

UNIVERSIDADE FEDERAL DO PARANÁ

JOSÉ MÁRCIO FACHIN



ABORDAGEM DE EVOLUÇÃO DIFERENCIAL HÍBRIDA AUTO ADAPTATIVA E  
APLICAÇÃO NA REALIZAÇÃO DE CALIBRAÇÃO AUTOMÁTICA DE FUNÇÕES DE  
SOFTWARES DE MOTORES AUTOMOTIVOS

CURITIBA

2014

JOSÉ MÁRCIO FACHIN

ABORDAGEM DE EVOLUÇÃO DIFERENCIAL HÍBRIDA AUTO ADAPTATIVA E  
APLICAÇÃO NA REALIZAÇÃO DE CALIBRAÇÃO AUTOMÁTICA DE FUNÇÕES DE  
*SOFTWARES* DE MOTORES AUTOMOTIVOS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, Área de Concentração Sistema Eletrônicos – Controle e Automação, Departamento de Engenharia Elétrica, Setor de Tecnologia, Universidade Federal do Paraná, como parte das exigências para a obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Leandro dos Santos Coelho

CURITIBA

2014



### ATA DE DEFESA DE MESTRADO

Aos oito dias do mês de dezembro de 2014, na Sala PK 07 do Departamento de Engenharia Elétrica, foi instalada pela Prof<sup>ª</sup>. Dr<sup>ª</sup>. Thelma Solange Piazza Fernandes, Coordenadora do Programa de Pós-Graduação em Engenharia Elétrica, a Banca Examinadora para a centésima sexagésima quarta Dissertação de Mestrado do PPGEE, na Área de Concentração em **SISTEMAS ELETRÔNICOS**. Estiveram presentes no ato, além da Coordenadora do Curso de Pós-Graduação, professores, alunos e visitantes.

A Banca Examinadora, atendendo determinação do Colegiado do Programa de Pós-Graduação em Engenharia Elétrica, ficou constituída pelos professores doutores **Leandro dos Santos Coelho (Orientador - UFPR)**; **Myriam Regattieri de Biase da Silva (UTFPR)**; **Gilberto Reynoso Meza (PUCPR)**; **Roman Kuiava (UFPR)** e **Gideon Villar Leandro (UFPR)**.

Às 09:00 horas, a banca iniciou os trabalhos, convidando o candidato **JOSÉ MARCIO FACHIN** a fazer a apresentação da dissertação intitulada "**ABORDAGEM DA EVOLUÇÃO DIFERENCIAL HÍBRIDA AUTO ADAPTATIVA E APLICAÇÃO NA REALIZAÇÃO DE CALIBRAÇÃO AUTOMÁTICA DE FUNÇÕES DE SOFTWARES DE MOTORES AUTOMOTIVOS**". Encerrada a apresentação, iniciou-se a fase de arguição pelos membros participantes.

Tendo em vista a dissertação e a arguição, a banca decidiu pela:

**APROVAÇÃO**

**NÃO APROVAÇÃO** do candidato, (de acordo com a determinação da Resolução 63/03-CEPE).

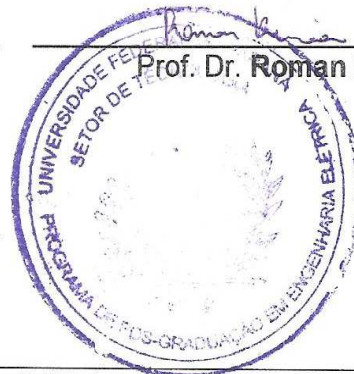
Curitiba, 08 de dezembro 2014.

Prof. Dr. Leandro dos Santos Coelho

Prof<sup>ª</sup>. Dr<sup>ª</sup>. Myriam Regattieri de Biase da Silva

Prof. Dr. Gilberto Reynoso Meza

Prof. Dr. Gideon Villar Leandro



Prof. Dr. Roman Kuiava

## **AGRADECIMENTOS**

A Deus por ter me dado saúde e força para superar todas as dificuldades.

Ao meu orientador professor Leandro dos Santos Coelho por seu incentivo, orientação, ensinamentos e todo o seu auxílio no desenvolvimento deste trabalho.

À minha família, em especial a minha mãe Maria Ezildinha e ao meu falecido pai José Fachin pois tudo o que sou hoje devo a eles. São exemplos de vida que sempre busco seguir.

À minha filha Sofia, o meu maior presente, cujo sorriso sempre me dá forças para seguir em frente e superar qualquer adversidade que tenha que enfrentar.

E principalmente a minha querida esposa Priscila, por seu companheirismo, amizade, amor, paciência e apoio incondicional.

E, finalmente, a todos que direta ou indiretamente contribuíram de alguma forma para que este objetivo fosse alcançado, o meu muito obrigado.

## RESUMO

A criação de normas de emissões de gases veiculares cada vez mais restritivas, a necessidade de redução de consumo de combustíveis, requisitos rigorosos de segurança e a busca para atender as demandas de clientes cada vez mais exigentes, fizeram com que a complexidade dos *softwares* automotivos crescesse exponencialmente ao longo dos últimos anos, o que acarretou em um aumento significativo no número de parâmetros a serem calibrados. Consequentemente, o processo de calibração demandará mais esforço e tempo para que os motores automotivos possam atender aos requisitos do projeto. Neste sentido, algoritmos de otimização baseados na evolução diferencial são uma ferramenta útil na realização de calibração automática dos parâmetros dos *softwares* de motores automotivos devido a sua eficiência na busca de soluções adequadas para problemas complexos de otimização. Portanto, o objetivo principal desta dissertação é o desenvolvimento de uma nova versão de um algoritmo baseado na evolução diferencial para solução de problemas mono-objetivo sujeito apenas a restrições laterais e a aplicação deste algoritmo na realização de calibração automática de algumas funções de *softwares* utilizados para o controle e funcionamento de motores automotivos. Nesta dissertação são apresentadas duas versões de um algoritmo desenvolvido baseando-se nas idéias da evolução diferencial, o qual foi denominado de *Evolução Diferencial Híbrida Auto Adaptativa*, ou em inglês, *Hybrid Self Adaptive Differential Evolution*. O desenvolvimento das duas abordagens apresentadas neste trabalho teve como objetivo a implementação de um algoritmo capaz de ter um bom equilíbrio entre o *exploration* e *exploitation*, o que é importante para aumentar a precisão de um algoritmo meta-heurístico populacional, e que também fosse capaz de adaptar os seus valores de parâmetros de controle durante o processo de otimização, o que é crucial para aumentar a velocidade de convergência e reduzir o risco de estagnação da população durante o processo de otimização. Para avaliar o desempenho e eficiência dos algoritmos propostos, as duas versões, juntamente com alguns algoritmos estado da arte referentes a variantes do algoritmo de evolução diferencial, foram aplicados a três estudos de casos formado por conjuntos de funções propostos para o *IEEE (Institute of Electrical and Electronics Engineers) Congress on Evolutionary Computation* de 2005, 2011 e 2013 e seus resultados foram comparados. Para analisar os resultados obtidos em cada um dos estudos de caso, e, assim, analisar o desempenho dos algoritmos propostos, foram utilizados os testes não paramétricos de *Mann-Whitney* e o teste *post hoc* de *Friedman*. Os resultados experimentais obtidos mostram que, analisando-se de uma maneira global, as duas versões do algoritmo de *Evolução Diferencial Híbrida Auto Adaptativa* apresentam um excelente desempenho, demonstrando, deste modo, sua eficácia e eficiência. Uma vez avaliados seus desempenhos, as duas abordagens foram aplicadas na realização da calibração automática de um modelo responsável por calcular a pressão de combustível no *rail* em motores diesel com sistema *common rail* e na realização da calibração automática de um modelo responsável por calcular a pressão do ar no coletor de admissão em motores a gasolina. Os resultados obtidos demonstram a factibilidade e a eficiência da utilização dos algoritmos de otimização propostos nesta dissertação na realização de calibração automática.

Palavras-chave: Evolução diferencial. Otimização mono-objetivo. Metaheurísticas. Calibração de motores automotivos. Calibração automática.

## ABSTRACT

The growing demand for improving fuel economy and performance, tighter emissions standards, strict vehicle safety regulations and higher requests for driving comfort have led to an exponential increase of the automotive *software* complexity over the past few years, which resulted in an increase in the number of control parameters to be calibrated. As a consequence, the calibration process will require a higher effort and a significant amount of time will be spent to achieve an optimized calibration to meet all the project requirements. In this sense, optimization algorithms based on differential evolution are an useful tool to perform automatic calibration of the control parameters due to its ability to find appropriate solutions to complex optimization problems. Therefore, the main objective of this thesis is the development of a new version of an algorithm based on differential evolution for solving bound-constrained single-objective optimization problems and the application of this algorithm to perform automatic calibration of some *software* functions used by the automotive engine management systems. In this thesis it is presented two version of an algorithm, so-called *Hybrid Self Adaptive Differential Evolution*, which was developed based on the differential evolution ideas. The aim of the development of these two approaches presented on this thesis was to create an algorithm with a good balance between exploration and exploitation, which is important to increase the accuracy and efficiency of a population-based stochastic algorithm, and which is able to adapt its control parameters values over the optimization process, which is also crucial to increase the speed convergence and reduce the risk of search stagnation. To verify the performance of the proposed approaches, the two versions, along with some state-of-arts differential evolution algorithms, were tested over three case studies, composed of sets of benchmark functions proposed for the IEEE (*Institute of Electrical and Electronics Engineers*) Congress on Evolutionary Computation of 2005, 2011 and 2013, and their results were compared. To evaluate the results obtained in each of the case study, and thus analyze the performance of the proposed algorithms, the non-parametric tests of Mann-Whitney and Friedman were used. An overall analysis of experimental results shows that the two versions of the Hybrid Self Adaptive Differential Evolution algorithm have an exceptional performance, which demonstrates the effectiveness and efficiency of the proposed algorithms. After evaluating their performance, both approaches have been employed to perform, firstly, an automatic calibration of a model responsible for estimation of the fuel rail pressure on a common rail diesel engine and then to perform an automatic calibration of a model responsible for estimation of the air pressure in the intake manifold on a gasoline engine. The results demonstrate the feasibility and efficiency of the proposed algorithms to perform automatic calibration.

Key words: Differential evolution. Single-objective optimization. Metaheuristics. Automotive engine calibration. Automatic calibration

## LISTA DE ILUSTRAÇÕES

FIGURA 1: Exemplo bidimensional de geração de um vetor doador.....	36
FIGURA 2: Pseudocódigo algoritmo HSADE-v1 .....	56
FIGURA 3: Exemplo bidimensional de geração de um vetor quase oposto .....	58
FIGURA 4: Pseudocódigo Inicialização_Quase_Oposicional.....	59
FIGURA 5: Pseudocódigo Abordagem_ED.....	66
FIGURA 6: Pseudocódigo Busca_Exploitation.....	68
FIGURA 7: Pseudocódigo Abordagem_Quase_Oposicional.....	70
FIGURA 8: Pseudocódigo Reinicialização_População.....	72
FIGURA 9: Pseudocódigo algoritmo HSADE-v2 .....	73
FIGURA 10: Exemplo de cálculo da média generalizada em função do parâmetro $\alpha$ para o conjunto $B = \{1,2,3,4,5,6,7,8,9,10\}$ .....	78
FIGURA 11: Pseudocódigo Abordagem_ED_v2.....	80
FIGURA 12: Pseudocódigo Busca_Exploitation_v2 .....	83
FIGURA 13: Pseudocódigo Reinicialização_População_v2.....	83
FIGURA 14: Evolução das probabilidades de cada uma das estratégias de geração de vetores experimentais para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	170
FIGURA 15: Evolução dos parâmetros $\mu_{F,m}$ e $\mu_{CR,m}$ , $m = 1,2$ , para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	171
FIGURA 16: Evolução do fator de ajuste $\omega$ para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	172
FIGURA 17: Evolução dos parâmetros $\mu_F$ e $\mu_{CR}$ para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	176
FIGURA 18: Evolução da probabilidade $\beta$ para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	177
FIGURA 19: Evolução do fator de ajuste $\omega$ para quatro funções do CEC2005 (a) $F_4$ , (b) $F_6$ , (c) $F_{14}$ , (d) $F_{16}$ para o caso $D = 30$ .....	178
FIGURA 20: Exemplo de um sistema de controle de motor a gasolina aspirado com injeção indireta.....	179
FIGURA 21: Metodologia para a realização da calibração automática.....	182

