systems Research*, v. 65, p. 71–81, 2003.
- ABIDO, M. Multiobjective particle swarm optimization for environmental/economic dispatch problem. *Electric Power Systems Research*, v. 79, p. 1105–1113, 2009.
- ABIDO, M. A. Environmental/economic power dispatch using multiobjective evolutionary algorithms. *IEEE Transactions on Power Systems*, v. 18, n. 4, p. 1529–1537, 2003.
- ABRAHAM, A.; JAIN, L.; GOLDBERG, R. *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*. Berlin: Springer, 2005.
- AICKELIN, U.; CAYZER, S. *The Danger Theory and Its Application to Artificial Immune Systems*. Bristol, Setembro 2002.
- ALI, M. Z. *Using cultural algorithms to solve optimization problems with a social fabric approach*. Tese (Doutorado) — Wayne State University, 2008.
- ALI, M. Z.; REYNOLDS, R. G. An intelligent social fabric influence component in cultural algorithms for knowledge learning in dynamic environments. In: *WI-IAT '09: Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*. Washington, DC, USA: IEEE Computer Society, 2009. p. 161–168. ISBN 978-0-7695-3801-3.
- ALI, M. Z.; REYNOLDS, R. G.; CHE, X. Genetic programming theory and practice vi. In: _____. US: Springer, 2009. (Genetic and Evolutionary Computation), cap. Genetic Programming for Incentive-Based Design within a Cultural Algorithms Framework., p. 1–19.
- ALIEV, R. A.; ALIEV, R. *Soft Computing & Its Applications*. Singapore: World Scientific Publishing Company, 2001. ISBN 9810247001.
- ALMEIDA, C. P.; GONÇALVES, R. A.; DELGADO, M. R. A hybrid immune-based system for the protein folding problem. In: COTTA, C.; HEMER, J. V. (Ed.). *Seventh European Conference on Evolutionary Computation in Combinatorial Optimisation - EvoCOP 2007*. Valencia, Spain: Springer, 2007. (Lecture Notes in Computer Science).

- ALMEIDA, C. P. et al. Ta-pfp: A transgenetic algorithm to solve the protein folding problem. In: ISDA. *7th International Conference on Intelligent Systems Design and Applications*. Rio de Janeiro, Brasil, 2007.
- ALVES, R. et al. Discovering multi-label hierarchical classification rules for protein function prediction. In: PLASTINO, A. et al. (Ed.). *Proc. II Workshop em Algoritmos e Aplicações de Mineração de Dados (Workshop on Algorithms and Applications of Data Mining)*. Sociedade Brasileira de Computação, 2007. p. 87–90. ISBN 9788576691358. Disponível em: <http://www.cs.kent.ac.uk/pubs/2007/2631>.
- ALVES, R. T.; DELGADO, M. R.; FREITAS, A. A. Multi-label hierarchical classification of protein functions with artificial immune systems. In: *BSB '08: Proceedings of the 3rd Brazilian symposium on Bioinformatics*. Santo André, Brazil: Springer-Verlag, 2008. p. 1–12. ISBN 978-3-540-85556-9.
- ALVES, R. T. et al. An artificial immune system for fuzzy-rule induction in data mining. In: YAO, X. et al. (Ed.). *8th International Conference on Parallel Problem Solving from Nature*. Birmingham, UK: Springer-Verlag, 2004. (Lecture Notes in Computer Science, v. 3242), p. 1011–1020.
- ANCHOR, K. P. et al. Extending the computer defense immune system: Network intrusion detection with a multiobjective evolutionary programming approach. In: ICARIS. *1st International Conference on Artificial Immune Systems*. University of Kent at Canterbury, 2002. p. 12–21.
- ANILE, A. M. et al. Direct and immunological computing with quasi-newton method for the protein folding problem. *International Journal Natural Computing*, 2006.
- ANISHCHENKO, V. et al. *Nonlinear Dynamics of Chaotic and Stochastic Systems: Tutorial and Modern Developments*. Berlin: Springer, 2007.
- ARANTES, M. B.; OLIVEIRA, G. T. S.; SARAMAGO, S. F. P. Evolução diferencial aplicada à solução de alguns problemas de engenharia de produção. *FAMAT em Revista*, n. 6, p. 48–61, 2006.
- BACK, T. *Evolutionary Algorithm in Theory and Practice*. Oxford: Oxford University Press, 1996.
- BALAMURUGAN, R.; SUBRAMANIAN, S. Self-adaptive differential evolution based power economic dispatch of generators with valve-point effects and multiple fuel options. *International Journal of Computer Science and Engineering*, v. 1, n. 1, 2007.
- BALTHROP, J.; FORREST, S.; GLICKMAN, M. Coverage and generalization in an artificial immune system. In: *GECCO*. New York: Morgan Kaufmann, 2002a. p. 3–10.
- BALTHROP, J.; FORREST, S.; GLICKMAN, M. Revisiting lysis: Parameters and normal behaviour. In: *Congress on Evolutionary Computation*. Honolulu: ACM, 2002b.
- BAZARAA, M. S.; SHERALI, H. D.; SHETTY, C. M. *Nonlinear Programming*. New Jersey: John Wiley & Sons, 2006.

- BÄCK, T.; SCHWEFEL, H. paul; INFORMATIK, F. Evolutionary computation: An overview. In: *Proceedings of IEEE International Conference on Evolutionary Computation*. Nagoya: IEEE Press, 1996. p. 20–29.
- BECERRA, R. L. *Algoritmos Culturales Aplicados a Optimizacion con Restricciones y Optimizacion Multiobjetivo*. Tese (Doutorado) — Instituto Politécnico Nacional do México, 2002.
- BECERRA, R. L. *Use of Domain Information to Improve the Performance of an Evolutionary Algorithm*. Tese (Doutorado) — Center for Research and Advanced Studies of the National Polytechnic Institute of Mexico - Computer Science Department, Mexico, 2007.
- BECERRA, R. L.; COELLO, C. A. C. Culturizing differential evolution for constrained optimization. In: *Fifth Mexican International Conference in Computer Science*. Colima: [s.n.], 2004. p. 304–311.
- BECERRA, R. L.; COELLO, C. A. C. Optimization with constraints using a cultured differential evolution approach. In: *Genetic and Evolutionary Computation Conference*. [S.l.: s.n.], 2005a. p. 27–34.
- BECERRA, R. L.; COELLO, C. A. C. Optimization with constraints using a cultured differential evolution approach. In: GECCO. *Genetic and Evolutionary Computation Conference*. Washington, DC, 2005b. p. 27–34.
- BECERRA, R. L.; COELLO, C. A. C. Cultured differential evolution for constrained optimization. *Computer Methods in Applied Mechanics and Engineering*, v. 195, n. 33–36, p. 4303–4322, 2006.
- BECERRA, R. L.; COELLO, C. A. C. Cultured differential evolution for constrained optimization. *Computer Methods in Applied Mechanics and Engineering*, v. 195, p. 4303–4322, 2006.
- BECERRA, R. L.; COELLO, C. A. C. Parallel problem solving from nature - ppsn ix. In: _____. Iceland: Springer Berlin Heidelberg, 2006c. (Lecture Notes in Computer Science, v. 4193), cap. Solving Hard Multiobjective Optimization Problems Using ϵ -Constraint with Cultured Differential Evolution, p. 543–552.
- BECERRA, R. L. et al. Alternative techniques to solve hard multiobjective optimization problems. In: *9th annual conference on Genetic and evolutionary computation*. London, UK: [s.n.], 2007. p. 757–754.
- BECERRA, R. L.; SANTANA-QUINTERO, L. V.; COELLO, C. A. C. Multi-objective evolutionary algorithms for knowledge discovery from data bases. In: _____. Berlin: Springer, 2008. cap. Knowledge Incorporation in Multi-Objective Evolutionary Algorithms, p. 23–46.
- BERSINI, H. Artificial immune systems and their applications. In: DASGUPTA, D. Berlin, German: Springer-Verlag, 1999. cap. The Endogenous Double Plasticity of the Immune Network and the Inspiration to tbe Drawn for Engineering Aircrafts, p. 22–44.

- BEZERRA, G. B.; CASTRO, L. N. de. Bioinformatics data analysis using an artificial immune network. In: *International Conference on Artificial Immune Systems*. Edinburgh, UK: Springer-Verlag, 2003a. (Lecture Notes in Computer Science, v. 2787), p. 22–33.
- BEZERRA, G. B.; CASTRO, L. N. de. A hybrid approach for gene expression data clustering. In: *International Conference on Bioinformatics and Computational Biology*. Ribeirão Preto, SP: [s.n.], 2003b.
- BHATTACHARYA, A.; CHATTOPADHYAY, P. K. Biogeography-based optimization for different economic load dispatch problems. *IEEE Transactions on Power Systems*, 2010.
- BONABEAU, E.; DORIGO, M.; THERAULAZ, G. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford: Oxford University Press, 1999.
- BRADLEY, D. W.; TYRRELL, A. M. Immunotronics: Hardware fault tolerance inspired by the immune system. In: ICES. *International Conference on Evolvable Systems*. Edinburgh, Scotland, UK, 2000. p. 11–20. Disponível em: <citeseer.ist.psu.edu-/bradley00immunotronics.html>.
- BRADLEY, E.; TIBSHIRANI, R. J. *An Introduction to the Bootstrap*. USA: Chapman & Hall/CRC, 1994.
- CAI, J. et al. Chaotic ant swarm optimization to economic dispatch. *Electrical Power Systems Research*, v. 77, n. 10, p. 1373–1378, 2007.
- CAIA, J. et al. A multi-objective chaotic particle swarm optimization for environmental/economic dispatch. *Energy Conversion and Management*, v. 50, n. 5, p. 1318–1325, 2009.
- CAMPELO, F. et al. A novel multiobjective immune algorithm using nondominated sorting. In: *International IGTE Symposium on Numerical Field Calculation in Electrical Engineering*. Gras, Austria: [s.n.], 2004.
- CAMPELO, F. et al. A modified immune network algorithm for multimodal electromagnetic problems. *IEEE Transactions on Magnetics*, v. 42, n. 4, p. 1111–1114, 2006.
- CAMPELO, F.; IGARASHI, F. G. G. H. Overview of artificial immune systems for multiobjective optimization. In: *4th International Conference on Evolutionary Multi-Criterion Optimization*. Matsushima, Japan: [s.n.], 2007. p. 937–951.
- CANHAM, R.; JACKSON, A.; TYRRELL, A. Robot error detection using an artificial immune system. In: EH. *NASA/DoD Conference Evolvable Hardware*. Illinois, USA, 2003. p. 199–207.
- CASELLA, G.; BERGER, R. L. *Statistical Inference*. 2. ed. Pacific Grove: Duxbury, 2001.
- CASTRO, L. de; TIMMIS, J. An artificial immune network for multimodal function optimization. In: *IEEE Congress on Evolutionary Computation*. Washington DC: ACM, 2002a.

- CASTRO, L. de; TIMMIS, J. *Artificial Immune Systems: A New Computational Approach*. London. UK.: Springer-Verlag, 2002b. 357 p. ISBN 1852335947.
Disponível em: <<http://www.cs.ukc.ac.uk/pubs/2002/1507>>.
- CASTRO, L. N. d. de; ZUBEN, F. J. V. Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems*, v. 6, n. 3, p. 239–251, 2002.
- CETINKAYA, N. Optimization algorithm for combined economic and emission dispatch with security constraints. In: ICCSA 2009. *The 3rd International Conference on Complex Systems and Applications*. Berkeley, USA, 2009.
- CHAKRABARTI, N. S. R.; CHATTOPADHYAY, P. Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on Evolutionary Computation*, v. 7, n. 1, p. 83–94, 2003.
- CHATURVEDI, K. T.; PANDIT, M.; SRIVASTAVA, L. Particle swarm optimization with time varying acceleration coefficients for non-convex economic power dispatch. *Electrical Power and Energy Systems*, v. 31, p. 249–257, 2009.
- CHRISTIAN, B.; ANDREA, R. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, v. 35, n. 3, p. 268–308, 2003.
- COELHO, L.; LEE, C.-S. Solving economic load dispatch problems in power systems using chaotic and gaussian particle swarm optimization approaches. *International Journal of Electrical Power & Energy Systems*, v. 30, n. 5, p. 297–307, 2008.
- COELHO, L.; MARIANI, V. Particle swarm approach based on quantum mechanics and harmonic oscillator potential well for economic load dispatch with valve-point effects. *Energy Conversion and Management*, v. 49, n. 11, p. 1373–1378, 2008.
- COELHO, L. d. S.; ALMEIDA, A. D. de; MARIANI, V. C. Cultural differential evolution approach to optimize the economic dispatch of electrical energy using thermal generators. In: *IEEE International Conference on Emerging Technologies and Factory Automation*. [S.l.: s.n.], 2008. p. 1378–1383.
- COELHO, L. d. S.; MARIANI, V. C. An improved harmony search algorithm for power economic load dispatch. *Energy Conversion and Management*, v. 50, n. 10, p. 2522–2526, 2009.
- COELHO, L. S.; MARIANI, V. C. Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. *IEEE Transactions on Power Systems*, v. 21, n. 2, 2006.
- COELHO, L. S.; SOUZA, R. T. T.; MARIANI, V. C. Improved differential evolution approach based on cultural algorithms and diversity measure applied to solve economic load dispatch problems. *Mathematics and Computers in Simulation*, v. 79, p. 3136–3147, 2009.
- COELLO, C. A. C.; BECERRA, R. L. Adding knowledge and efficient data structures to evolutionary programming: A cultural algorithm for constrained optimization. In: *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*. New York, USA: [s.n.], 2002a. p. 201–209.

- COELLO, C. A. C.; BECERRA, R. L. Adding knowledge and efficient data structures to evolutionary programming: A cultural algorithm for constrained optimization. In: *Genetic and Evolutionary Computation Conference*. New York, USA: [s.n.], 2002b. p. 201–209.
- COELLO, C. A. C.; CORTÉS, N. C. Solving multiobjective optimization problems using an artificial immune system. *Genetic Programming and Evolvable Machines*, v. 6, n. 2, p. 163–190, Junho 2005.
- COELLO, C. A. C.; CORTÉS, N. C. Solving multiobjective optimization problems using an artificial immune system. *Genetic Programming and Evolvable Machines*, v. 6, n. 2, p. 163–190, 2005.
- COELLO, C. A. C.; CORTÉS, N. C. An approach to solve multiobjective optimization problems based on an artificial immune system. In: TIMMIS, J.; BENTLEY, P. (Ed.). *1st International Conference on Artificial Immune Systems*. UK: [s.n.], 2002. p. 212–221.
- COELLO, C. A. C.; LAMONT, G. B. *Applications of Multi-Objective Evolutionary Algorithms*. Singapore: World Scientific, 2004.
- COELLO, C. A. C.; LAMONT, G. B.; VELDHUIZEN, D. A. V. *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2. ed. New York, USA: Springer, 2007.
- COHON, J. L.; MARKS, D. H. A review and evaluation of multiobjective programming techniques. *Water Resources Research*, v. 11, n. 2, p. 208–220, 1975.
- CORDON, O.; HERRERA, F.; STUTZLE, T. A review of the ant colony optimization metaheuristic: Basis, models and new trends. *Mathware and Soft Computing*, v. 9, n. 2-3, p. 141–175, 2002.
- CORTÉS, D. T.-P. N. C.; COELLO, C. A. C. Handling constraints in global optimization using an artificial immune system. In: ICARIS. *4th International Conference on Artificial Immune Systems*. Banff, Canada, 2005. p. 234–247.
- CORTÉS, N. C.; COELLO, C. A. C. Multiobjective optimization using ideas from the clonal selection principle. In: CANTÚ-PAZ, E. (Ed.). *Workshop Program of Genetic and Evolutionary Computation Conference*. Chicago, USA: [s.n.], 2003a. p. 158–170.
- CORTÉS, N. C.; COELLO, C. A. C. Multiobjective optimization using ideas from the clonal selection principle. In: BARRY, A. (Ed.). *Using Artificial Immune Systems to Solve Optimization Problems*. Chicago, USA: [s.n.], 2003b. p. 312–315.
- COSTA, A. M. et al. Makespan minimization on parallel processors: An immune-based approach. In: TIMMIS, J.; BENTLEY, P.; HART, E. (Ed.). *Special Sessions on Artificial Immune Systems in the Congress on Evolutionary Computation*. Honolulu, Hawaii: ACM, 2002. (Lecture Notes in Computer Science, v. 2787), p. 11–21.
- CUTELLO, V. et al. Exploring the capability of immune algorithms: A characterization of hypermutation operators. In: *International Conference on Artificial Immune Systems*. Catania, Italy: Springer-Verlag, 2004. (Lecture Notes in Computer Science, v. 3239), p. 263–276.

- CUTELLO, V. et al. Immune algorithms with aging operators for the string folding problem and the protein folding problem. In: *EVOCOP. Evolutionary Computation in Combinatorial Optimization*. Lausanne, Switzerland, 2005a. p. 80–90.
- CUTELLO, V. et al. Immune algorithms with aging operators for the string folding problem and the protein folding problem. In: *European Conference on Computation in Combinatorial Optimization*. [S.l.]: Springer-Verlag, 2005b. (Lecture Notes in Computer Science, v. 3448), p. 80–90.
- CUTELLO, V.; NARZISI, G.; NICOSIA, G. A multi-objective evolutionary approach to the protein structure prediction problem. *Journal of the Royal Society Interface*, v. 3, n. 6, p. 139–151, 2006.
- CUTELLO, V. et al. Real coded clonal selection algorithm for global numerical optimization using a new inversely proportional hypermutation operator. In: *The 21st Annual ACM Symposium on Applied Computing - SAC 2006*. Dijon, France: ACM Press, 2006. v. 2, p. 950–954.
- CUTELLO, V. et al. On the convergence of immune algorithms. In: *The First IEEE Symposium on Foundations of Computational Intelligence - FOCI 2007*. Honolulu, Hawaii: [s.n.], 2007.
- CUTELLO, V.; NICOSIA, G.; PAVONE, M. A hybrid immune algorithm with information gain for the graph coloring problem. In: *GECCO. Genetic and Evolutionary Computation Conference*. Illinois, USA, 2003. (Lecture Notes in Computer Science, v. 2723), p. 171 ff.
- CUTELLO, V.; NICOSIA, G.; PAVONE, M. Exploring the capability of immune algorithms: A characterization of hypermutation operators. In: *ICARIS. Third International Conference on Artificial Immune Systems*. Catania, Italy, 2004a. p. 263–276.
- CUTELLO, V.; NICOSIA, G.; PAVONE, M. An immune algorithm with hyper-macromutations for the dill's 2d hydrophobic - hydrophilic model. In: *Congress on Evolutionary Computation*. Portland, Oregon: IEEE Press, 2004b. v. 1, p. 1074–1080.
- CUTELLO, V. et al. An immune algorithm for protein structure prediction on lattice models. *IEEE Transactions on Evolutionary Computation*, v. 10, 2006.
- DANESHYARI, W.; YEN, G. G. Cultural mopso: A cultural framework to adapt parameters of multiobjective particle swarm optimization. In: *IEEE Congress on Evolutionary Computation*. Hong Kong, China: IEEE Press, 2008. p. 1325–1332.
- DAS, D. B.; PATVARDHAN, C. New multi-objective stochastic search technique for economic load dispatch. In: *IEE Proceedings- Transmission and Distribution Generation*. Washington: IEEE, 1998. v. 145, n. 6, p. 747–752.
- DASGUPTA, D.; GONZALEZ, F. An immunity-based technique to characterize intrusions in computer networks. *IEEE Transactions on Evolutionary Computation*, v. 6, n. 3, p. 281–291, 2002.