FIGURA 22: Quadro esquemático do processo de calibração automática.....	183
FIGURA 23: Otimização: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	188
FIGURA 24: Detalhe do resultado processo de otimização entre 500 e 750 segundos.....	188
FIGURA 25: Detalhe do resultado processo de otimização entre 750 e 1000 segundos....	188
FIGURA 26: Detalhe do resultado processo de otimização entre 1250 e 1450 segundos..	189
FIGURA 27: Detalhe do resultado processo de otimização entre 1550 e 1850 segundos..	189
FIGURA 28: Validação: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	190
FIGURA 29: Detalhe do resultado da validação entre 400 e 650 segundos.....	191
FIGURA 30: Detalhe do resultado da validação entre 750 e 950 segundos.....	191
FIGURA 31: Detalhe do resultado da validação entre 1250 e 1500 segundos.....	191
FIGURA 32: Otimização: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	193
FIGURA 33: Detalhe do resultado processo de otimização entre 500 e 750 segundos.....	193
FIGURA 34: Detalhe do resultado processo de otimização entre 750 e 1000 segundos....	193
FIGURA 35: Detalhe do resultado processo de otimização entre 1250 e 1450 segundos..	194
FIGURA 36: Detalhe do resultado processo de otimização entre 1550 e 1850 segundos..	194
FIGURA 37: Validação: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	195
FIGURA 38: Detalhe do resultado da validação entre 400 e 650 segundos.....	196
FIGURA 39: Detalhe do resultado da validação entre 750 e 950 segundos.....	196
FIGURA 40: Detalhe do resultado da validação entre 1250 e 1500 segundos.....	196
FIGURA 41: Otimização: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	200
FIGURA 42: Detalhe do resultado processo de otimização entre 30 e 90 segundos .....	200
FIGURA 43: Detalhe do resultado processo de otimização entre 150 e 250 segundos.....	201
FIGURA 44: Detalhe do resultado processo de otimização entre 300 e 400 segundos.....	201
FIGURA 45: Detalhe do resultado processo de otimização entre 800 e 950 segundos.....	201
FIGURA 46: Detalhe do resultado processo de otimização entre 1000 e 1100 segundos..	202
FIGURA 47: Validação: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	203
FIGURA 48: Detalhe do resultado da validação entre 100 e 250 segundos.....	203



FIGURA 49: Detalhe do resultado da validação entre 350 e 500 segundos.....	204
FIGURA 50: Detalhe do resultado da validação entre 540 e 640 segundos.....	204
FIGURA 51: Detalhe do resultado da validação entre 640 e 710 segundos.....	204
FIGURA 52: Otimização: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	206
FIGURA 53: Detalhe do resultado processo de otimização entre 30 e 90 segundos .....	206
FIGURA 54: Detalhe do resultado processo de otimização entre 150 e 250 segundos.....	206
FIGURA 55: Detalhe do resultado processo de otimização entre 300 e 400 segundos.....	207
FIGURA 56: Detalhe do resultado processo de otimização entre 800 e 950 segundos.....	207
FIGURA 57: Detalhe do resultado processo de otimização entre 1000 e 1100 segundos..	207
FIGURA 58: Validação: comparação entre o valor real e o valor estimado para a pressão do <i>rail</i> .....	209
FIGURA 59: Detalhe do resultado da validação entre 100 e 250 segundos.....	209
FIGURA 60: Detalhe do resultado da validação entre 350 e 500 segundos.....	209
FIGURA 61: Detalhe do resultado da validação entre 540 e 640 segundos.....	210
FIGURA 62: Detalhe do resultado da validação entre 640 e 710 segundos.....	210
FIGURA 63: Evolução do EMQ para o modelo de pressão do <i>rail</i> durante o processo de otimização .....	214
FIGURA 64: Evolução do EMQ para o modelo da pressão no coletor durante o processo de otimização .....	215
FIGURA 65: Comparação do valor estimado da pressão do <i>rail</i> utilizando-se a calibração gerada pelo algoritmo HSADE-v1 e pelo algoritmo JADE <i>com arquivo</i> .....	216
FIGURA 66: Comparação do valor estimado da pressão do <i>rail</i> utilizando-se a calibração gerada pelo algoritmo HSADE-v1 e pelo algoritmo JADE <i>com arquivo</i> .....	218
FIGURA 67: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 10$ .....	230
FIGURA 68: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 10$ .....	231
FIGURA 69: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 10$ .....	232

FIGURA 70: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ , (g) $F_{25}$ para o caso $D = 10$ .	233
FIGURA 71: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 30$ .	234
FIGURA 72: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 30$ .	235
FIGURA 73: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 30$ .	236
FIGURA 74: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ , (g) $F_{25}$ para o caso $D = 30$ .	237
FIGURA 75: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 50$ .	238
FIGURA 76: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 50$ .	239
FIGURA 77: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 50$ .	240
FIGURA 78: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ , (g) $F_{25}$ para o caso $D = 50$ .	241
FIGURA 79: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, <i>EPpSDE</i> , DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a) $P_1$ , (b) $P_2$ , (c) $P_3$ , (d) $P_4$ , (e) $P_5$ , (f) $P_6$ .	243
FIGURA 80: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a) $P_7$ , (b) $P_8$ , (c) $P_9$ , (d) $P_{10}$ , (e) $P_{11}$ , (f) $P_{12}$ .	244

FIGURA 81: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a) $P_{13}$ , (b) $P_{14}$ , (c) $P_{15}$ , (d) $P_{16}$ , (e) $P_{17}$ , (f) $P_{18}$ .	245
FIGURA 82: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a) $P_{19}$ , (b) $P_{20}$ , (c) $P_{21}$ , (d) $P_{22}$ .	246
FIGURA 83: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 10$ .	248
FIGURA 84: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 10$ .	249
FIGURA 85: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 10$ .	250
FIGURA 86: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ para o caso $D = 10$ .	251
FIGURA 87: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{25}$ , (b) $F_{26}$ , (c) $F_{27}$ , (d) $F_{28}$ para o caso $D = 10$ .	252
FIGURA 88: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 30$ .	253
FIGURA 89: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 30$ .	254
FIGURA 90: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 30$ .	255
FIGURA 91: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ para o caso $D = 30$ .	256

FIGURA 92: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{25}$ , (b) $F_{26}$ , (c) $F_{27}$ , (d) $F_{28}$ para o caso $D = 30$ .....	257
FIGURA 93: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a) $F_1$ , (b) $F_2$ , (c) $F_3$ , (d) $F_4$ , (e) $F_5$ , (f) $F_6$ para o caso $D = 50$ .....	258
FIGURA 94: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_7$ , (b) $F_8$ , (c) $F_9$ , (d) $F_{10}$ , (e) $F_{11}$ , (f) $F_{12}$ para o caso $D = 50$ .....	259
FIGURA 95: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{13}$ , (b) $F_{14}$ , (c) $F_{15}$ , (d) $F_{16}$ , (e) $F_{17}$ , (f) $F_{18}$ para o caso $D = 50$ .....	260
FIGURA 96: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{19}$ , (b) $F_{20}$ , (c) $F_{21}$ , (d) $F_{22}$ , (e) $F_{23}$ , (f) $F_{24}$ para o caso $D = 50$ .....	261
FIGURA 97: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a) $F_{25}$ , (b) $F_{26}$ , (c) $F_{27}$ , (d) $F_{28}$ para o caso $D = 50$ .....	262

## LISTA DE TABELAS

TABELA 1 – RESUMO COMPARATIVO ENTRE AS DUAS VERSÕES DO ALGORITMO HSADE .....	84
TABELA 2 – VALORES DOS PARÂMETROS DO ALGORITMO HSADE-V1 .....	95
TABELA 3 – VALORES DOS PARÂMETROS DO ALGORITMO HSADE-V2.....	96
TABELA 4 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2005 .....	101
TABELA 5 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	102
TABELA 6 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	102
TABELA 7 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	103
TABELA 8 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2005.....	106
TABELA 9 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	108
TABELA 10 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	108
TABELA 11 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	108
TABELA 12 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005.....	109

TABELA 13 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	111
TABELA 14 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	112
TABELA 15 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005.....	113
TABELA 16 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	115
TABELA 17 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	116
TABELA 18 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005.....	117
TABELA 19 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	119
TABELA 20 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	120
TABELA 21 – DIMENSÃO DOS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011.....	121
TABELA 22 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011.....	124
TABELA 23 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	125
TABELA 24 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011.....	126

TABELA 25 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	127
TABELA 26 – RESULTADOS OBTIDOS PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 .....	128
TABELA 27 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	130
TABELA 28 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	131
TABELA 29 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2013 .....	136
TABELA 30 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	137
TABELA 31 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	137
TABELA 32 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	138
TABELA 33 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE <i>FRIEDAN</i> PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2013 .....	141
TABELA 34 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	142
TABELA 35 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	142
TABELA 36 – RESULTADO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	143

TABELA 37 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013.....	144
TABELA 38 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	147
TABELA 39 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	148
TABELA 40 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013.....	149
TABELA 41 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	152
TABELA 42 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	153
TABELA 43 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013.....	154
TABELA 44 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE.....	157
TABELA 45 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE.....	158
TABELA 46 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005.....	163
TABELA 47 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005.....	163
TABELA 48 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005.....	163
TABELA 49 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA OS PROBLEMAS DO CEC2011.....	164
TABELA 50 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013.....	164



TABELA 51 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 .....	164
TABELA 52 – RESULTADO COMPARATIVO DO TESTE DE <i>MANN-WHITNEY</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 .....	165
TABELA 53 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 .....	165
TABELA 54 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 .....	165
TABELA 55 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 .....	166
TABELA 56 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA OS PROBLEMAS DO CEC2011 .....	166
TABELA 57 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 .....	166
TABELA 58 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 .....	167
TABELA 59 – RESULTADO COMPARATIVO DO TESTE <i>POST HOC</i> DE <i>FRIEDMAN</i> PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 .....	167
TABELA 60 – MÉDIA E DESVIO PADRÃO DO VALOR DO EMQ OBTIDOS PARA O MODELO DE CÁLCULO DA PRESSÃO DO RAIL .....	212
TABELA 61 – MÉDIA E DESVIO PADRÃO DO VALOR DO EMQ OBTIDOS PARA O MODELO DE CÁLCULO DA PRESSÃO NO COLETOR DE ADMISSÃO .....	212
TABELA 62 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE .....	213
TABELA 63 – RESULTADOS DO TESTE DE <i>MANN-WHITNEY</i> CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE .....	213

## LISTA DE SIGLAS

<b>AHCXLS</b>	- <i>Adaptive Hill-Climbing Crossover-based Local Search</i>
<b>CEC</b>	- <i>Congress on Evolutionary Computation</i>
<b>CEC2005</b>	- <i>Congress on Evolutionary Computation de 2005</i>
<b>CEC2011</b>	- <i>Congress on Evolutionary Computation de 2011</i>
<b>CEC2013</b>	- <i>Congress on Evolutionary Computation de 2013</i>
<b>CoDE</b>	- <i>Composite Differential Evolution</i>
<b>DEahcSPX</b>	- <i>Differential Evolution using Adaptive Hill-Climbing crossover-based local search and Simplex Crossover</i>
<b>DEGL</b>	- <i>Differential Evolution with Global and Local neighborhoods</i>
<b>ECU</b>	- <i>Engine Control Unit.</i>
<b>ED</b>	- <i>Evolução Diferencial</i>
<b>EMQ</b>	- <i>Erro Médio Quadrático</i>
<b>EPSDE</b>	- <i>Ensemble of Parameters and mutation Strategies in Differential Evolution</i>
<b>FWER</b>	- <i>Family-wise error rate</i>
<b>HSADe</b>	- <i>Hybrid Self Adaptive Differential Evolution</i>
<b>IEEE</b>	- <i>Institute of Electrical and Electronics Engineers</i>
<b>JADE</b>	- <i>J Adaptive Differential Evolution</i>
<b>jDE</b>	- <i>j Differential Evolution</i>
<b>SaDE</b>	- <i>Self-adaptive Differential Evolution</i>
<b>VPA</b>	- <i>Valor-p Ajustado</i>