- DASGUPTA DIPANKAR; MICHALEWICZ, Z. . *Evolutionary Algorithms in Engineering Applications*. USA: Springer, 2001.
- DE CASTRO, L. N. *Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais*. Tese (Doutorado) — Universidade Estadual de Campinas - Faculdade de Engenharia Elétrica e de Computação, 2001.
- DE CASTRO, L. N. *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. 1. ed. USA: Chapman & Hall/CRC, 2006.
- DE CASTRO, L. N.; TIMMIS, J. *Artificial Immune Systems: A New Computational Intelligence Approach*. Great Britain: Springer, 2002.
- DE CASTRO, L. N.; VON ZUBEN, F. J. The clonal selection algorithm with engineering applications. In: GECCO. *Workshop on Artificial Immune Systems and Their Applications*. Las Vegas, Nevada, USA, 2000. p. 36–37.
- DE CASTRO, L. N.; VON ZUBEN, F. J. Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems*, v. 6, n. 3, p. 239–251, 2002.
- DEB, K. *Multi-Objective Optimization using Evolutionary Algorithms*. England: Wiley, 2001.
- DEB, K. et al. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transaction on Evolutionary Computation*, v. 6, n. 2, p. 182–197, 2002.
- DECASTRO, L. N. et al. *Recent Developments In Biologically Inspired Computing*. Hershey, PA, USA: IGI Publishing, 2004. ISBN 1591403138.
- DING, Y.; REN, L. Fuzzy self-tuning immune feedback controller for tissue hyperthermia. In: FUZZ. *IEEE International Conference on Fuzzy Systems*. San Antonio, Texas, 2000. v. 1, p. 534–538.
- DONG, W. et al. The optimization of mixture ratio of model sand based on simplex particle swarm optimization algorithm. In: *in Chinese Control and Decision Conference - CCDC'2008*. Yantai, China: [s.n.], 2008. p. 3868–3872.
- DORIGO, M.; MANIEZZO, V.; COLORNI, A. *The Ant System: An Autocatalytic Optimizing Process*. Milano, Italy, 1991.
- DORIGO, M.; STUTZLE, T. *The Ant Colony Optimization Metaheuristic: Algorithms, Applications and Advances*. Norwell, MA, USA: Kluwer Academic Publishers, 2003. 251–285 p.
- DORIGO, M.; STUTZLE, T. *Ant Colony Optimization*. London, UK: The MIT Press, 2004.
- DURHAM, W. *Co-Evolution: Genes, Culture and Human Diversity*. Stanford, USA: Stanford University Press, 1994.

- DURILLO, J. J. et al. *jMetal: A Java Framework for Developing Multi-Objective Optimization Metaheuristics*. E.T.S.I. Informática, Campus de Teatinos, December 2006.
- EBERHART, R. C.; SHI, Y.; KENNEDY, J. *Swarm Intelligence*. San Diego, CA: Morgan Kaufmann, 2001. (The Morgan Kaufmann Series in Evolutionary Computation).
- ENGELBRECHT, A. P. *Computational Intelligence: An Introduction*. 2. ed. New York: Wiley, 2007.
- ESPONDA, F.; FORREST, S.; HELMAN, P. A formal framework for positive and negative detection schemes. *IEEE Transactions on Systems, Man and Cybernetics*, v. 34, n. 1, 2004.
- FARAG, A.; AL-BAIYAT, S.; CHENG, C. Economic load dispatch multiobjective optimization procedures using linear programming techniques. *IEEE Transactions on Power Systems*, v. 10, n. 2, p. 731–738, 1995.
- FOGEL, D. B. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, 3rd Edition*. New York: Wiley - IEEE Press, 2005. ISBN 9780471669517.
- FONSECA, V. G.; FONSECA, C. M.; HALL, A. O. Inferential performance assessment of stochastic optimisers and the attainment function. In: *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*. London, UK: Springer-Verlag, 2001. p. 213–225.
- FRANÇA, F. O.; VON ZUBEN, F. J.; DE CASTRO, L. N. An artificial immune network for multimodal function optimization on dynamic environments. In: *GECCO. Genetic and Evolutionary Computation*. Washington DC, USA, 2005. p. 289–296.
- FREEMAN, L. M. *Industrial Applications of Genetic Algorithms*. Boca Raton, FL, USA: CRC Press, Inc., 1998. ISBN 0849398010.
- FRESCHI, F.; COELLO, C. A. C.; REPETTO, M. Handbook of research on artificial immune systems and natural computing: Applying complex adaptive technologies. In: _____. Hershey, New York: IGI Publishing, 2009a. cap. Multiobjective Optimization and Artificial Immune Systems: A Review, p. 1–21.
- FRESCHI, F.; COELLO, C. A. C.; REPETTO, M. Handbook of research on artificial immune systems and natural computing: Applying complex adaptive technologies. In: _____. New York: Hershey, PA : Medical Information Science Reference, 2009b. cap. Multiobjective Optimization and Artificial Immune Systems: A Review, p. 1–21.
- FRESCHI, F.; REPETTO, M. Multiobjective optimization by a modified artificial immune system algorithm. In: *4th International Conference on Artificial Immune Systems*. Banff, Canada: [s.n.], 2005. p. 248–261.
- FRESCHI, F.; REPETTO, M. Vis: an artificial immune network for multi-objective optimization. *Engineering Optimization*, v. 38, n. 8, p. 975–996, 2006.

- GAO, F. et al. International conference on computational science. In: _____. China: Springer Berlin Heidelberg, 2007. (Lecture Notes in Computer Science, v. 4490), cap. Cultural Particle Swarm Algorithms for Constrained Multi-objective Optimization, p. 1611–3349.
- GAO, J.; WANG, J. Wbmoais: A novel artificial immune system for multiojective optimization. *Computers & Operations Research*, v. 37, p. 50–61, 2010.
- GARCÍA, S. et al. A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the cec'2005 special session on real parameter optimization. *Journal of Heuristics*, Kluwer Academic Publishers, Hingham, MA, USA, v. 15, n. 6, p. 617–644, 2009. ISSN 1381-1231.
- GARRETT, S. Parameter-free, adaptive clonal selection. In: CEC. *Congress on Evolutionary Computation*. San Diego, CA, USA: ACM, 2004. v. 1, p. 1052–1058.
- GEEM, Z. W. *Music-Inspired Harmony Search Algorithm: Theory and Applications*. USA: Springer, 2009.
- GIBBONS, J. *Nonparametric Statistical Inference*. USA: CRC, 1985.
- GOLDBERG, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Massachusetts, USA: Addison Wesley Longman, 1989.
- GONÇALVES, R. A. et al. A cultural immune system for economic load dispatch with non-smooth cost functions. In: LNCS. *6th International Conference on Artificial Immune System - ICARIS 2007*. Santos, Brasil: Springer, 2007. (Lecture Notes in Computer Science).
- GONÇALVES, R. A. et al. Solving economic load dispatch problem by natural computing intelligent systems. In: *International Conference on Intelligent System Applications to Power Systems (ISAP 2009)*. Rio de Janeiro, Brazil: [s.n.], 2009. p. 1–12.
- GONÇALVES, R. A. et al. A fuzzy cultural immune system for economic load dispatch with non-smooth cost functions. In: *Simpósio Brasileiro de Pesquisa Operacional*. João Pessoa, PB, Brasil: [s.n.], 2008. p. 1–12.
- GONZALEZ, F.; DASGUPTA, D. Anomaly detection using real-valued negative selection. *Journal of Genetic Programming and Evolvable Machines*, v. 4, p. 382–403, 2003.
- GONZÁLEZ, F. A.; DASGUPTA, D. Anomaly detection using real-valued negative selection. *Genetic Programming and Evolvable Machines*, v. 4, n. 4, 2003.
- GREENSMITH, J.; AICKELIN, U.; TWYCROSS, J. Detecting danger: Applying a novel immunological concept to intrusion detection systems. In: ACDM. *6th International Conference in Adaptive Computing in Design and Manufacture*. Clifton, Bristol, UK, 2004.
- GROSAN, C.; ABRAHAM, A. Hybrid evolutionary algorithms. In: _____. USA: Springer, 2007. cap. Hybrid Evolutionary Algorithms: Methodologies, Architectures, and Reviews, p. 1–17.

- GUERRERO, R. P. *Differential evolution-based power dispatch algorithms*. Dissertação (Mestrado) — Unoversity of Puerto Rico, Mayaguez, 2004.
- HANSEN, M. P.; JASZKIEWICZ, A. *Evaluating quality of approximations to the non-dominated set*. Denmark, 1998.
- HOFMEYR, S.; FORREST, S. Immunity by design. In: *GECCO*. Orlando, Florida, USA: Morgan Kaufmann, 1999. p. 1289–1296.
- HOLLAND, J. H. Outline for a logical theory of adaptive systems. *J. ACM*, ACM, New York, NY, USA, v. 9, n. 3, p. 297–314, 1962. ISSN 0004-5411.
- HORN, J.; NAFPLIOTIS, N.; GOLDBERG, D. E. A niched pareto genetic algorithm for multiobjective optimization. In: *In Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*. Piscataway, NJ: IEEE, 1994. p. 82–87.
- HOU, Y. et al. Economic dispatch of power systems based on the modified particle swarm optimization algorithm. In: *IEEE PES. IEEE/PES Transmission and Distribution Conference and Exhibition*. China, 2005.
- HUANG, H.; LIU, M.; GU, X. A cultural algorithm based on multilayer belief spaces and its application in neural network fault classifier. In: *7th World Congress on Intelligent Control and Automation*. China: [s.n.], 2008. p. 3990–3995.
- IACOBAN, R.; REYNOLDS, R.; BREWSTER, J. Cultural swarms: assessing the impact of culture on social interaction and problem solving. In: *IEEE Swarm Intelligence Symposium*. Indiana, USA: [s.n.], 2003a. p. 212–219.
- IACOBAN, R.; REYNOLDS, R.; BREWSTER, J. Cultural swarms: modeling the impact of culture on social interaction and problem solving. In: *IEEE Swarm Intelligence Symposium*. Indiana, USA: [s.n.], 2003b. p. 205–211.
- IACOBAN, R.; REYNOLDS, R.; BREWSTER, J. Cultural swarms: modeling the impact of culture on social interaction and problem solving. In: *IEEE Swarm Intelligence Symposium*. Indiana, USA: [s.n.], 2003c. p. 205–211.
- ISHAK, S.; ABIDIN, A.; RAHMAN, T. Static var compensator planning using artificial immune system for loss minimisation and voltage improvement. In: *PECON. National Power and Energy Conference*. Malaysia, 2004. p. 41–45.
- ISHIGURO, A. et al. Moderationism in the immune system: Gait acquisition of a legged robot using the metadynamics function. In: *SMC. IEEE System, Man, and Cybernetics Conference*. San Diego, USA, 1998. p. 3827–3832.
- JONG, K. D. Evolutionary computation. *Wiley Interdisciplinary Reviews: Computational Statistics*, v. 1, p. 52–56, 2009.
- KARABOGA, D.; OKDEM, S. Asimple and global optimization algorithm for engineering problems: Differential evolution algorithm. *Journal of Electrical Engineering*, p. 53–60, 2004.

- KENNEDY, J.; EBERHART, R. C. Particle swarm optimization. In: ICNN. *The IEEE International Conference on Neural Networks*. Australia, 1995. p. 12–13.
- KEPHART, J. et al. Artificial immune systems and their applications. In: _____. USA: Springer-Verlag, 1998. cap. Blueprint for a Computer Immune System, p. 242–260.
- KEPHART, J.; SORKIN, G.; SWIMMER, M. An immune system for cyberspace. *Systems, Man, and Cybernetics*, v. 1, p. 879–884, 1997.
- KHAMSAWANG, S.; POTHIIYA, S.; BOONSENG, S. Distributed tabu search algorithm for solving the economic dispatch problem. In: TENCON. *TENCON*. Chiang Mai, Thailand, 2004. p. 484–487.
- KIM, D. H. Tuning of a pid controller using an artificial immune network model and local fuzzy set. In: IFA/NAFIPS. *International Conference IFSA World Congress and 20th NAFIPS*. Vancouver, Canada, 2001. v. 5, p. 2698–2703.
- KIM, J.; BENTLEY, P. J. Towards an artificial immune system for network intrusion detection: An investigation of dynamic clonal selection. In: *Congress on Evolutionary Computation*. Seoul, Korea: ACM, 2001. p. 1244–1252.
- KIM, J.; ONG, A.; OVERILL, R. E. Design of an artificial immune systems as a novel anomaly detector for combating financial fraud in retail sector. In: *Congress on Evolutionary Computation*. Canberra, Australia: ACM, 2003. p. 405–412.
- KING, R. T. F. A.; RUGHOOPUTH, H. C. S.; DEB, K. Evolutionary multi-objective environmental/economic dispatch: Stochastic versus deterministic approaches. In: *EMO*. Guanajuato, Mexico: [s.n.], 2005. p. 677–691.
- KNOWLES, J.; CORNE, D. Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Transactions on Evolutionary Computation*, v. 7, n. 2, p. 100–116, 2003.
- KNOWLES, J.; CORNE, D.; DEB, K. *Multiobjective Problem Solving from Nature: From Concepts to Applications*. USA: Springer, 2008.
- KNOWLES, J.; THIELE, L.; ZITZLER, E. *A tutorial on the performance assessment of stochastic multiobjective optimizers*. Zurich, 2006.
- KNOWLES, J. D.; CORNE, D. W. Approximating the nondominated front using the pareto archived evolution strategy. *Evolutionary Computation*, v. 8, n. 2, p. 149–172, 2000.
- KORTE, B.; VYGEN, J. *Combinatorial Optimization: Theory and Algorithms*. 4. ed. USA: Springer, 2008.
- KOZA, J. R. *Genetic programming: a paradigm for genetically breeding populations of computer programs to solve problems*. Stanford, CA, USA, 1990.
- KOZA, J. R. *Genetic Programming: a Paradigm for Genetically Breeding Computer Population of Computer Programs to Solve Problems*. London, UK: MIT Press, 1992.

- KOZA, J. R. *Genetic Programming II: On the Programming of Computers by Means of Natural Evolution*. London, UK: MIT Press, 1994.
- KOZA, J. R. et al. *Genetic Programming III: Darwinian Invention and Problem Solving*. San Mateo, California: Morgan Kaufmann, 1999.
- KOZA, J. R. et al. *Genetic Programming IV: Routine Human-Competitive Machine Intelligence*. USA: Kluwer Academic Publishers, 2003.
- KRAHE, B. *Personality and Social Psychology: Towards a Synthesis*. London: Sage, 2003.
- KUK, J. N. *Um Sistema Imune Fuzzy Cultural Aplicado ao Problema de Despacho Econômico de Energia Elétrica*. Dissertação (Mestrado) — Universidade Tecnológica Federal do Paraná, Curitiba, 2009.
- LACERDA, E. G. M.; CARVALHO, A. C. P. L. *Sistemas Inteligentes: Aplicações a Recursos Hídricos e Ciências Ambientais*. 1999. Capítulo: Introdução aos Algoritmo Genéticos.
- LAU, H. Y. K.; KO, A. An immuno robotic system for humanitarian search and rescue. In: ICARIS. *6th International Conference on Artificial Immune Systems*. Santos. RJ, Brasil, 2007. (Lecture Notes in Computer Science, v. 4628), p. 191–203.
- LAUMANNNS, M.; THIELE, L.; ZITZLER, E. An efficient, adaptive parameter variation scheme for methaheuristics based on the epsilon-constraint method. *European Journal of Operational Research*, v. 169, p. 932–942, 2006.
- LAW, H. Y. K.; WONG, V. W. K. Immunologic control framework for automated material handling. In: TIMMIS, J.; BENTLEY, P.; HART, E. (Ed.). *Second International Conference on Artificial Immune Systems*. Edinburgh, UK, 2004. (Lecture Notes in Computer Science, v. 2787), p. 57–68.
- LEE, D. et al. Immune-based framework for exploratory bio-information retrieval from the semantic web. In: TIMMIS, J.; BENTLEY, P.; HART, E. (Ed.). *Second International Conference on Artificial Immune Systems*. Edinburgh, UK, 2003. (Lecture Notes in Computer Science 2787), p. 128–135.
- LEE, S. et al. A new immunotronic approach to hardware fault detection using symbiotic evolution. In: IWINAC. *First International Work-Conference on the Interplay Between Natural and Artificial Computation*. Canary Islands, Spain, 2005. (Lecture Notes in Computer Science, v. 3562), p. 133–142.
- LEVENE, H. Contributions to probability and statistics: Essays in honor of harold hotelling. In: _____. Stanford, USA: Stanford University Press, 1960. cap. Robust tests for equality of variances, p. 278–292.
- LING, S. H. et al. Improved genetic algorithm for economic load dispatch with valve-point loadings. In: IECON. *The 29th Annual Conference of the IEEE Industrial Electronics Society*. Slovenia, 2003. p. 442–447.