## LISTA DE SIMBOLOS

$\vec{X}$	Vetor pertencente a $\mathbb{R}^D$
$X_j$	$j$ -ésima componente do vetor $\vec{X}$
$D$	Número de parâmetros do problema a ser otimizado
$\vec{X}^{max}$	Limite superior das variáveis do problema a ser resolvido
$\vec{X}^{min}$	Limite inferior das variáveis do problema a ser resolvido
$X_j^{max}$	$j$ -ésima componente do vetor $\vec{X}^{max}$
$X_j^{min}$	$j$ -ésima componente do vetor $\vec{X}^{min}$
$S$	Espaço de busca do problema de otimização
$\vec{X}^*$	Mínimo global do problema de otimização
$\vec{X}_l$	Mínimo local do problema de otimização
$NP$	Tamanho da população
$G$	Geração atual do processo de otimização
$Gmax$	Número máximo de gerações do processo de otimização
$\vec{X}_{i,G}$	$i$ -ésimo indivíduo da população na geração $G$
$X_{j,i,G}$	$j$ -ésima componente do indivíduo $\vec{X}_{i,G}$
$\vec{X}_{i,0}$	Indivíduo pertencente à população inicial
$rand_{j,i}[0,1]$	Gerador de números aleatórios com distribuição uniforme no intervalo $[0,1]$ avaliado independentemente para cada $j$ -ésima componente do $i$ -ésimo indivíduo
$\vec{V}_{i,G}$	Vetor doador relativo ao $i$ -ésimo vetor alvo da geração $G$
$F$	Fator de mutação
$V_{j,i,G}$	$j$ -ésima componente do vetor doador $\vec{V}_{i,G}$
$\vec{U}_{i,G}$	Vetor experimental relativo ao $i$ -ésimo vetor alvo da geração $G$
$CR$	Taxa de cruzamento
$U_{j,i,G}$	$j$ -ésima componente do vetor experimental $\vec{U}_{i,G}$
$\vec{X}_{best,G}$	Indivíduo da população na geração $G$ que apresenta o menor valor de função objetivo
$K$	Fator de recombinação

$rand[0,1]$	Gerador de números aleatórios com distribuição uniforme no intervalo $[0,1]$
$F_i$	Fator de mutação associado ao $i$ -ésimo indivíduo da população atual
$CR_i$	Taxa de cruzamento associado ao $i$ -ésimo indivíduo da população atual
$\lambda$	Número de melhores indivíduos da população atual utilizados na formação do vetor base no operador de mutação <i>DE/target-to-pbest</i>
$\vec{X}_{best,G}^p$	Indivíduo escolhido aleatoriamente e uniformemente a partir do conjunto formado pelos $\lambda$ melhores indivíduos da população atual.
$p$	Parâmetro que define a “ganância” ( <i>greediness</i> ) do operador de mutação <i>DE/target-to-pbest</i>
$A$	Arquivo externo formado pelo conjunto de soluções inferiores recentemente exploradas e armazenadas utilizadas pela estratégia <i>DE/target-to-pbest/1</i> com arquivo
$\varphi$	Tamanho do arquivo externo $A$
$P$	População corrente
$\mu_F$	Parâmetro de locação da distribuição de Cauchy utilizada para gerar o fator de mutação
$\mu_{CR}$	Média da distribuição normal utilizada para gerar a taxa de cruzamento
$c$	Parâmetro que define a taxa de adaptação dos parâmetros $\mu_F$ e $\mu_{CR}$
$S_F$	Conjunto de todos os fatores de mutação bem sucedidos na geração $G$
$S_{CR}$	Conjunto de todas as taxas de cruzamento bem sucedidas na geração $G$
$\tau_1$	Probabilidade de alteração dos parâmetros $F_i$
$\tau_2$	Probabilidade de alteração dos parâmetros $CR_i$
$F_l$	Limite inferior do fator de mutação
$F_u$	Limite superior do fator de mutação
$p_{k,G}$	Probabilidade da $k$ -ésima estratégia ser escolhida na geração $G$
$LP$	Número de gerações necessárias para atualizar as probabilidades
	$p_{k,G}$

$ns_{k,g}$	Número de vetores experimentais bem sucedidos nas últimas $LP$ gerações gerados pela $k$ -ésima estratégia
$nf_{k,g}$	Número de vetores experimentais gerados pela $k$ -ésima estratégia e descartados durante o processo de seleção nas últimas $LP$ gerações
$CRm_k$	Média da distribuição normal associada a $k$ -ésima estratégia utilizada para gerar a taxa de cruzamento
$\vec{L}_{i,G}$	O vetor doador local relativo ao $i$ -ésimo vetor alvo da geração $G$
$\vec{X}_{n\_best,i,G}$	Indivíduo com menor valor de função objetivo dentre aqueles localizados na vizinhança do vetor alvo $\vec{X}_{i,G}$
$\vec{g}_{i,G}$	Vetor doador global relativo ao $i$ -ésimo vetor alvo da geração $G$
$\vec{X}_{g\_best,G}$	Indivíduo com menor valor de função objetivo dentre todos os indivíduos da população na geração $G$
$\eta$	Fator de escala utilizado para a geração do vetor $\vec{L}_{i,G}$ e $\vec{g}_{i,G}$
$\psi$	Fator de escala utilizado para a geração do vetor $\vec{L}_{i,G}$ e $\vec{g}_{i,G}$
$\theta$	Parâmetro responsável por controlar o balanço entre a <i>exploration</i> e <i>exploitation</i> do algoritmo DEGL
$\check{x}$	Oposto do número real $x$
$\check{x}^q$	Quase oposto do número real $x$
$\check{X}^q$	Vetor quase oposto de $\vec{X}$
$\check{X}$	Vetor oposto de $\vec{X}$
$\check{x}_j$	Oposto da $j$ -ésima componente de $\vec{X}$
$\check{x}_j^q$	Quase oposto da $j$ -ésima componente de $\vec{X}$
$rand_j[0,1]$	Gerador de números aleatórios com distribuição uniforme no intervalo $[0,1]$ avaliado independentemente para cada $j$ -ésima componente de $\vec{X}$ .
$\beta$	Probabilidade da busca voltada à <i>exploitation</i>
$\Omega$	Número de gerações para atualização das probabilidades das estratégias de geração dos vetores experimentais
$\omega$	Fator de ajuste com o objetivo de ajustar os movimentos dos vetores experimentais

$p_m$	Probabilidade da $m$ -ésima estratégia a ser escolhida
$ns_m$	Número de vetores experimentais bem sucedidos nas últimas $\Omega$ gerações gerados pela $m$ -ésima estratégia
$ns$	Número total de vetores experimentais bem sucedidos nas últimas $\Omega$ gerações
$\mu_{F,m}$	Parâmetro de locação da distribuição de Cauchy utilizada para gerar o fator de mutação associado a $m$ -ésima estratégia
$S_{F,m}$	Conjunto de todos os fatores de mutação bem sucedidos associados à estratégia $m$ na geração $G$
$\mu_{CR,m}$	Média da distribuição normal utilizada para gerar a taxa de cruzamento associada a $m$ -ésima estratégia
$S_{CR,m}$	Conjunto de todas as taxas de cruzamento bem sucedidas associadas à estratégia $m$ na geração $G$
$MIN_j^G$	Valor mínimo da $j$ -ésima componente da população atual
$MAX_j^G$	Valor máximo da $j$ -ésima componente da população atual
$\sigma_j$	Desvio padrão da distribuição uniforme utilizada para reinicializar a população em caso de estagnação
$\vartheta$	Tamanho do arquivo de armazenagem do histórico da média dos parâmetros de controle $F$ e $CR$
$H_F$	Arquivo responsável por armazenar o histórico da média do parâmetro de controle $F$
$H_{CR}$	Arquivo responsável por armazenar o histórico da média do parâmetro de controle $CR$
$m_{SF}$	Média dos parâmetros de controle $F$ bem sucedidos de cada geração
$m_{SCR}$	Média dos parâmetros de controle $CR$ bem sucedidos de cada geração
$\vec{X}_{Melhor}$	Melhor solução encontrada pelo algoritmo de otimização em um experimento

## SUMÁRIO

1	INTRODUÇÃO .....	26
1.1	OBJETIVOS .....	28
1.1.1	Objetivo geral .....	28
1.1.2	Objetivos específicos.....	28
1.2	ESTRUTURA DA DISSERTAÇÃO .....	29
2	FORMULAÇÃO DO PROBLEMA DE OTIMIZAÇÃO .....	31
3	EVOLUÇÃO DIFERENCIAL.....	33
3.1	ALGORITMO CLÁSSICO ED .....	34
3.1.1	Geração da população inicial.....	35
3.1.2	Mutação.....	35
3.1.3	Cruzamento.....	36
3.1.4	Seleção .....	37
3.2	VARIAÇÕES DO ALGORITMO ED .....	38
3.3	ESCOLHA DOS PARÂMETROS DE CONTROLE.....	40
3.3.1	Tamanho da população .....	41
3.3.2	Fator de mutação .....	41
3.3.3	Taxa de cruzamento .....	42
3.4	TRATAMENTO DAS RESTRIÇÕES.....	42
4	ESTADO DA ARTE.....	44
4.1	JADE .....	44
4.2	jDE .....	46
4.3	CoDE.....	47
4.4	EPSDE .....	48
4.5	SaDE .....	49
4.6	DEGL.....	51
4.7	DEahcSPX .....	53
5	ABORDAGEM DE EVOLUÇÃO DIFERENCIAL HÍBRIDA AUTO ADAPTATIVA.....	54
5.1	PRIMEIRA VERSÃO DO ALGORITMO HSADE (HSADE-v1) .....	55
5.1.1	Inicialização da população.....	56
5.1.2	Abordagem de Evolução Diferencial.....	60
5.1.2.1	Adaptação da estratégia de geração do vetor experimental .....	60

5.1.2.2	Adaptação dos parâmetros de controle $F$ e $CR$ .....	62
5.1.3	Busca voltada à <i>Exploitation</i> .....	66
5.1.4	Abordagem quase oposicional.....	68
5.1.5	Reinicialização da população .....	71
5.2	SEGUNDA VERSÃO DO ALGORITMO HSADE (HSADE-v2) .....	72
5.2.1	Abordagem de Evolução Diferencial.....	74
5.2.1.1	Estratégia de geração do vetor experimental .....	74
5.2.1.2	Adaptação dos parâmetros de controle $F$ e $CR$ .....	75
5.2.2	Busca voltada à <i>Exploitation</i> .....	80
5.2.3	Reinicialização da população .....	83
5.3	RESUMO COMPARATIVO.....	83
6	AVALIAÇÃO DO DESEMPENHO DO ALGORITMO PROPOSTO .....	87
6.1	GERAÇÃO DOS DADOS PARA ANÁLISE.....	88
6.2	CRITÉRIO DE DESEMPENHO .....	89
6.2.1	Teste de <i>Mann-Whitney</i> .....	90
6.2.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	92
6.3	ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2005.....	94
6.3.1	Definição dos parâmetros.....	95
6.3.2	Resultados experimentais e discussão.....	97
6.3.2.1	Análise do desempenho do algoritmo HSADE-v1.....	97
6.3.2.1.1	Teste de <i>Mann-Whitney</i> .....	97
6.3.2.1.2	Teste de <i>Post hoc</i> de <i>Friedman</i> .....	100
6.3.2.2	Análise do desempenho do algoritmo HSADE-v2.....	103
6.3.2.2.1	Teste de <i>Mann-Whitney</i> .....	103
6.3.2.2.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	106
6.4	ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2011.....	121
6.4.1	Definição dos parâmetros.....	121
6.4.2	Resultados experimentais e discussão.....	122
6.4.2.1	Análise do desempenho do algoritmo HSADE-v1.....	123
6.4.2.1.1	Teste de <i>Mann-Whitney</i> .....	123
6.4.2.1.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	123
6.4.2.2	Análise do desempenho do algoritmo HSADE-v2.....	125
6.4.2.2.1	Teste de <i>Mann-Whitney</i> .....	125
6.4.2.2.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	126
6.5	ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2013.....	132



6.5.1	Definição dos parâmetros .....	132
6.5.2	Resultados experimentais e discussão .....	133
6.5.2.1	Análise do desempenho do algoritmo HSADE-v1 .....	133
6.5.2.1.1	Teste de <i>Mann-Whitney</i> .....	133
6.5.2.1.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	135
6.5.2.2	Análise do desempenho do algoritmo HSADE-v2 .....	138
6.5.2.2.1	Teste de <i>Mann-Whitney</i> .....	138
6.5.2.2.2	Teste <i>post hoc</i> de <i>Friedman</i> .....	140
6.6	ANÁLISE GERAL DOS RESULTADOS EXPERIMENTAIS .....	159
6.7	ANÁLISE SOBRE A ADAPTAÇÃO DOS PARÂMETROS DOS ALGORITMOS PROPOSTOS .....	167
6.7.1	Algoritmo HSADE-v1 .....	167
6.7.2	Algoritmo HSADE-v2 .....	173
7	CALIBRAÇÃO DE MOTORES AUTOMOTIVOS .....	179
7.1	APLICAÇÃO DO ALGORITMO HSADE NA REALIZAÇÃO DA CALIBRAÇÃO AUTOMÁTICA DE MOTORES AUTOMOTIVOS .....	181
7.1.1	Metodologia .....	182
7.1.2	Estudo de caso: calibração do modelo de cálculo da pressão de combustível no <i>rail</i> .....	184
7.1.2.1	Aquisição dos dados .....	186
7.1.2.2	Resultados experimentais e discussão .....	186
7.1.2.2.1	Análise do algoritmo HSADE-v1 .....	187
7.1.2.2.2	Análise do algoritmo HSADE-v2 .....	192
7.1.3	Estudo de caso: calibração do modelo de cálculo da pressão do ar no coletor de admissão .....	197
7.1.3.1	Aquisição dos dados .....	198
7.1.3.2	Resultados experimentais e discussão .....	198
7.1.3.2.1	Análise do algoritmo HSADE-v1 .....	199
7.1.3.2.2	Análise do algoritmo HSADE-v2 .....	205
7.1.4	Análise de desempenho .....	210
7.1.4.1	Resultados experimentais e discussão .....	211
7.1.5	Análise geral dos algoritmos propostos .....	218
8	CONCLUSÕES E TRABALHOS FUTUROS .....	221
	REFERÊNCIAS .....	224
	APÊNDICES .....	229

## 1 INTRODUÇÃO

Custo, qualidade e tempo são os principais fatores do desenvolvimento de qualquer projeto de engenharia. Entretanto ao longo dos últimos anos, cada vez mais é necessário o desenvolvimento de projetos de engenharia visando a obtenção de uma maior eficiência com o menor custo operacional, ou seja, desenvolver produtos a curto prazo, com custos cada vez mais reduzidos, mas sempre com a qualidade necessária (LIBERATORE, 2013; ANUAR, 2011).

Neste sentido a otimização vem desempenhando um papel fundamental em diversas áreas da engenharia permitindo, assim, obter produtos através do melhor compromisso em termos de custo, qualidade e tempo.

Diversos métodos de otimização foram desenvolvidos nestas últimas décadas. Dentre estes diversos métodos destacam-se os algoritmos evolutivos que buscam inspiração na biologia evolutiva. Os algoritmos evolutivos são formados por uma ampla família de algoritmos de otimização estocástica, dentre os quais podem ser citados os algoritmos genéticos, a programação evolutiva, a programação genética e a estratégia evolutiva (BINITHA; SATHYA, 2012). Além destes, outro algoritmo evolutivo, denominado Evolução Diferencial (ED), vem obtendo destaque no cenário científico nos últimos anos devido a sua facilidade de implementação e eficiência na busca de soluções adequadas para problemas complexos de otimização (geralmente problemas representados por funções não lineares, multimodais, não diferenciáveis ou descontínuas) (DAS; SUGANTHAN, 2011; SUGANTHAN, 2012).

O algoritmo de evolução diferencial foi apresentado pela primeira vez por Storn e Price (1995) e desde então esta técnica vem sendo largamente estudada e evoluída. O algoritmo de evolução diferencial é uma metaheurística baseada em estratégias evolutivas populacionais que utiliza operadores genéticos de mutação, cruzamento e seleção para evoluir a população de soluções potenciais do problema a ser resolvido em direção ao ótimo global.

Devido à sua simplicidade e eficácia, a evolução diferencial tornou-se uma técnica muito popular e vem sendo amplamente utilizada em diversas áreas da engenharia das quais podem ser citadas, como exemplo, a aplicação no campo da

aerodinâmica (ROGALSKY; DERKSEN; KOCABIYIK, 1999), comunicação de satélites (SALMAN et al., 2010), sistemas elétricos de potência (CHELLIAHA et al., 2014), projeto de sistemas mecânicos (SARUHAN, 2014), sistemas nucleares (MAIO; BARONCHELLI; ZIO, 2014), sistemas hidrotérmicos (BASU, 2014) e engenharia química (VAZQUEZ-OJEDA et al., 2013).

Outra área da engenharia na qual a otimização tem um potencial de aplicação é a engenharia automotiva.

A criação de normas de emissões de gases veiculares cada vez mais restritivas, a necessidade de redução de consumo de combustíveis, requisitos rigorosos de segurança e a busca para atender as demandas de clientes cada vez mais exigentes, fizeram com que a complexidade dos *softwares* automotivos crescesse exponencialmente ao longo dos últimos anos (JIANG; NUTTER; GULLITTI, 2012; SAMPATH; HEGDE, 2013; FRANCOIS, 2013; MENON; VENUGOPAL, 2014). Ao mesmo tempo, a necessidade de entregar produtos com maior qualidade, mais confiáveis, competitivos e no menor prazo possível faz com que objetivos de custo, qualidade e rapidez sejam importantes no desenvolvimento de projetos de *softwares* automotivos.

Uma das áreas da engenharia automotiva mais afetada por estes desafios é a área responsável pela calibração dos parâmetros dos *softwares* utilizados para o controle e funcionamento dos motores automotivos.

A calibração é uma fase crucial no desenvolvimento de *software* do motor e pode ser definida como o processo de ajustar adequadamente os parâmetros de funções implementadas em *softwares*, utilizados para o controle e funcionamento dos motores automotivos, a fim de atender aos requisitos do projeto em relação ao desempenho, emissões, consumo e dirigibilidade (KAJI, 2008; LANGOUËT et al., 2008).

Em geral, os métodos de calibração convencionais são baseados em testes em dinamômetro de rolos e em veículo e são normalmente realizados manualmente. Entretanto, com o aumento da complexidade dos *softwares*, houve um aumento significativo no número de parâmetros a serem calibrados, fazendo com que o processo de calibração demandasse mais esforço e tempo para que os motores automotivos pudessem atender aos requisitos do projeto em termos de consumo, desempenho, emissões e dirigibilidade.

Sendo assim, para que os objetivos de custo, qualidade e rapidez sejam alcançados, torna-se cada vez mais necessária a utilização de ferramentas computacionais no processo de calibração dos parâmetros de *softwares* de motores automotivos.

Neste sentido, portanto, algoritmos de otimização baseados na evolução diferencial são uma ferramenta com potencial para ser aplicada na realização de calibração automática dos parâmetros dos *softwares* de motores automotivos.

## 1.1 OBJETIVOS

### 1.1.1 Objetivo geral

Desenvolver uma nova versão de um algoritmo baseado na evolução diferencial para solução de problemas mono-objetivo sujeito apenas a restrições laterais, aplicá-lo na resolução de problemas testes propostos no *Congress on Evolutionary Computation* (CEC) e também aplicá-lo a um problema real relacionado à engenharia automotiva relativo à realização de calibração automática de algumas funções de *softwares* utilizados para o controle e funcionamento de motores automotivos.

### 1.1.2 Objetivos específicos

Os objetivos específicos relativos a este trabalho são:

- Estudar o método de evolução diferencial;
- Realizar um levantamento e estudo das variações baseadas no algoritmo de evolução diferencial consideradas como estado da arte;
- Desenvolver e implementar um algoritmo baseado na evolução diferencial para solução de problemas mono-objetivo sujeito apenas a restrições laterais;
- Aplicar o algoritmo proposto, juntamente com alguns outros algoritmos baseados na evolução diferencial, na resolução de conjuntos de problemas propostos para a competição do *Institute of*

*Electrical and Electronics Engineers (IEEE) Congress on Evolutionary Computation (CEC)* em 2005, 2011 e 2013;

- Avaliar o desempenho e eficiência do algoritmo proposto através da avaliação dos resultados obtidos em cada um dos três estudos de caso, utilizando-se para isto o teste não paramétrico de *Mann-Whitney* e o teste *post hoc* de *Friedman*;
- Aplicar o algoritmo proposto para realizar a calibração automática de parâmetros de uma função de um *software* utilizado para controle de motores diesel;
- Aplicar o algoritmo proposto para realizar a calibração automática de parâmetros de uma função de um *software* utilizado para controle de motores gasolina.

## 1.2 ESTRUTURA DA DISSERTAÇÃO

O restante desta dissertação está organizado da seguinte maneira.

No *capítulo 2* é apresentada a formulação matemática do problema de otimização, bem como alguns conceitos e definições relativos a problemas de otimização que serão utilizados ao longo desta dissertação.

O *capítulo 3* destaca os princípios e definições relativos ao algoritmo ED que servirá de base para o desenvolvimento do algoritmo proposto neste trabalho. Neste capítulo são abordadas as etapas do algoritmo clássico ED e algumas de suas variações. Nesta seção também é apresentada uma discussão sobre a escolha dos parâmetros de controle e também as metodologias para tratar as restrições no algoritmo ED às quais estão sujeitas os problemas de otimização.

No *capítulo 4* são descritos brevemente alguns algoritmos de otimização estocásticos baseados na ED e que serão usados para comparação durante a avaliação do desempenho do algoritmo proposto neste trabalho.

No *capítulo 5* é apresentado o algoritmo proposto para resolução de problemas de otimização mono-objetivo sujeitos apenas a restrições laterais denominado *Evolução Diferencial Híbrida Auto Adaptativa* ou em inglês, *Hybrid Self Adaptive Differential Evolution (HSADE)*. Neste trabalho, duas versões para o algoritmo HSADE foram desenvolvidas. Cada uma destas duas versões é descrita em detalhes neste capítulo.

No *capítulo 6* são relatados os estudos de casos utilizados nesta dissertação para avaliar o desempenho e eficiência dos algoritmos propostos, bem como os resultados obtidos.

No *capítulo 7* são abordados os estudos de casos relativo à realização de calibração automática de duas funções de *softwares* utilizados para o controle e funcionamento de motores automotivos.

Este trabalho é concluído no *capítulo 8*.

## 2 FORMULAÇÃO DO PROBLEMA DE OTIMIZAÇÃO

Em um problema de otimização deseja-se obter um vetor  $\vec{X} \in \mathbb{R}^D$  que otimize uma determinada função denominada por *função objetivo*. Este problema pode estar sujeito ou não a restrições de igualdade, desigualdade e restrições laterais. Sendo assim, procura-se encontrar o melhor valor possível para a função objetivo para o qual o valor atribuído às variáveis não viole nenhuma das restrições.

Sem perda de generalidade, um problema de otimização sempre pode ser formulado como um problema de minimização, pois  $\max f(\vec{X}) = -\min(-f(\vec{X}))$  (DAS; SUGANTHAN, 2011). Desta forma, caso deseja-se maximizar uma determinada função  $f(\vec{X})$ , basta encontrar o ponto que minimize a função  $-f(\vec{X})$ . Consequentemente, durante todo este trabalho, todo problema de otimização será considerado como um problema de minimização.

Sendo assim, um problema de minimização pode ser escrito da seguinte forma (BAZARAA; JARVIS; SHERALLI, 1990):

$$\text{Minimizar} \quad f(\vec{X}), \quad f: \mathbb{R}^D \rightarrow \mathbb{R} \quad (1)$$

Sujeito a

$$c_r(\vec{X}) = 0, \quad r = 1, 2, \dots, m \quad (2)$$

$$h_t(\vec{X}) \leq 0, \quad t = m + 1, m + 2, \dots, n \quad (3)$$

$$\vec{X}^{min} \leq \vec{X} \leq \vec{X}^{max} \quad (4)$$

onde  $f(\vec{X})$  representa a função objetivo a ser minimizada,  $c_r(\vec{X})$  e  $h_t(\vec{X})$  representam as restrições de igualdade e desigualdade respectivamente e  $\vec{X}^{min}$ ,  $\vec{X}^{max}$  são vetores  $D$ -dimensionais que representam os limites mínimos e máximos respectivamente impostos aos pontos  $\vec{X} \in \mathbb{R}^D$  sendo denominados como restrições laterais.