- LIU, H. et al. A new solution to economic emission load dispatch using immune genetic algorithm. In: *IEEE Conference on Cybernetics and Intelligent Systems*. Bangkok, Thailand: [s.n.], 2006.
- MACKERELL, A. D. J. et al. The encyclopedia of computational chemistry. In: _____. London, UK: John Wiley & Sons, 1998. v. 1, cap. CHARMM: The Energy Function and Its Parameterization with an Overview of the Program, p. 271–277.
- MANIEZZO, V.; CARBONARO, A. Ant colony optimization: an overview. In: *III Metaheuristics International Conference*. Angra dos Reis, Brazil: [s.n.], 1999.
- MATZINGER, P. An innate sense of danger. *Seminar in Immunology*, vol. 5, p. pg. 399–415, 1998.
- MAY, P.; MANDER, K.; TIMMIS, J. Mutation testing: An artificial immune systems approach. In: *UK Software Testing Workshop*. York, UK: [s.n.], 2003a.
- MAY, P.; MANDER, K.; TIMMIS, J. Software vaccination: An artificial immune systems approach to mutation testing. In: TIMMIS, J.; BENTLEY, P.; HART, E. (Ed.). *International Conference on Artificial Immune Systems*. Edinburgh, UK: [s.n.], 2003b. p. 81–92.
- MENG, K. et al. Quantum-inspired particle swarm optimization for valve-point economic load dispatch. *IEEE Transactions on Power Systems*, v. 25, n. 1, p. 215–222, 2010.
- MICHAUD, S. R. et al. Protein structure prediction with ea immunological computation. In: *Genetic and Evolutionary Computation Conference*. California, USA: [s.n.], 2001. p. 1367–1374.
- MITCHELL, M.; TAYLOR, C. E. Evolutionary computation: An overview. *Annual Review of Ecology and Systematics*, v. 30, p. 593–616, 1999.
- MOORE, D. S. et al. *Practice of Business Statistics*. 1. ed. USA: W. H. Freeman, 2003.
- NOMAN, N.; IBA, H. Differential evolution for economic load dispatch problems. *Electric Power Systems Research*, v. 78, n. 3, p. 1322–1331, 2008.
- NUNES, L. Artificial immune systems as a novel soft computing paradigm. *Soft Computing Journal*, v. 7, n. 7, Julho 2003.
- OLIVEIRA, G. T. d. S. *Estudo e Aplicações da Evolução Diferencial*. Dissertação (Mestrado) — Universidade Federal de Uberlândia, Uberlândia, MG, 2006.
- ONWUBOLU, G. C.; BABU, B. V. *New Optimization Techniques in Engineering*. USA: Springer, 2004. (Studies in Fuzziness and Soft Computing).
- PACHECO, J. D. D. *Computational Power of Killers and Helpers in the Immune System*. Dissertação (Mestrado) — Faculdade de Ciências da Universidade de Lisboa, 2004.
- PALANICHAMY, C.; BABU, N. S. Analytical solution for combined economic and emissions dispatch. *Electric Power Systems Research*, v. 78, n. 7, p. 1129–1139, 2008.

- PARK, J.; JEONG, Y.; LEE, W. An improved particle swarm optimization for economic dispatch problems with non-smooth cost functions. In: *IEEE Power Engineering Society General Meeting*. Montreal, Que: [s.n.], 2006.
- PARSOPOULOS, K. E.; VRAHATIS, M. N. Particle swarm optimization method in multiobjective. In: *Symposium on Applied Computing*. Madrid, Spain: [s.n.], 2002a.
- PARSOPOULOS, K. E.; VRAHATIS, M. N. Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, v. 1, p. 235–306, 2002.
- PAULA, F. S. de; CASTRO, L. N. de; GEUS, P. L. de. An intrusion detection system using ideas from the immune system. In: *Proc. of the Int. Conf. on Evolutionary Computation (CEC 2004)*. Seattle, USA: [s.n.], 2004. p. 1059–1066.
- PEREZ-GUERRERO, R.; CEDENIO-MALDONADO, R. Economic power dispatch with non-smooth cost functions using differential evolution. In: *Proceedings of the 37th Annual North American Power Symposium*. Arlington, Virginia: [s.n.], 2005. p. 183–190.
- PRICE, K.; STORN, M.; LAMPINEN, J. A. *Differential Evolution: A Practical Approach to Global Optimization*. Berlin: Springer, 2005. (Natural Computing Series).
- QIN, H. et al. Multi-objective cultured differential evolution for generating optimal trade-offs in reservoir flood control operation. *Water Resources Management*, p. 1573–1650, 2010.
- RAHMAN, T. A.; YASIN, Z.; ABDULLAH, W. Artificial-immune-based for solving economic dispatch in power system. In: *PECON. National Power and Energy Conference*. Malaysia, 2004. p. 31–35.
- RAY, T.; LIEW, K. M. Society and civilization: an optimization algorithm based on the simulation of social behavior. *IEEE Transaction on Evolutionary Computation*, v. 7, n. 4, p. 386–396, 2003.
- RENFREW, A. C. Dynamical modeling and the study of change in archeology. In: _____. England: Edinburgh University Press, 1994. cap. Dynamic Modeling in Archeology: What, When, and Where?
- REYES-SIERRA, M.; COELLO, C. A. C. Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International Journal of Computational Intelligence Research*, v. 2, n. 3, p. 287–308, 2006.
- REYNOLDS, R.; CHUNG, C. Fuzzy approaches to acquiring experimental knowledge in cultural algorithms. In: *IEEE International Conference on Tools with Artificial Intelligence*. Newport Beach, CA: [s.n.], 1997. p. 260–267.
- REYNOLDS, R.; JIN, X. Regional schemata for real-valued constrained function optimization using cultural algorithms. *Journal of Natural Computing*, 2002.

- REYNOLDS, R.; PENG, B.; BREWSTER, J. Cultural swarms ii: virtual algorithm emergence. In: *Congress on Evolutionary Computation*. Canberra, Australia: ACM, 2003. v. 3, p. 1972–1979.
- REYNOLDS, R.; ZHU, S. Knowledge-based function optimization using fuzzy cultural algorithms with evolutionary programming. *IEEE Transactions on Systems, Man and Cybernetics - Part B*, v. 31, n. 1, p. 1–18, 2001.
- REYNOLDS, R. G. An introduction to cultural algorithm. In: *3rd Annual Congerence on Evolutionary Programming*. San Diego, California: [s.n.], 1994.
- REYNOLDS, R. G. Advances in evolutionary computation. In: _____. USA: McGraw Hill Press, 1999a. cap. An Overview of Cultural Algorithms.
- REYNOLDS, R. G. New ideas in optimization. In: _____. USA: McGraw-Hill, 1999b. cap. Cultural Algorithms: Theory and Applications, p. 367–377.
- REYNOLDS, R. G. *Cultural and Social Evolution in Dynamic Environments*. 2001. CASOS 2001.
- REYNOLDS, R. G. Knowledge swarms and cultural evolution. In: *Proceedings of American Anthropological Association Annual Meeting*. London, UK: Waveland Press, 2001b.
- REYNOLDS, R. G. *Tutorial on Cultural Algorithms*. Indiana, USA: IEEE Press, 2003. IEEE Swarm Intelligence Symposium.
- REYNOLDS, R. G.; ALI, M.; ALOMARI, R. S. Optimization problem solving using predator/prey games and cultural algorithms. In: *IEEE Symposium on Computational Intelligence and Games*. Tahoe: IEEE Press, 2006. p. 119–125.
- REYNOLDS, R. G.; ALI, M. Z. Embedding a social fabric component into cultural algorithms toolkit for an enhanced knowledge-driven engineering optimization. *International Journal of Intelligent Computing and Cybernetics*, v. 1, n. 4, p. 563–597, 2008.
- REYNOLDS, R. G.; ALI, M. Z. Mining the social fabric of archaic urban centers with cultural algorithms. *IEEE Computer*, v. 41, n. 1, p. 64–72, 2008.
- REYNOLDS, R. G.; ALI, M. Z. The social fabric approach as an approach to knowledge integration in cultural algorithms. In: *IEEE Congress on Evolutionary Computation*. Hong Kong, China: IEEE Press, 2008c. p. 4200–4207.
- REYNOLDS, R. G.; PENG, B. Cultural algorithms: Modeling of how cultures learn to solve problems. In: *ICTAI*. Boca Raton, FL, USA: IEEE Computer Society, 2004. p. 166–172.
- REYNOLDS, R. G.; PENG, B. Cultural algorithms: computational modeling of how cultures learn to solve problems: an engineering example. *Cybernetics and Systems*, v. 36, n. 8, p. 753–771, 2005.

- REYNOLDS, R. G.; PENG, B. Knowledge integration on-the-fly in swarm intelligent systems. In: *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence - ICTAI 2006*. Washington: IEEE, 2006a.
- REYNOLDS, R. G.; PENG, B. Knowledge integration on-the-fly in swarm intelligent systems. In: *IEEE International Conference on Tools with Artificial Intelligence*. Washington: IEEE Computer Society, 2006b. p. 197–210.
- REYNOLDS, R. G.; PENG, B. Knowledge integration on-the-fly in swarm intelligent systems. In: *ICTAI. IEEE International Conference on Tools with Artificial Intelligence*. Washington, DC: IEEE, 2006c. p. 197–210.
- REYNOLDS, R. G.; PENG, B.; ALI, M. Z. The role of culture in the emergence of decision-making roles: An example using cultural algorithms. *Complexity*, v. 13, n. 3, p. 27–42, 2008.
- REYNOLDS, R. G.; ZANONI, E. Why cultural evolution can proceed faster than biological evolution. In: *Proceedings of International Symposium on Simulating Societies*. Surrey, England: Kluwer Academic Publishers, 1992. p. 81–93.
- RIVERA, D. C.; BECERRA, R. L.; COELLO, C. A. C. Cultural algorithms, an alternative heuristic to solve the job shop scheduling problem. *Engineering Optimization*, v. 39, n. 1, p. 69–85, 2007.
- SALEEM, S. M. *Knowledge-Based Solution to Dynamic Optimization Problems using Cultural Algorithms*. Tese (Doutorado) — Wayne State University, Detroit, Michigan, 2001.
- SARAJANOVIC, S.; BOUDEC, J. L. *An Artificial Immune System Approach with Secondary Response for Misbehavior Detection in Mobile Ad-Hoc Networks*. Switzerland, 2003.
- SCHAFFER, J. D. Multiple objective optimization with vector evaluated genetic algorithms. In: *Proceedings of the 1st International Conference on Genetic Algorithms*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1985. p. 93–100. ISBN 0-8058-0426-9.
- SECKER, A.; FREITAS, A.; TIMMIS, J. A danger theory inspired approach to web mining. In: *Proceedings of the 2nd International Conference on Artificial Immune Systems*. Springer, 2003. (Lecture Notes in Computer Science, v. 2787), p. 156–167. Disponível em: <<http://www.cs.kent.ac.uk/people/rpg/ads3/AISEC.pdf>>.
- SECKER, A.; FREITAS, A.; TIMMIS, J. Towards a danger theory inspired artificial immune system for web mining. In: SCIME, A. (Ed.). *Web Mining: applications and techniques*. Idea Group, 2005. p. 145–168. ISBN 1591404150. Disponível em: <<http://www.cs.kent.ac.uk/pubs/2005/1947>>.
- SELVAKUMAR, A.; THANUSHKODI, K. Optimization using civilized swarm: solution to economic dispatch with multiple minima. *Electric Power Systems Research*, v. 79, n. 1, p. 8–16, 2009.

- SELVAKUMAR, A. I.; THANUSHKODI, K. New particle swarm optimization solution to nonconvex economic dispatch problems. *IEEE Transactions on Power Systems*, v. 22, n. 1, p. 42–51, 2007.
- SINHA, N.; PURKAYASTHA, B. Pso embedded evolutionary programming technique for nonconvex economic load dispatch. In: IEEE PES. *IEEE PES - Power Systems Conference and Exposition*. Brasil, 2004. p. 66–71.
- SISKOS, Y.; SPYRIDAKOS, A. Intelligent multicriteria decision support: Overview and perspectives. *European Journal of Operational Research*, v. 116, n. 2, p. 236–246, 1999.
- SOUSA, J. S. de et al. An immune-evolutionary algorithm for multiple rearrangements of gene expression data. *Genetic Programming and Evolvable Machines*, v. 5, p. 157–179, 2004.
- SPEARS, W. M. et al. An overview of evolutionary computation. In: *ECML '93: Proceedings of the European Conference on Machine Learning*. London, UK: Springer-Verlag, 1993. p. 442–459. ISBN 3-540-56602-3.
- SPENDLEY, W.; HEXT, G.; HIMSWORTH, F. Sequential application of simplex designs in optimisation and evolutionary operation. *Technometrics*, n. 4, p. 441–461, 1962.
- SRINIVAS, N.; DEB, K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, v. 2, n. 3, p. 221–248, 1994.
- STEPNEY, S. et al. Conceptual frameworks for artificial immune systems. *International Journal of Unconventional Computing*, 2005.
- STERNBERG, M.; REYNOLDS, R. Using cultural algorithms to support re-engineering of rule-based expert systems in dynamic performance environments: a case study in fraud detection. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 4, p. 225–243, 1997.
- STORN, R.; PRICE, K. *Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces*. Berkeley, 1995. Disponível em: <http://www.icsi.berkeley.edu/~storn/TR-95-012.pdf>.
- STROGATZ, S. *Nonlinear dynamics and chaos*. [S.l.]: Perseus Publishing, 2000.
- SU, C.-T.; LIN, C.-T. New approach with a hopfield modeling framework to economic dispatch. *IEEE Transactions on Power Systems*, v. 15, n. 2, p. 541–545, 2000.
- SUNDARAM, R. K. *A First Course in Optimization Theory*. Cambridge, UK: Cambridge University Press, 1996.
- TANG, W.; LI, Y. Constrained optimization using triple spaces cultured genetic algorithm. In: *The 4th International Conference on Natural Computation*. Jinan, China: [s.n.], 2008.
- TIMMIS, J. *An Introduction to Artificial Immune Systems*. Dezembro 2001. Tutorial - ES2001.

- TIMMIS, J. *An Introduction to Artificial Immune Systems*. 2004. Tutorial - ICARIS 2004.
- TIMMIS, J. et al. An overview of artificial immune systems. In: PATON, R. et al. (Ed.). *"Computation in Cells and Tissues: Perspectives and Tools for Thought"*. Springer, 2004, (Natural Computation Series). p. 51–86. ISBN 3540003584. Disponible em: <http://www.cs.kent.ac.uk/pubs/2004/1994>.
- TWYXCROSS, J.; AICKELIN, U. Towards a conceptual framework for innate immunity. In: *International Conference on Artificial Immune Systems*. Alberta, Canada: Springer, 2005.
- VARGAS, P. A. et al. Implementation of an immuno-genetic network on a real khepera ii robot. In: CEC. *Congress on Evolutionary Computation*. Newport Beach, CA, USA, 2003. v. 1, p. 420–426.
- VENKATESH, P.; GNANADASS, R.; PADHY, N. P. Comparison and application of evolutionary programming techniques to combined economic emission dispatch with line flow constraints. *IEEE Transactions on Power Systems*, v. 18, n. 2, p. 688–697, 2003.
- VICTOIRE, T.; JEYAKUMAR, A. Hybrid pso-sqp for economic dispatch with valve-point effect. *Electric Power Systems Research*, v. 71, n. 1, p. 51–59, 2004.
- VLACHOGIANNIS, J. G.; LEE, K. Y. Economic load dispatch - a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based pso. *IEEE Transactions on Power Systems*, v. 24, n. 2, p. 991–1001, 2009.
- WALKER, J. H.; GARRETT, S. M. Dynamic function optimisation: Comparing the performance of clonal selection and evolution strategies. In: TIMMIS, J.; BENTLEY, P.; HART, E. (Ed.). *Proc. Of the Second International Conference on Artificial Immune Systems*. Edinburgh, UK, 2003. (Lecture Notes in Computer Science, v. 2787), p. 273–284.
- WANG, F.; LIU, D. *Advances in Computational Intelligence: Theory and Applications*. Hackensack, NJ, USA: World Scientific Publishing, 2006.
- WANG, L.; HIRSBRUNNER, B. Immune mechanism based computer security design. In: ICMLC. *International Conference on Machine Learning and Cybernetics*. Beijing, China, 2002. v. 4, p. 1887–1893.
- WANG, L.; SINGH, C. Stochastic economic emission load dispatch through a modified swarm optimization algorithm. *Electric Power Systems Research*, v. 78, n. 1, p. 1466–1476, 2008.
- WANG, S. K.; CHIOU, J. P.; LIU, C. W. Non-smooth/nonconvex economic dispatch by a novel hybrid differential evolution algorithm. *IET Generation, Transmission & Distribution*, v. 1, n. 5, p. 793–803, 2007.
- WANG, X. L.; MAHFOUF, M. Acsamo: An adaptive multiobjective optimization algorithm using the clonal selection principle. In: *2nd European Symposium on Nature-Inspired Smart Information Systems*. Tenerife, Spain: ACM, 2006. p. 1–12.

- WATKINS, A. *Exploiting Immunological Metaphors in the Development of Serial, Parallel and Distributed Learning Algorithms*. Tese (Doutorado) — Computing Laboratory, University of Kent, Canterbury, UK, March 2005. Disponível em: <http://www.cs.kent.ac.uk/pubs/2005/2178>.
- WATKINS, A.; TIMMIS, J. Artificial Immune Recognition System (AIRS): Revisions and Refinements. In: TIMMIS, J.; BENTLEY, P. (Ed.). *1st International Conference on Artificial Immune Systems*. University of Kent at Canterbury: University of Kent at Canterbury Printing Unit, 2002. p. 173–181. ISBN 1902671325. Disponível em: <http://www.cs.ukc.ac.uk/pubs/2002/1505>.
- WATKINS, A.; TIMMIS, J.; BOGGESS, L. Artificial Immune Recognition System (AIRS): An Immune Inspired Supervised Machine Learning Algorithm. *Genetic Programming and Evolvable Machines*, Kluwer, v. 5, n. 1, March 2004. Disponível em: <http://www.cs.kent.ac.uk/pubs/2004/1634>.
- WATKINS, A.; TIMMIS, T.; BOGGESS, L. Artificial immune recognition system (airs): An immune inspired supervised machine learning algorithm. *Genetic Programming and Evolvable Machines*, v. 5, n. 1, p. 291–317, 2004.
- WHITLEY, D. *An Overview of Evolutionary Algorithms: Practical Issues and Common Pitfalls*. 2004.
- WOOD, A. J.; WOLLENBERG, B. F. *Power Generation, Operation, and Control*. London, UK: John Wiley & Sons, 1996.
- WOOD, M. Bootstrapped confidence intervals as an approach to statistical inference. *Organizational Research Methods*, v. 8, n. 4, p. 454–470, 2005.
- YOUNG, H. P. New palgrave dictionary of economics. In: _____. 2. ed. London: Macmillan, 2008. cap. Social Norms.
- YUAN, X. et al. A chaotic hybrid cultural algorithm for constrained optimization. In: *Proceedings of the 2008 Second International Conference on Genetic and Evolutionary Computing*. Huibei Province, China: [s.n.], 2008. p. 307 – 310.
- ZHANG, X. et al. Immune multiobjective optimization algorithm for unsupervised feature selection. In: *10th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*. Singapore: [s.n.], 2006. p. 484–494.
- ZHANG, Z. Immune optimization algorithm for constrained nonlinear multiobjective optimization problems. *Applied Soft Computing Journal*, v. 7, n. 3, p. 840–857, 2007.
- ZHOU, J.; DASGUPTA D.; ZHILING, Y.; HONGMEI, T. Analysis of dental images using artificial immune systems. In: CEC. *IEEE Congress on Evolutionary Computation*. [S.l.]: Idea Group, 2006. p. 528–535.
- ZITZLER, E. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. Tese (Doutorado) — Federal Institute of Technology Zurich, 1999.