As equações de (1) a (4) representam a formulação genérica do problema de otimização. Entretanto, o foco deste trabalho é o desenvolvimento de um algoritmo para solução de problemas de minimização de funções reais sujeito

apenas a restrições laterais. Sendo assim, o problema de otimização pode ser reformulado e expresso da seguinte forma:

$$\min_{\vec{X} \in S} f(\vec{X}), \quad f: S \subseteq \mathbb{R}^D \rightarrow \mathbb{R} \quad (5)$$

onde

$$S = \{\vec{X} \in \mathbb{R}^D \mid X_j^{min} \leq X_j \leq X_j^{max}, j = 1, \dots, D\} \quad (6)$$

O conjunto  $S$  definido em (6) é denominado *espaço de busca*. Para problemas sem restrições  $S = \mathbb{R}^D$ .

Em outras palavras, no processo de otimização busca-se determinar um vetor  $\vec{X}^* \in S$  tal que  $f(\vec{X}^*) \leq f(\vec{X})$  para todo  $\vec{X} \in S$ . O vetor  $\vec{X}^*$  é chamado de *ponto de mínimo global*. O vetor  $\vec{X}^*$  também é chamado de *solução ótima* do problema ou *ótimo global*.

Um dos desafios enfrentados por algoritmos usados para resolução de problemas de otimização está relacionado à presença de mínimos locais. Um *ponto de mínimo local*  $\vec{X}_l \in S$  é definido como:

$$\exists \delta > 0 : f(\vec{X}_l) \leq f(\vec{X}), \quad \forall \vec{X} \in S \cap N_\delta(\vec{X}_l) \quad (7)$$

onde

$$N_\delta(\vec{X}_l) \stackrel{\text{def}}{=} \{\vec{X} \mid \|\vec{X} - \vec{X}_l\| < \delta\} \quad (8)$$

A presença de pontos de mínimos locais pode acarretar na convergência prematura de algoritmos de otimização fazendo com que eles fiquem presos a um mínimo local.

Neste contexto, podem-se definir as funções de acordo com a quantidade de pontos de mínimos. Por definição, uma função é denominada *multimodal* se ela possui múltiplos pontos de mínimos locais. Por outro lado, uma função é denominada *unimodal* se possui apenas um ponto de mínimo que é equivalente ao ponto de mínimo global.



### 3 EVOLUÇÃO DIFERENCIAL

A evolução diferencial é um algoritmo considerado simples, robusto e eficiente na solução de problemas de otimização. Este algoritmo é uma meta heurística baseada em estratégias evolutivas populacionais apresentado pela primeira vez por Storn e Price (1995) para resolução de problemas de otimização de funções contínuas não lineares e não diferenciáveis.

O conceito básico do algoritmo ED é a utilização de uma população de indivíduos que representam soluções potenciais dentro do espaço de busca do problema a ser resolvido e que evoluem ao longo de gerações utilizando-se para isto de operadores genéticos.

No algoritmo ED, o processo de otimização inicia-se pela criação da população inicial e em seguida, a cada geração, esta população é eventualmente modificada dando lugar a uma nova população de mesmo tamanho constituída por novos indivíduos que se mostrem melhores ou mais promissores formando assim uma nova geração.

O processo de modificação da população a cada geração é realizada por meio de operadores genéticos que foram inspirados no princípio da evolução e seleção natural. Mais especificamente, o algoritmo ED utiliza três operadores genéticos: *mutação*, *cruzamento* e *seleção*.

Neste processo, a modificação da população inicia-se pela geração de novos indivíduos denominados *vetores modificados* ou *doadores*. Para cada indivíduo da população, denominado *vetor alvo*, um vetor doador é gerado. Na versão clássica do algoritmo ED, os vetores doadores são gerados pela adição da diferença ponderada entre dois indivíduos da população escolhidos aleatoriamente e uniformemente a um terceiro indivíduo da população também escolhido aleatoriamente e uniformemente. A esta operação é dado o nome de *mutação*.

Em seguida, as componentes do vetor doador são misturadas com as componentes do respectivo vetor alvo gerando, assim, o chamado *vetor tentativa* ou *vetor experimental*. A esta operação é dado o nome de *cruzamento*.

Por último, o vetor experimental é comparado com o respectivo vetor alvo. Caso o vetor experimental corresponda a uma melhor solução de acordo com algum critério de seleção pré-estabelecido, então o vetor alvo será substituído pelo vetor

experimental na próxima geração. Caso contrário, o vetor alvo é mantido. A esta operação é dado o nome de *seleção*.

Este processo é repetido até que um critério de parada seja atingido. Normalmente utiliza-se como critério de parada quando se atinge um número máximo de gerações ou quando o melhor indivíduo encontrado fornecer uma solução dentro de uma precisão pré-estabelecida.

Na próxima seção é apresentada com mais detalhes cada uma das etapas da versão clássica do algoritmo ED.

### 3.1 ALGORITMO CLÁSSICO ED

A primeira versão do algoritmo ED, também conhecida como a versão clássica, foi desenvolvida e apresentada por Storn e Price (1995).

ED é um algoritmo de busca que utiliza um conjunto de  $NP$  vetores  $D$ -dimensionais denominado *população*. O parâmetro  $NP$  define o tamanho da população e é fixo não sofrendo, portanto, alteração durante todo o processo de otimização. Cada vetor da população é denominado um *indivíduo da população* e possui dimensão  $D$ , onde  $D$  representa o número de parâmetros do problema a ser otimizado.

Como os indivíduos da população podem ser modificados a cada geração, a seguinte notação será utilizada para identificar o  $i$ -ésimo indivíduo da população pertencente à geração atual  $G$ :

$$\vec{X}_{i,G} = [X_{1,i,G}, X_{2,i,G}, \dots, X_{D,i,G}] \quad (9)$$

para  $i = 1, 2, \dots, NP$  e  $G = 0, 1, \dots, Gmax$ , onde  $Gmax$  corresponde ao número máximo de gerações do processo de otimização.

Basicamente o algoritmo ED pode ser dividido em quatro etapas: inicialização, mutação, cruzamento e seleção.

### 3.1.1 Geração da população inicial

O primeiro passo do algoritmo ED é a geração da população inicial ( $G = 0$ ). A população inicial,  $\vec{X}_{i,0}$ ,  $i = 1, 2, \dots, NP$ , é formada por vetores gerados aleatoriamente e uniformemente distribuídos entre os limites máximo  $\vec{X}^{max} = [X_1^{max}, X_2^{max}, \dots, X_D^{max}]$  e mínimo  $\vec{X}^{min} = [X_1^{min}, X_2^{min}, \dots, X_D^{min}]$  do problema, cobrindo, assim, o espaço de busca.

Sendo assim, o cálculo da  $j$ -ésima componente do  $i$ -ésimo indivíduo da população inicial é realizado da seguinte forma:

$$X_{j,i,0} = X_j^{min} + rand_{j,i}[0,1] \cdot (X_j^{max} - X_j^{min}) \quad (10)$$

para  $i = 1, 2, \dots, NP$  e  $j = 1, 2, \dots, D$ .  $rand_{j,i}[0,1]$  corresponde ao resultado de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$  avaliado independentemente para cada  $j$ -ésima componente do  $i$ -ésimo indivíduo.

A geração da população inicial poderia ser realizada utilizando-se também distribuições não uniformes, como por exemplo, distribuições Gaussianas. Entretanto, quando não há nenhum conhecimento prévio sobre a localização do ponto de mínimo global (o que é normalmente o caso), recomenda-se a utilização de distribuição de probabilidade uniforme (PRICE; STORN; LAMPINEN, 2005).

### 3.1.2 Mutaç o

O operador muta o, tamb m chamado de *muta o diferencial*,   respons vel por gerar, a cada gera o  $G$ , um vetor doador  $\vec{V}_{i,G}$  para cada vetor alvo  $\vec{X}_{i,G}$ ,  $i = 1, 2, \dots, NP$ , atrav s da seguinte f rmula:

$$\vec{V}_{i,G} = \vec{X}_{r_0,G} + F \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}), \quad r_0 \neq r_1 \neq r_2 \neq i \quad (11)$$

onde  $r_0, r_1, r_2$  s o inteiros, escolhidos aleatoriamente tal que  $r_0, r_1, r_2 \in \{1, 2, \dots, NP\}$ .  $F$    denominado *fator de muta o* e   um n mero real e constante

pertencente ao intervalo  $[0,2]$  que controla a amplificação do chamado *vetor diferença* ( $\vec{X}_{r_1,G} - \vec{X}_{r_2,G}$ ).

Como pode ser observado na equação (11), na mutação diferencial um vetor  $\vec{X}_{r_0,G}$ , denominado como *vetor base*, é perturbado pela adição de um vetor diferença ( $\vec{X}_{r_1,G} - \vec{X}_{r_2,G}$ ) ponderado e escolhido aleatoriamente e uniformemente.

Este processo é ilustrado na FIGURA 1 para um caso de um espaço de busca bideimensional onde se observam as curvas de níveis de uma função objetivo arbitrária, bem como todos os vetores envolvidos na geração do vetor doador.

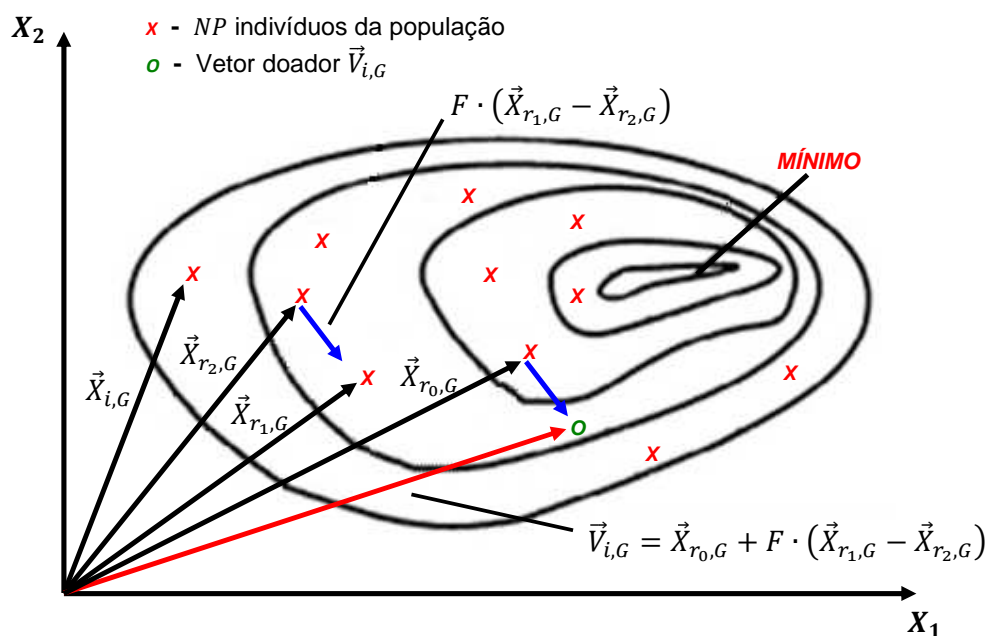


FIGURA 1: Exemplo de geração de um vetor doador.

FONTE: Adaptado de Storn e Price (1995)

### 3.1.3 Cruzamento

O operador de cruzamento é utilizado para aumentar a diversidade dos novos indivíduos que sofreram mutação. No cruzamento, as componentes do vetor doador  $\vec{V}_{i,G}$  são misturadas às componentes do respectivo vetor alvo  $\vec{X}_{i,G}$  gerando, assim, o vetor experimental  $\vec{U}_{i,G} = [U_{1,i,G}, U_{2,i,G}, \dots, U_{D,i,G}]$  de acordo com a seguinte expressão:

$$U_{j,i,G} = \begin{cases} V_{j,i,G}, & \text{se } rand_{j,i}[0,1] \leq CR \text{ ou } j = j_{rand} \\ X_{j,i,G}, & \text{caso contrário} \end{cases} \quad (12)$$

para  $i = 1, 2, \dots, NP$  e  $j = 1, 2, \dots, D$ .  $CR \in [0,1]$ , denominado como *taxa de cruzamento*, é um número real e constante que define a probabilidade do cruzamento ocorrer representando, portanto, a probabilidade do vetor experimental herdar as componentes do vetor doador.  $j_{rand}$  é um número inteiro escolhido aleatoriamente e uniformemente a partir de  $\{1, 2, \dots, D\}$  cuja função é assegurar que o vetor experimental herde pelo menos uma componente do vetor doador.

O cruzamento definido pela equação (12) é denominado como *cruzamento binomial* ou *uniforme*, pois todas as componentes do vetor experimental têm a mesma probabilidade de herdar seus valores do vetor doador (PRICE; STORN; LAMPINEN, 2005).