- ZITZLER, E.; LAUMANNNS, M.; BLEULER, S. A Tutorial on Evolutionary Multiobjective Optimization. In: GANDIBLEUX, X. et al. (Ed.). *Metaheuristics for Multiobjective Optimisation*. New York: Springer-Verlag, 2004. (Lecture Notes in Economics and Mathematical Systems).
- ZITZLER, E.; LAUMANNNS, M.; THIELE, L. Spea2: Improving the strength pareto evolutionary algorithm. In: *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*. Athens, Greece: CIMNE, 2001. p. 95–100.
- ZITZLER, E.; THIELE, L. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, v. 3, n. 4, p. 257–271, 1999.
- ZITZLER, E. et al. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, v. 7, n. 2, p. 117–132, 2003.

APÊNDICE A -- Coeficientes Utilizados pelos Estudos de Caso

A.1 Dados dos Estudos de Caso Mono-objetivo

Tabela 64: Mono-objetivo - Estudo de Caso 1 e 2 - Coeficientes do custo de combustível.

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

Tabela 65: Mono-objetivo - Estudo de Caso 3 - Coeficientes do custo de combustível.

Ger	P_i^{min}	P_i^{max}	a	b	c	e	f
1	36	114	0,00690	6,73	94,705	100	0,084
2	36	114	0,00690	6,73	94,705	100	0,084
3	60	120	0,02028	7,07	309,54	100	0,084
4	80	190	0,00942	8,18	369,03	150	0,063
5	47	97	0,01140	5,35	148,89	120	0,077
6	68	140	0,01142	8,05	222,33	100	0,084
7	110	300	0,00357	8,03	278,71	200	0,042
8	135	300	0,00492	6,99	391,98	200	0,042
9	135	300	0,00573	6,60	455,76	200	0,042
10	130	300	0,00605	12,90	722,82	200	0,042
11	94	375	0,00515	12,90	635,20	200	0,042
12	94	375	0,00569	12,80	654,69	200	0,035
13	125	500	0,00421	12,50	913,40	300	0,035
14	125	500	0,00752	8,84	1760,4	300	0,035
15	125	500	0,00708	9,15	1728,3	300	0,035
16	125	500	0,00708	9,15	1728,3	300	0,035
17	220	500	0,00313	7,97	647,85	300	0,035
18	220	500	0,00313	7,95	649,69	300	0,035
19	242	550	0,00313	7,97	647,83	300	0,035
20	242	550	0,00313	7,97	647,81	300	0,035
21	254	550	0,00298	6,63	785,96	300	0,035
22	254	550	0,00298	6,63	785,96	300	0,035
23	254	550	0,00284	6,66	794,53	300	0,035
24	254	550	0,00284	6,66	794,53	300	0,035
25	254	550	0,00277	7,10	801,32	300	0,035
26	254	550	0,00277	7,10	801,32	300	0,035
27	10	150	0,52124	3,33	1055,1	120	0,077
28	10	150	0,52124	3,33	1055,1	120	0,077
29	10	150	0,52124	3,33	1055,1	120	0,077
30	47	97	0,01140	5,35	148,89	120	0,077
31	60	190	0,00160	6,43	222,92	150	0,063
32	60	190	0,00160	6,43	222,92	150	0,063
33	60	190	0,00160	6,43	222,92	150	0,063
34	90	200	0,00010	8,95	107,87	200	0,042
35	90	200	0,00010	8,62	116,58	200	0,042
36	90	200	0,00010	8,62	116,58	200	0,042
37	25	110	0,01610	5,88	307,45	80	0,098
38	25	110	0,01610	5,88	307,45	80	0,098
39	25	110	0,01610	5,88	307,45	80	0,098
40	242	550	0,00313	7,97	647,83	300	0,035

Tabela 66: Mono-objetivo - Estudo de Caso 4 - Coeficientes do custo de combustível.

Gen	P_i^{min}	P_i^{max}	a	b	c
1	150	600	0,00068	18,19	1000
2	50	200	0,00071	19,26	970
3	50	200	0,00650	19,80	600
4	50	200	0,00500	19,10	700
5	50	160	0,00738	18,10	420
6	20	100	0,00612	19,26	360
7	25	125	0,00790	17,14	490
8	50	150	0,00813	18,92	660
9	50	200	0,00522	18,27	765
10	30	150	0,00573	18,92	770
11	100	300	0,00480	16,69	800
12	150	500	0,00310	16,76	970
13	40	160	0,00850	17,36	900
14	20	130	0,00511	18,70	700
15	25	185	0,00398	18,70	450
16	20	80	0,07120	14,26	370
17	30	85	0,00890	19,14	480
18	30	120	0,00713	18,92	680
19	40	120	0,00622	18,47	700
20	30	100	0,00773	19,79	850

Tabela 67: Mono-objetivo - Estudo de Caso 4 - Coeficientes de perda na transmissão de energia - Matriz B - Devem ser multiplicados por $1 * 10^{-5}$

8,70	0,43	-4,61	0,36	0,32	-0,66	0,96	-1,60	0,80	-0,10	3,60	0,64	0,79	2,10	1,70	0,80	-3,20	0,70	0,48	-0,70
0,43	8,30	-0,97	0,22	0,75	-0,28	5,04	1,70	0,54	7,20	-0,28	0,98	-0,46	1,30	0,80	-0,20	0,52	-1,70	0,80	0,20
-4,61	-0,97	9,00	-2,00	0,63	3,00	1,70	-4,30	3,10	-2,00	0,70	-0,77	0,93	4,60	-0,30	4,20	0,38	0,70	-2,00	3,60
0,36	0,22	-2,00	5,30	0,47	2,62	-1,96	2,10	0,67	1,80	-0,45	0,92	2,40	7,60	-0,20	0,70	-1,00	0,86	1,60	0,87
0,32	0,75	0,63	0,47	8,60	-0,80	0,37	0,72	-0,90	0,69	1,80	4,30	-2,80	-0,70	2,30	3,60	0,80	0,20	-3,00	0,50
-0,66	-0,28	3,00	2,62	-0,80	11,8	-4,90	0,30	3,00	-3,00	0,40	0,78	6,40	2,60	-0,20	2,10	-0,40	2,30	1,60	-2,10
0,96	5,04	1,70	-1,96	0,37	-4,90	8,24	-0,90	5,90	-0,60	8,50	-0,83	7,20	4,80	-0,90	-0,10	1,30	0,76	1,90	1,30
-1,60	1,70	-4,30	2,10	0,72	0,30	-0,90	1,20	-0,96	0,56	1,60	0,80	-0,40	0,23	0,75	-0,56	0,80	-0,30	5,30	0,80
0,80	0,54	3,10	0,67	-0,90	3,00	5,90	-0,96	0,93	-0,30	6,50	2,30	2,60	0,58	-0,10	0,23	-0,30	1,50	0,74	0,70
-0,10	7,20	-2,00	1,80	0,69	-3,00	-0,60	0,56	-0,30	0,99	-6,60	3,90	2,30	-0,30	2,80	-0,80	0,38	1,90	0,47	-0,26
3,60	-0,28	0,70	-0,45	1,80	0,40	8,50	1,60	6,50	-6,60	10,7	5,30	-0,60	0,70	1,90	-2,60	0,93	-0,60	3,80	-1,50
0,64	0,98	-0,77	0,92	4,30	0,78	-0,83	0,80	2,30	3,90	5,30	8,00	0,90	2,10	-0,70	5,70	5,40	1,50	0,70	0,10
0,79	-0,46	0,93	2,40	-2,80	6,40	7,20	-0,40	2,60	2,30	-0,60	0,90	11,00	0,87	-1,00	3,60	0,46	-0,90	0,60	1,50
2,10	1,30	4,60	7,60	-0,70	2,60	4,80	0,23	0,58	-0,30	0,70	2,10	0,87	3,80	0,50	-0,70	1,90	2,30	-0,97	0,90
1,70	0,80	-0,30	-0,20	2,30	-0,20	-0,90	0,75	-0,10	2,80	1,90	-0,70	-1,00	0,50	11,0	1,90	-0,80	2,60	2,30	-0,10
0,80	-0,20	4,20	0,70	3,60	2,10	-0,10	-0,56	0,23	-0,80	-2,60	5,70	3,60	-0,70	1,90	10,8	2,50	-1,80	0,90	-2,60
-3,20	0,52	0,38	-1,00	0,80	-0,40	1,30	0,80	-0,30	0,38	0,93	5,40	0,46	1,90	-0,80	2,50	8,70	4,20	-0,30	0,68
0,70	-1,70	0,70	0,86	0,20	2,30	0,76	-0,30	1,50	1,90	-0,60	1,50	-0,90	2,30	2,60	-1,80	4,20	2,20	0,16	-0,30
0,48	0,80	-2,00	1,60	-3,00	1,60	1,90	5,30	0,74	0,47	3,80	0,70	0,60	-0,97	2,30	0,90	-0,30	0,16	7,60	0,69
-0,70	0,20	3,60	0,87	0,50	-2,10	1,30	0,80	0,70	-0,26	-1,50	0,10	1,50	0,90	-0,10	-2,60	0,68	-0,30	0,69	7,00

A.2 Dados dos Estudos de Caso Multiobjetivo

Tabela 68: Multiobjetivo - Estudo de Caso 1, 2 e 3 - Coeficientes do custo de combustível.

Gerador	P_i^{min}	P_i^{max}	a_i	b_i	c_i
1	0,05	0,50	100	200	10
2	0,05	0,60	120	150	10
3	0,05	1,00	40	180	20
4	0,05	1,20	60	100	10
5	0,05	1,00	40	180	20
6	0,05	0,60	100	150	10

Tabela 69: Multiobjetivo - Estudo de Caso 1, 2 e 3 - Coeficientes da emissão de poluentes.

Gerador	α_i	β_i	γ_i	ζ_i	λ_i
1	$6,490e^{-2}$	$-5,544e^{-2}$	$4,091e^{-2}$	$2,0e^{-4}$	2,857
2	$5,638e^{-2}$	$-6,047e^{-2}$	$2,543e^{-2}$	$5,0e^{-4}$	3,333
3	$4,586e^{-2}$	$-5,094e^{-2}$	$4,258e^{-2}$	$1,0e^{-6}$	8,000
4	$3,380e^{-2}$	$-3,550e^{-2}$	$5,326e^{-2}$	$2,0e^{-3}$	2,000
5	$4,586e^{-2}$	$-5,094e^{-2}$	$4,258e^{-2}$	$1,0e^{-6}$	8,000
6	$5,151e^{-2}$	$-5,555e^{-2}$	$6,131e^{-2}$	$1,0e^{-5}$	6,667

Tabela 70: Multiobjetivo - Estudo de Caso 2 - Coeficientes de perda na transmissão de energia - Matriz B, Vetor B_0 e Coeficiente B_{00}

B =	0,1382	-0,0299	0,0044	-0,0022	-0,0010	-0,0008
	-0,0299	0,0487	-0,0025	0,0004	0,00016	0,0041
	0,0044	-0,0025	0,0182	-0,0070	-0,0066	-0,0066
	-0,0022	0,0004	-0,0070	0,0137	0,0050	0,0033
	-0,0010	0,0016	-0,0066	0,0050	0,0109	0,0005
	-0,0008	0,0041	-0,0066	0,0033	0,0005	0,0244
$B_0 =$	-0,0535	0,0030	-0,0085	0,0004	0,0001	0,0015
$B_{00} =$	0,000986					

Tabela 71: Multiobjetivo - Estudo de Caso 3 - Coeficientes de perda na transmissão de energia - Matriz B

0,1382	-0,0299	0,0044	-0,0022	-0,0010	-0,0008
-0,0299	0,0487	-0,0025	0,0004	0,00016	0,0041
0,0044	-0,0025	0,0182	-0,0070	-0,0066	-0,0066
-0,0022	0,0004	-0,0070	0,0137	0,0050	0,0033
-0,0010	0,0016	-0,0066	0,0050	0,0109	0,0005
-0,0008	0,0041	-0,0066	0,0033	0,0005	0,0244

Tabela 72: Multiobjetivo - Estudo de Caso 4 - Coeficientes do custo de combustível.

Gerador	P_i^{min}	P_i^{max}	a_i	b_i	c_i
1	10	125	0,15247	38,53973	756,79886
2	10	150	0,10587	46,15916	451,32513
3	40	250	0,02803	40,39655	1049,32513
4	35	210	0,03546	38,30553	1243,5311
5	130	325	0,02111	36,32782	1658,5696
6	125	315	0,01799	38,27041	1356,65920

Tabela 73: Multiobjetivo - Estudo de Caso 4 - Coeficientes da emissão de poluentes.

Gerador	α_i	β_i	γ_i
1	0,00419	0,32767	13,85932
2	0,00419	0,32767	13,85932
3	0,00683	-0,54551	40,2669
4	0,00683	-0,54551	40,2669
5	0,00461	-0,51116	42,89553
6	0,00461	-0,51116	42,89553

Tabela 74: Multiobjetivo - Estudo de Caso 4 - Coeficientes de perda na transmissão de energia - Matriz B

0,000091	0,000091	0,000091	0,000031	0,000031	0,000029
0,000091	0,000091	0,000091	0,000031	0,000031	0,000029
0,000091	0,000091	0,000091	0,000031	0,000031	0,000029
0,000031	0,000031	0,000031	0,000062	0,000062	0,000028
0,000031	0,000031	0,000031	0,000062	0,000062	0,000028
0,000029	0,000029	0,000029	0,000028	0,000028	0,000072

Tabela 75: Multiobjetivo - Estudo de Caso 5 e 6 - Coeficientes do custo de combustível.

Gerador	P_i^{min}	P_i^{max}	a_i	b_i	c_i
1	50	1000	150	1,89	0,0050
2	50	1000	115	2,00	0,0055
3	50	1000	40	3,50	0,0060
4	50	1000	122	3,15	0,0050
5	50	1000	125	3,05	0,0050
6	50	1000	70	2,75	0,0070
7	50	1000	70	3,45	0,0070
8	50	1000	70	3,45	0,0070
9	50	1000	130	2,45	0,0050
10	50	1000	130	2,45	0,0050
11	50	1000	135	2,35	0,0055
12	50	1000	200	1,30	0,0045
13	50	1000	70	3,45	0,0070
14	50	1000	45	3,89	0,0060

Tabela 76: Multiobjetivo - Estudo de Caso 5 - Coeficientes da emissão de poluentes.

Gerador	α_i	β_i	γ_i
1	0,016	-1,500	23,333
2	0,031	-1,820	21,022
3	0,013	-1,249	22,050
4	0,012	-1,355	22,983
5	0,020	-1,900	21,313
6	0,007	0,805	21,900
7	0,015	-1,401	23,001
8	0,018	-1,800	24,003
9	0,019	-2,000	25,121
10	0,012	-1,360	22,990
11	0,033	-2,100	27,010
12	0,018	-1,800	25,101
13	0,018	-1,810	24,313
14	0,030	-1,921	27,119

Tabela 77: Multiobjetivo - Estudo de Caso 5 e 6 - Coeficientes de perda na transmissão de energia - Matriz B, Vetor B_0 e Coeficiente B_{00}

0,042741	0,030108	0,019242	0,021506	-0,00288	-0,00400	-0,00447	-0,00272	-0,00323	-0,00694	-0,00745	-0,01952	-0,01217	-0,01718
0,030108	0,037946	0,020710	0,020912	-0,00363	-0,00525	-0,00448	-0,00366	-0,00359	-0,00695	-0,01018	-0,02004	-0,01844	-0,02057
0,019242	0,02071	0,026780	0,024696	-0,00247	-0,00378	-0,00298	-0,00239	-0,00231	-0,00467	-0,00786	-0,01583	-0,01529	-0,01668
0,021506	0,020912	0,024696	0,024393	-0,00232	-0,00352	-0,00309	-0,00223	-0,0023	-0,00475	-0,00715	-0,016	-0,1346	-0,01588
-0,00288	-0,00363	-0,00247	-0,00232	0,009543	0,003659	0,002951	0,003740	0,003341	0,002486	0,001192	-0,00279	-0,00288	-0,00331
-0,00400	-0,00525	-0,00378	-0,00352	0,003659	0,010678	0,005763	0,003740	0,003341	0,002486	0,001192	-0,00279	-0,00288	-0,00331
-0,00447	-0,00448	-0,00378	-0,00352	0,002951	0,005763	0,008092	0,003370	0,003566	0,003054	0,001252	-0,00252	-0,00192	-0,00272
-0,00272	-0,00366	-0,00239	-0,00223	0,003116	0,00374	0,00337	0,003876	0,003746	0,002934	0,002063	-0,00152	-0,00142	-0,00188
-0,00323	-0,00359	-0,00231	-0,0023	0,004207	0,003341	0,003566	0,003746	0,005404	0,002869	0,001477	-0,00225	-0,00189	-0,00254
-0,00694	-0,00695	-0,00467	-0,00475	0,002066	0,002486	0,003054	0,002934	0,002869	0,006738	0,003054	0,001212	0,001331	0,000955
-0,00745	-0,01018	-0,00786	-0,00715	0,000366	0,001192	0,001293	0,002063	0,001477	0,003054	0,008576	0,006171	0,008179	0,007260
-0,01952	-0,02004	-0,01583	-0,01600	-0,00365	-0,00279	-0,00252	-0,00152	-0,00225	0,001212	0,006171	0,036153	0,018390	0,020017
-0,01217	-0,01844	-0,01529	-0,01346	-0,00381	-0,00288	-0,00192	-0,00142	-0,00189	0,001331	0,008179	0,018390	0,033117	0,029414
-0,01718	-0,02057	-0,01668	-0,01588	-0,00424	-0,00331	-0,00272	-0,00188	-0,00254	0,000955	0,007260	0,020017	0,029414	0,041297
$B_0 =$	-0,538520	-0,283225	-0,1929400	-0,26424	0,017755	0,021917	0,012216	0,014007	0,0044072	0,032732	0,217820	0,032560	0,155630
$B_{00} =$	0,028378												

Tabela 78: Multiobjetivo - Estudo de Caso 6 - Coeficientes da emissão de poluentes.

Gerador	α_i	β_i	γ_i
1	23,333	-1,500	0,016
2	21,022	-1,820	0,031
3	22,050	-1,249	0,013
4	22,983	-1,355	0,012
5	21,313	-1,900	0,020
6	21,900	0,805	0,007
7	23,001	-1,401	0,015
8	24,003	-1,800	0,018
9	25,121	-2,000	0,019
10	22,990	-1,360	0,012
11	27,010	-2,100	0,033
12	25,101	-1,800	0,018
13	24,313	-1,810	0,018
14	27,119	-1,921	0,030