#### 3.1.4 Seleção

A última operação do algoritmo ED é a seleção. Durante esta operação, o melhor entre o vetor alvo  $\vec{X}_{i,G}$  e o seu respectivo vetor experimental  $\vec{U}_{i,G}$  é escolhido para entrar na população na geração seguinte. Basicamente, a qualidade de cada indivíduo é avaliada utilizando-se a função objetivo. Caso o vetor experimental  $\vec{U}_{i,G}$  resulte em um valor para a função objetivo menor que o vetor alvo  $\vec{X}_{i,G}$ , o vetor alvo  $\vec{X}_{i,G}$  será substituído pelo vetor experimental  $\vec{U}_{i,G}$  na próxima geração. Neste caso diz-se que o vetor experimental  $\vec{U}_{i,G}$  foi bem sucedido. Caso contrário, o vetor experimental  $\vec{U}_{i,G}$  é descartado e o vetor alvo  $\vec{X}_{i,G}$  é mantido para a próxima geração.

Em outras palavras, este processo pode ser escrito como segue:

$$\vec{X}_{i,G+1} = \begin{cases} \vec{U}_{i,G}, & \text{se } f(\vec{U}_{i,G}) < f(\vec{X}_{i,G}), \\ \vec{X}_{i,G}, & \text{caso contrário} \end{cases}, \quad i = 1, 2, \dots, NP \quad (13)$$

para  $G = 0, 1, \dots, Gmax$ .  $f(\cdot)$  representa a função objetivo do problema a ser minimizado.

### 3.2 VARIAÇÕES DO ALGORITMO ED

A versão do algoritmo ED apresentado na seção 3.1 não é a única versão deste algoritmo. Ao longo dos anos, inúmeras variantes foram desenvolvidas aplicando-se modificações nos operadores de mutação ou cruzamento. Sendo assim, com o intuito de facilitar a classificação e identificação destas diferentes variantes, utiliza-se a notação *DE/x/y/z* introduzida por Storn e Price (1997) onde:

- *x* – especifica o vetor base a ser perturbado;
- *y* – determina o número de vetores diferenças utilizados para perturbar o vetor base;
- *z* – indica o esquema de cruzamento adotado.

A versão clássica do algoritmo ED apresentada na seção 3.1 é definida como *DE/rand/1/bin* (STORN; PRICE, 1997), pois o vetor base é escolhido aleatoriamente e uniformemente ( $x = rand$ ), apenas 1 vetor diferença é utilizado para perturbar o vetor base ( $y = 1$ ) e o cruzamento é do tipo binomial ( $z = bin$ )

Algumas das variantes frequentemente usadas na literatura alteram o operador mutação selecionando o vetor base baseando-se no valor de sua função objetivo. Este é o caso das versões *DE/best/1/bin*, *DE/best/2/bin* e *DE/rand-to-best/1/bin*<sup>1</sup> apresentadas por Storn (1996) nas quais se utiliza o indivíduo da população atual que apresenta o menor valor para a função objetivo para formar o vetor a ser perturbado.

Nestas versões, o vetor doador  $\vec{V}_{i,G}$  é gerado da seguinte forma:

- *DE/best/1/bin*:

$$\vec{V}_{i,G} = \vec{X}_{best,G} + F \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) \quad (14)$$

- *DE/best/2/bin*:

$$\vec{V}_{i,G} = \vec{X}_{best,G} + F \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) + F \cdot (\vec{X}_{r_3,G} - \vec{X}_{r_4,G}) \quad (15)$$

- *DE/rand-to-best/1/bin*:

$$\vec{V}_{i,G} = \vec{X}_{i,G} + K \cdot (\vec{X}_{best,G} - \vec{X}_{i,G}) + F \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) \quad (16)$$

<sup>1</sup> A estratégia *DE/rand-to-best* também é conhecida na literatura como *DE/target-to-best* (PRICE; STORN; LAMPINEN, 2005).

onde  $r_1, r_2, r_3, r_4$  são escolhidos aleatoriamente e uniformemente tal que  $r_1, r_2, r_3, r_4 \in \{1, 2, \dots, NP\}$  e  $r_1 \neq r_2 \neq r_3 \neq r_4 \neq i$ .  $\vec{X}_{best,G}$  corresponde ao indivíduo da população atual  $G$  que apresenta o menor valor para a função objetivo, ou seja,  $f(\vec{X}_{best,G}) \leq f(\vec{X}_{i,G})$  para qualquer  $i \in \{1, 2, \dots, NP\}$ .  $F$  é o mesmo fator de mutação utilizado na versão clássica (equação (11)) e  $K$  é um número real, constante que usualmente é escolhido  $K = F$  para reduzir o número de parâmetros de controle a serem ajustados (STORN, 1996).

A vantagem da estratégia *DE/best/1/bin* quando comparada com a estratégia clássica *DE/rand/1/bin*, é o fato de que a utilização do vetor  $\vec{X}_{best,G}$  como vetor base aumenta a velocidade de convergência devido a sua característica gulosa pois a geração dos vetores experimentais é direcionada para o melhor indivíduo da população corrente. Por outro lado, diminui-se a diversidade da população potencial de vetores experimentais aumentando-se o risco de uma convergência prematura, ou seja, aumenta-se o risco de o algoritmo ficar preso em um mínimo local e, portanto aumenta-se o risco de o algoritmo não convergir para o mínimo global.

O objetivo das estratégias *DE/best/2/bin* e *DE/rand-to-best/1/bin* é exatamente melhorar o compromisso entre velocidade de convergência e diversidade da população. Nestas duas versões a diversidade da população potencial de vetores experimentais é aumentada pela adição de dois vetores diferença ao vetor base no caso da variante *DE/best/2/bin* e pela escolha do vetor base localizado entre  $\vec{X}_{best,G}$  e o vetor alvo  $\vec{X}_{i,G}$  no caso da variante *DE/rand-to-best/1/bin*.

Outra variante comumente utilizada na literatura é a variante *DE/rand/2/bin* (HUANG; QIN; SUGANTHAN, 2006) na qual dois vetores diferenças são utilizados na formação do vetor doador  $\vec{V}_{i,G}$ :

$$\vec{V}_{i,G} = \vec{X}_{r_0,G} + F \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) + F \cdot (\vec{X}_{r_3,G} - \vec{X}_{r_4,G}) \quad (17)$$

onde  $r_0, r_1, r_2, r_3, r_4$  são inteiros, escolhidos aleatoriamente e uniformemente tal que  $r_0, r_1, r_2, r_3, r_4 \in \{1, 2, \dots, NP\}$ ,  $r_0 \neq r_1 \neq r_2 \neq r_3 \neq r_4 \neq i$  e  $F$  é o fator de mutação.

A utilização de dois vetores diferenças na formação do vetor doador pode resultar em uma melhor perturbação do que a variante clássica que utiliza apenas um vetor diferença para perturbar o vetor base, o que pode ser útil para evitar que o algoritmo fique preso a mínimos locais aumentando, portanto, as chances de alcançar o mínimo global (QIN; HUANG; SUGANTHAN, 2009). Entretanto, a desvantagem desta variante é a diminuição da velocidade de convergência.

O operador de cruzamento é conhecido por não possuir a propriedade de invariância rotacional. Por este motivo, Price (1999) propôs uma estratégia denominada *DE/crrant-to-rand/1* que possui a propriedade de invariância rotacional, ou seja, o desempenho da estratégia não depende da orientação do sistema de coordenadas na qual a função objetivo é avaliada. Na estratégia *DE/crrant-to-rand/1* o cruzamento é substituído por uma recombinação linear aritmética e o vetor experimental é calculado através da seguinte equação:

$$\vec{U}_{i,G} = \vec{X}_{i,G} + K \cdot (\vec{X}_{r_1,G} - \vec{X}_{i,G}) + F \cdot (\vec{X}_{r_2,G} - \vec{X}_{r_3,G}) \quad (18)$$

onde  $r_1, r_2, r_3$  são inteiros, escolhidos aleatoriamente e uniformemente tal que  $r_1, r_2, r_3 \in \{1, 2, \dots, NP\}$  e  $r_1 \neq r_2 \neq r_3 \neq i$ .  $F$  é o fator de mutação e  $K$  é o fator de recombinação normalmente escolhido aleatoriamente e uniformemente dentro do intervalo  $[0, 1]$ .

### 3.3 ESCOLHA DOS PARÂMETROS DE CONTROLE

Como foi apresentado na seção 3.1, o algoritmo ED é um algoritmo simples e que depende apenas de três parâmetros: o tamanho da população  $NP$ , o fator de mutação  $F$  e a taxa de cruzamento  $CR$ . Estes parâmetros devem ser definidos pelo usuário.

Entretanto a eficiência do algoritmo ED depende da escolha adequada destes parâmetros. Gamperle, Muller e Koumoutsakos (2002) demonstraram através de testes empíricos que a capacidade do algoritmo ED de encontrar o mínimo global e a sua velocidade de convergência são dependentes da escolha dos parâmetros de controle  $NP$ ,  $F$  e  $CR$ . Além disto, a escolha apropriada dos parâmetros depende da função a ser minimizada e da variante do algoritmo ED utilizada no processo de



otimização (GAMPERLE; MULLER; KOUMOUTSAKOS, 2002; PRICE; STORN; LAMPINEN, 2005).

Portanto a escolha dos valores dos parâmetros de controle não é uma tarefa fácil. Vários estudos foram realizados para determinar quais seriam os valores ideais para cada um destes parâmetros.

Nas seções subsequentes, será dada uma breve ideia de como devem ser realizadas estas escolhas.

### 3.3.1 Tamanho da população

Storn e Price (1997) sugerem que  $NP$  esteja entre  $5D$  e  $10D$  enquanto Gamperle, Muller e Koumoutsakos (2002) propõem que  $NP$  seja escolhido entre  $3D$  e  $8D$ . Por outro lado, Rönkkönen, Kukkonen e Price (2005) sugerem que  $NP$  seja escolhido entre  $2D$  e  $40D$ . Entretanto, como pode ser observado, os valores sugeridos ainda situam-se em uma larga gama de valores.

Basicamente, aumentando-se o tamanho da população, aumenta-se a diversidade do universo de potenciais vetores experimentais e minimiza-se, portanto, o risco de estagnação devido a uma convergência prematura. Isto se torna mais importante para funções multimodais. Por outro lado, grandes valores de  $NP$  aumentam o tempo computacional. Portanto a escolha de  $NP$  é um compromisso entre tempo de cálculo e convergência.

### 3.3.2 Fator de mutação

De acordo com Rönkkönen, Kukkonen e Price (2005), típicos valores para o fator de mutação  $F$  seriam  $0,4 < F \leq 0,95$  sendo  $F = 0,9$  um bom compromisso entre velocidade e probabilidade de convergência. Storn e Price (1997), por sua vez, sugerem que uma boa escolha inicial para  $F$  seria  $0,5$  e que valores menores que  $0,4$  e maiores que  $1$  poderiam ocasionar uma degradação do desempenho do algoritmo. Todavia, Gamperle, Muller e Koumoutsakos (2002) consideram que uma boa escolha inicial para  $F$  seria  $0,6$  e destacam também que embora valores elevados de  $F$  aumentem a probabilidade de o algoritmo escapar de um mínimo local, valores  $F > 1$  diminuem consideravelmente a velocidade de convergência.

Price, Storn e Lampinen (2005) estabelecem que o fator de mutação  $F$  deve possuir um limite superior e inferior. Com relação ao limite superior, o fator de mutação  $F$  deve ser escolhido de tal forma que  $F < 1$  porque se observou através de testes empíricos que valores de  $F > 1$  aumentam o tempo de convergência. Com relação ao limite inferior, o fator de mutação  $F$  deve ser de tal forma que  $F > 0,4$ . Isto se deve ao fato de que, em geral, o operador seleção tende a reduzir a diversidade da população, enquanto a mutação aumenta esta diversidade. Para evitar uma convergência prematura,  $F$  deve ser suficientemente grande para neutralizar o efeito da seleção. Além disto, valores pequenos de  $F$  reduzem a chance de se escapar de um mínimo local.

### 3.3.3 Taxa de cruzamento

$CR$  define a taxa de cruzamento. Quanto menor o valor de  $CR$ , menor a probabilidade de o vetor experimental receber componentes do vetor doador.

Para Gamperle, Muller e Koumoutsakos (2002) valores elevados para  $CR$  podem ocasionar convergência prematura ou diminuição da velocidade de convergência. Sendo assim, o valor para a taxa de cruzamento  $CR$  deve ser escolhido entre 0,3 e 0,9.

Rönkkönen, Kukkonen e Price (2005) sugerem que a taxa de cruzamento deve ser escolhida  $0 \leq CR \leq 0,2$  para funções separáveis ou  $0,9 \leq CR \leq 1$  para funções não separáveis e multimodais. Isto se deve ao fato de que é possível determinar o mínimo global otimizando cada um dos componentes da função separadamente no caso das funções separáveis. Isto já não é o caso das funções não separáveis para as quais é necessário que vários parâmetros sejam ajustados ao mesmo tempo para que o processo de otimização seja eficiente.

## 3.4 TRATAMENTO DAS RESTRIÇÕES

Durante o todo processo de otimização, após o cruzamento, uma ou mais componentes do vetor experimental podem estar fora do espaço de busca. Sendo assim, estas violações precisam ser tratadas para que a solução encontrada ao final

do processo de otimização seja uma solução viável, ou seja, que esteja dentro do espaço de busca.

Price, Storn e Lampinen (2005) apresentam quatro metodologias para tratar as restrições durante o processo de otimização no algoritmo ED. Estas metodologias podem ser resumidas como:

- a) *penalidade brick wall*: se qualquer componente do vetor experimental estiver fora do espaço de busca, é definido um valor o mais alto possível para a sua respectiva função objetivo para garantir que este vetor não seja selecionado novamente;
- b) *penalidade adaptativa*: nesta metodologia, adiciona-se ao valor da função objetivo do vetor que está violando a restrição um valor (penalidade) que depende do número de componentes do vetor experimental que estiverem fora do espaço de busca. Esta penalidade também pode depender da magnitude da violação;
- c) *reinicialização aleatória*: substituem-se as componentes do vetor experimental que violam as restrições por outras escolhidas aleatoriamente dentro do espaço de busca;
- d) *bounce back*: substituem-se as componentes do vetor experimental que violam as restrições por outras escolhidas aleatoriamente entre o limite violado e as respectivas componentes do vetor base utilizadas para gerar o vetor experimental.

A metodologia mais comum e que será utilizada neste trabalho será a metodologia *bounce back*.

Mais detalhadamente, na metodologia *bounce back*, se após cruzamento uma ou mais componentes do vetor experimental violarem as restrições laterais, as seguintes correções são feitas:

$$U_{j,i,G} = X_j^{min} + rand[0,1] \cdot (X_{j,r_0,G} - X_j^{min}), \quad \text{se } U_{j,i,G} < X_j^{min} \quad (19)$$

$$U_{j,i,G} = X_{j,r_0,G} + rand[0,1] \cdot (X_j^{max} - X_{j,r_0,G}), \quad \text{se } U_{j,i,G} > X_j^{max} \quad (20)$$

onde  $rand[0,1]$  é o resultado de um gerador uniforme de número aleatórios com resultado no intervalo  $[0,1]$  e  $X_{j,r_0,G}$  é a  $j$ -ésima componente do vetor  $\vec{X}_{r_0,G}$  utilizado para criar o vetor doador.

## 4 ESTADO DA ARTE

Ao longo dos últimos anos, inúmeras variantes do algoritmo ED foram desenvolvidas com o propósito de melhorar o desempenho no processo de otimização.

Nesta seção serão apresentadas brevemente algumas versões do algoritmo ED para a solução de problemas de otimização mono-objetivo que obtiveram um maior destaque no cenário científico (DAS; SUGANTHAN, 2011; SUGANTHAN, 2012).

Estes algoritmos serão usados para comparação durante avaliação do desempenho do algoritmo proposto neste trabalho.

### 4.1 JADE

Zhang e Sanderson (2009a) propuseram um novo algoritmo ED adaptativo denominado *J Adaptive Differential Evolution* (JADE). A principal contribuição deste algoritmo é a introdução de uma nova estratégia de mutação e um novo processo de adaptação dos parâmetros de controle.

JADE implementa uma nova estratégia de mutação, conhecida como *DE/current-to-pbest/1* podendo ou não utilizar um arquivo externo. O objetivo deste arquivo externo é fornecer informações sobre a direção de progresso dos indivíduos e também de melhorar a diversidade da população.

Na estratégia *DE/current-to-pbest/1*, para cada vetor alvo  $\vec{X}_{i,G}$ ,  $i = 1, 2, \dots, NP$  da população atual, um vetor doador  $\vec{V}_{i,G}$  é gerado de acordo com a seguinte equação:

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F_i \cdot (\vec{X}_{best,G}^p - \vec{X}_{i,G}) + F_i \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}), \quad r_1 \neq r_2 \neq i \quad (21)$$

onde  $F_i$  é o fator de mutação associado ao  $i$ -ésimo indivíduo  $\vec{X}_{i,G}$  sendo atualizado de forma adaptativa a cada geração.  $\vec{X}_{best,G}^p$  corresponde a um indivíduo escolhido aleatoriamente e uniformemente a partir do conjunto formado pelos  $\lambda$  melhores indivíduos da população atual onde  $\lambda$  é definido como:

$$\lambda = \max(\lfloor p \cdot NP \rfloor, 2) \quad (22)$$

sendo  $p$  um número real e constante pertencente ao intervalo  $(0, 1]$  e  $\lfloor \cdot \rfloor$  a função de define a parte inteira de um número real.

Nota-se na equação (22) que  $\lambda \geq 2$ . Isto é feito para garantir que ao menos dois melhores indivíduos sejam utilizados. Isto é mais importante no caso de populações pequenas.

Como já mencionado, JADE pode, opcionalmente, fazer uso de um arquivo externo que armazena as soluções inferiores recentemente exploradas, ou seja, vetores experimentais que forneceram piores resultados durante o processo de seleção para o problema em questão.

A principal diferença entre a utilização ou não de um arquivo externo está relacionada com a escolha dos indivíduos para a formação do vetor diferença ( $\vec{X}_{r_1,G} - \vec{X}_{r_2,G}$ ). Seja  $A$  o conjunto de soluções inferiores armazenadas recentemente exploradas e  $P$  a população corrente. Assim, os indivíduos  $\vec{X}_{r_1,G}$  e  $\vec{X}_{r_2,G}$  são escolhidos da seguinte forma:

- *DE/current-to-pbest/1 sem arquivo:*  $\vec{X}_{r_1,G}$  e  $\vec{X}_{r_2,G}$  são selecionados aleatoriamente a partir de  $P$ ;
- *DE/current-to-pbest/1 com arquivo:*  $\vec{X}_{r_1,G}$  é selecionado aleatoriamente a partir de  $P$  e  $\vec{X}_{r_2,G}$  é selecionado aleatoriamente a partir da união  $P \cup A$ .

A formação do arquivo  $A$  é bem simples. No início do processo de otimização o arquivo está vazio. Então, no final de cada geração, os vetores alvos, que foram substituídos pelos respectivos vetores experimentais durante a seleção, são adicionados ao arquivo. Se o tamanho do arquivo exceder um certo limite, algumas soluções são eliminadas de forma aleatória mantendo-se, portanto, o tamanho do arquivo fixo e dentro do limite estipulado.

Além da nova estratégia de mutação, JADE também implementa um novo processo de auto-adaptação dos parâmetros de controle que atualiza dinamicamente o fator de mutação e a taxa de cruzamento durante o processo de otimização.

A cada geração  $G$ , um fator de mutação  $F_i$  e uma taxa de cruzamento  $CR_i$  são gerados para cada vetor alvo  $\vec{X}_{i,G}$  da população atual de acordo com uma distribuição de Cauchy, com parâmetro de locação  $\mu_F$  e parâmetro de escala de 0,1, e de acordo com uma distribuição normal, com média  $\mu_{CR}$  e desvio padrão 0,1, respectivamente.

Os parâmetros  $\mu_{CR}$  e  $\mu_F$  são inicializados a 0,5 no início do processo de otimização e atualizados no final de cada geração, de acordo com as seguintes equações:

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{média}_L(S_F) \quad (23)$$

$$\mu_{CR} = (1 - c) \cdot \mu_{CR} + c \cdot \text{média}_A(S_{CR}) \quad (24)$$

onde  $c$  é uma constante real pertencente ao intervalo  $[0,1]$ .  $S_F$  é o conjunto de todos os fatores de mutação bem sucedidos e  $S_{CR}$  é o conjunto de todas as taxas de cruzamento bem sucedidas<sup>2</sup> na geração de  $G$ .  $\text{média}_A(\cdot)$  corresponde a média aritmética e  $\text{média}_L(\cdot)$  corresponde a média Lehmer que é definida como :

$$\text{média}_L(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F} \quad (25)$$

## 4.2 jDE

Um novo algoritmo adaptativo ED, denominado *j Differential Evolution* (jDE), foi proposto por Brest et al. (2006). Este algoritmo é baseado na versão clássica *DE/rand/1/bin*. Entretanto, os parâmetros de controle  $F$  e  $CR$  são adaptados durante o processo de otimização.

Nesta abordagem, um fator de mutação  $F_i$  e uma taxa de cruzamento  $CR_i$  são associados a cada indivíduo  $\vec{X}_{i,G}$  da população atual. Estes parâmetros são então utilizados para a geração do vetor experimental  $\vec{U}_{i,G}$  que irá concorrer com  $\vec{X}_{i,G}$ .

---

<sup>2</sup>  $F_i$  e  $CR_i$  são considerados bem sucedidos se o vetor experimental  $\vec{U}_{i,G}$ , gerado utilizando-se estes parâmetros de controle, for melhor que seu respectivo vetor alvo  $\vec{X}_{i,G}$

Os parâmetros  $F_i$  e  $CR_i$  são atualizados durante o processo de otimização de acordo com duas probabilidades  $\tau_1$  e  $\tau_2$ . No início do processo de otimização  $F_i$  e  $CR_i$  são inicializados a 0,5 e 0,9, respectivamente, e, em seguida, são alterados no início de cada geração  $G$  de acordo com as seguintes equações:

$$F_{i,G+1} = \begin{cases} F_i + rand_1[0,1] \cdot F_u, & \text{se } rand_2[0,1] < \tau_1 \\ F_{i,G}, & \text{caso contrário} \end{cases} \quad (26)$$

$$CR_{i,G+1} = \begin{cases} rand_3[0,1], & \text{se } rand_4[0,1] < \tau_2 \\ CR_{i,G}, & \text{caso contrário} \end{cases} \quad (27)$$

para  $i = 1, 2, \dots, NP$ .  $rand_j[0,1]$ ,  $j \in \{1,2,3,4\}$  corresponde ao resultado de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$ .  $\tau_1$  e  $\tau_2$  são números reais e constantes  $\in [0,1]$  que representam a probabilidade de alteração dos parâmetros  $F_i$  e  $CR_i$ , respectivamente. Os parâmetros  $F_l$  e  $F_u$  são valores fixos a serem definidos pelo usuário e correspondem aos limites inferior e superior respectivamente do fator de mutação.

Como  $F_{i,G+1}$  e  $CR_{i,G+1}$  são gerados no início de cada geração, antes de a mutação ocorrer, eles influenciam, portanto, os operadores de mutação, cruzamento e seleção na geração do novo indivíduo  $\vec{X}_{i,G+1}$ .

A idéia por trás deste processo de adaptação dos parâmetros de controle  $F$  e  $CR$  é de que os “valores mais adaptados dos parâmetros levam a melhores indivíduos, que por sua vez, possuem maiores chances de sobreviver e de produzir descendentes e, portanto, propagar estes valores mais adaptados” (BREST et al., 2006, tradução nossa).

### 4.3 CoDE

Recentemente, uma nova versão do algoritmo ED, denominado *Composite Differential Evolution* (CoDE), foi proposto por Wang, Cai e Zhang (2011). A idéia principal desta abordagem é a utilização de um conjunto de estratégias pré-definidas de geração do vetor experimental e um conjunto de combinações de valores pré-definidos para os parâmetros de controle  $F$  e  $CR$ , a fim de gerar um vetor experimental para competir com o vetor alvo.

O conjunto de estratégias de geração do vetor experimental consiste em:

- *DE/rand/1/bin*;
- *DE/rand/2/bin*;
- *DE/current-to-rand/1*.

O conjunto de combinações de valores pré-definidos para os parâmetros de controle  $F$  e  $CR$  consiste em:

- $F = 1,0$  e  $CR = 0,1$ ;
- $F = 1,0$  e  $CR = 0,9$ ;
- $F = 0,8$  e  $CR = 0,2$ .

No algoritmo CoDE, a cada geração e para cada vetor alvo, utilizam-se as três estratégias, cada uma associada a uma combinação de valores dos parâmetros de controle selecionada aleatoriamente a partir do conjunto de combinações de valores pré definidos, para gerar um novo vetor experimental.

Como resultado, três vetores experimentais são gerados para cada vetor alvo. Em seguida, o vetor experimental que produzir a melhor solução dentre os três vetores experimentais gerados é escolhido a fim de competir com o respectivo vetor alvo. A seleção é realizada da mesma forma que na versão clássica ED.

#### 4.4 EPSDE

O algoritmo identificado como *Ensemble of Parameters and mutation Strategies in Differential Evolution* (EPSDE) foi proposto por Mallipeddi et al. (2011). Este algoritmo também utiliza um conjunto pré-definido de estratégias de geração do vetor experimental e conjuntos de valores pré-definidos para os parâmetros de controle para gerar os vetores experimentais.

O conjunto de estratégias de geração de vetor experimental consiste em:

- *DE/best/2/bin*;
- *DE/rand/1/bin*;
- *DE/current-to-rand/1*.

O conjunto de valores para o fator de mutação é definido como  $F \in \{0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9\}$  e o conjunto de valores para a taxa de cruzamento é definido como  $CR \in \{0,4, 0,5, 0,6, 0,7, 0,8, 0,9\}$ .



No algoritmo EPSDE, cada indivíduo da população inicial é associado a uma estratégia de geração de vetor experimental, a um valor de fator de mutação e a um valor de taxa de cruzamento, cada um sendo selecionado aleatoriamente a partir de seus respectivos conjuntos.

Em seguida, para cada vetor alvo, um vetor experimental é gerado utilizando a respectiva combinação associada de estratégia de geração de vetor experimental e valores de fator de mutação e taxa de cruzamento. Se o vetor experimental gerado tiver um valor de função objetivo melhor que seu respectivo vetor alvo, então, a combinação usada para gerar o vetor experimental bem sucedido será armazenada e mantida para a próxima geração. Por outro lado, se o vetor alvo tiver um valor de função objetivo melhor, uma nova combinação será a ele associada para a próxima geração. Esta nova combinação de estratégia de geração de vetor experimental e valores de fator de mutação e taxa de cruzamento será escolhida aleatoriamente a partir de cada um dos respectivos conjuntos ou a partir das combinações bem sucedidas armazenadas com igual probabilidade.

Consequentemente, as combinações de estratégias de geração de vetor experimental e valores de fator de mutação e taxa de cruzamento que produzirem vetores experimentais bem sucedidos terão mais chances de serem utilizadas durante o processo de otimização (MALLIPEDDI et al., 2011).

#### 4.5 SaDE

Qin, Huang e Suganthan (2009) propuseram um algoritmo ED auto-adaptativo, denominado *Self-adaptive Differential Evolution* (SaDE), em que as estratégias de geração de vetor experimental e seus respectivos parâmetros de controle  $F$  e  $CR$  são gradualmente auto-adaptados a partir da experiência adquirida durante o processo de otimização na geração de soluções bem ou mal sucedidas.

SaDE utiliza um conjunto de estratégias de geração de vetor experimental definido por:

- *DE/rand/1/bin*;
- *DE/rand-to-best/2/bin*;
- *DE/rand/2/bin*;
- *DE/current-to-rand/1*.

A cada geração  $G$ , uma estratégia de geração de vetores experimentais é escolhida aleatoriamente para cada vetor alvo de acordo com uma probabilidade  $p_{k,G}$ , ( $k = 1,2,3,4$ ), onde  $p_{k,G}$  representa a probabilidade da  $k$ -ésima estratégia ser escolhida na geração  $G$ . Cada uma das probabilidades  $p_{k,G}$ , ( $k = 1,2,3,4$ ) é inicializada a 0,25 e, então, atualizada durante o processo de otimização através da seguinte equação:

$$p_{k,G} = \frac{S_{k,G}}{\sum_{k=1}^4 S_{k,G}} \quad (28)$$

onde

$$S_{k,G} = \frac{\sum_{g=G-LP}^{G-1} ns_{k,g}}{\sum_{g=G-LP}^{G-1} ns_{k,g} + \sum_{g=G-LP}^{G-1} nf_{k,g}} + \epsilon \quad (29)$$

onde  $ns_{k,g}$  é o número de vetores experimentais bem sucedidos na geração  $G$  gerados pela  $k$ -ésima estratégia.  $nf_{k,g}$  representa o número de vetores experimentais gerados pela  $k$ -ésima estratégia e descartados durante o processo de seleção na geração  $G$ .  $\epsilon$  é uma constante real não nula utilizada para evitar uma probabilidade nula caso  $ns_{k,g} = 0$ .

Além das estratégias de geração de vetor experimental, SaDE também promove uma auto-adaptação dos parâmetros de controle  $F$  e  $CR$ . A cada geração  $G$ , um fator de mutação  $F_i$  e uma taxa de cruzamento  $CR_i$  são gerados aleatoriamente para cada vetor alvo  $\vec{X}_{i,G}$  a partir de uma distribuição normal.  $F_i$  é gerado a partir de uma distribuição normal com média de 0,5 e desvio padrão de 0,3, enquanto  $CR_i$  é gerado a partir de uma distribuição normal com média  $CRm_k$ , ( $k = 1,2,3,4$ ) e desvio padrão de 0,1:

$$F_i \sim N(0,5, 0,3) \quad (30)$$

$$CR_i \sim N(CRm_k, 0,1) \quad (31)$$

para  $i = 1, 2, \dots, NP$ .

Um valor médio  $CRm_k$  é associado a cada uma das estratégias de geração de vetores experimentais ( $k = 1,2,3,4$ ). Dessa forma, o valor médio a ser utilizado

para o cálculo de  $CR_i$  dependerá da estratégia de geração de vetor experimental associado a  $\vec{X}_{i,G}$ .

Os parâmetros  $CRm_k$  são inicializados a 0,5 e, em seguida, com base nos valores de taxa de cruzamento que geraram vetores experimentais bem sucedidos durante o período de aprendizagem. Basicamente, ao final de cada geração,  $CRm_k$  é atualizado com o valor da mediana do conjunto formado pelos fatores de mutação bem sucedidos nas últimas  $LP$  gerações associados a  $k$ -ésima estratégia.

#### 4.6 DEGL

Com o objetivo de obter um melhor balanço entre *exploitation* e *exploration*, Das et al. (2009) desenvolveram um algoritmo baseado na evolução diferencial referenciado como *Differential Evolution with Global and Local neighborhoods* (DEGL).

Conforme definido por Das et al. (2009), *exploration* representa a capacidade de o algoritmo explorar toda a região do espaço de busca enquanto *exploitation* indica a capacidade de convergir para as soluções quase-ótimas o mais rapidamente possível.

No algoritmo DEGL, Das et al. (2009) propuseram uma nova estratégia de mutação baseada na variante *DE/rand-to-best/1*. Esta nova estratégia de mutação utiliza dois tipos de modelos de vizinhança, os quais são chamados de *modelo de vizinhança local* e *modelo de vizinhança global*. No modelo de vizinhança global, a mutação é realizada utilizando-se o indivíduo com o menor valor de função objetivo dentre toda a população na geração atual. Já no modelo de vizinhança local, a mutação é realizada utilizando-se o indivíduo que possua o menor valor de função objetivo dentre aqueles localizados ao redor de uma pequena vizinhança na geração atual.

Cada um dos modelos é responsável por gerar um vetor que combinados geram o vetor doador  $\vec{V}_{i,G}$ .

O modelo de vizinhança local é responsável por gerar um vetor  $\vec{L}_{i,G}$  denominado *vetor doador local*. Neste modelo, para cada vetor alvo  $\vec{X}_{i,G}$ , uma vizinhança de tamanho  $k$  ( $0 \leq k \leq \frac{(NP-1)}{2}$ ) é definida consistindo dos seguintes

vetores  $\vec{X}_{i-k,G}, \dots, \vec{X}_{i,G}, \dots, \vec{X}_{i+k,G}$ . O vetor doador local  $\vec{L}_{i,G}$  é então gerado utilizando-se a seguinte expressão:

$$\vec{L}_{i,G} = \vec{X}_{i,G} + \eta \cdot (\vec{X}_{n\_best,i,G} - \vec{X}_{i,G}) + \psi \cdot (\vec{X}_{p,G} - \vec{X}_{q,G}) \quad (32)$$

onde  $\vec{X}_{n\_best,i,G}$  corresponde ao indivíduo que possua o menor valor de função objetivo dentre aqueles localizados na vizinhança do vetor alvo  $\vec{X}_{i,G}$  na geração atual.  $p$  e  $q$  são inteiros, escolhidos aleatoriamente e uniformemente tal que  $p, q \in \{i - k, i + k\}$  e  $p \neq q \neq i$ .  $\eta$  e  $\psi$  são denominados fatores de escala.

O modelo de vizinhança global é responsável por gerar um vetor  $\vec{g}_{i,G}$  denominado *vetor doador global*. Neste modelo, para cada vetor alvo  $\vec{X}_{i,G}$ , um vetor doador global  $\vec{g}_{i,G}$  é gerado a partir da seguinte equação:

$$\vec{g}_{i,G} = \vec{X}_{i,G} + \eta \cdot (\vec{X}_{g\_best,G} - \vec{X}_{i,G}) + \psi \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) \quad (33)$$

onde  $\vec{X}_{g\_best,G}$  corresponde ao indivíduo que possua o menor valor de função objetivo dentre todos os indivíduos da população na geração atual.  $r_1$  e  $r_2$  são inteiros, escolhidos aleatoriamente e uniformemente tal que  $r_1, r_2 \in \{1, NP\}$  e  $r_1 \neq r_2 \neq i$ .  $\eta$  e  $\psi$  são os mesmos fatores de escala utilizados na geração do vetor doador local.

No final, os dois vetores  $\vec{L}_{i,G}$  e  $\vec{g}_{i,G}$  são combinados para formar o vetor doador  $\vec{V}_{i,G}$ :

$$\vec{V}_{i,G} = \theta \cdot \vec{g}_{i,G} + (1 - \theta) \cdot \vec{L}_{i,G} \quad (34)$$

O cruzamento e a seleção são realizados da mesma forma que no algoritmo clássico ED.

O parâmetro  $\theta \in [0,1]$  da equação (34) é responsável por controlar o balanço entre a *exploration* e *exploitation* do algoritmo. Em seu trabalho, Das et al. (2009) propuseram vários esquemas para determinação do parâmetro  $\theta$ . O esquema que será utilizado neste trabalho será o esquema no qual o parâmetro  $\theta$  é auto-adaptado durante o processo de otimização.

#### 4.7 DEahcSPX

Noman e Iba (2008) propuseram um novo algoritmo denominado *Differential Evolution using Adaptive Hill-Climbing crossover-based local search and Simplex Crossover*, também identificado por DEahcSPX, que combina o algoritmo clássico ED com uma busca local. O objetivo deste algoritmo é unir a capacidade de *exploration* do algoritmo ED e a capacidade de *exploitation* dos algoritmos de busca local, combinando-as de uma forma equilibrada.

No algoritmo DEahcSPX, a cada geração  $G$ , antes de se executar os operadores do algoritmo ED (mutação, cruzamento, seleção), uma estratégia de busca local denominada *Adaptive Hill-Climbing Crossover-based Local Search (AHCXLS)* é aplicada. A estratégia AHCXLS combina uma técnica de subida de encosta e um operador de cruzamento simplex proposto por Tsutsui, Yamamura e Higuchi (1999).

No algoritmo AHCXLS, utiliza-se um conjunto de  $np$  indivíduos formado por  $np - 1$  indivíduos selecionados aleatoriamente e o melhor indivíduo da população atual (indivíduo com o menor valor de função objetivo)  $\vec{X}_{best,G}$ . O operador de cruzamento simplex é então aplicado sobre os  $np$  indivíduos para gerar um vetor experimental  $C$ . Caso o vetor experimental  $C$  resulte em um valor de função objetivo menor que  $\vec{X}_{best,G}$ , o vetor  $\vec{X}_{best,G}$  será substituído pelo vetor experimental  $C$ .

O operador de cruzamento simplex é aplicado até que não haja mais melhoria na solução, ou seja, até que o vetor experimental  $C$  não seja melhor que o atual  $\vec{X}_{best,G}$ .

Após a execução da busca local AHCXLS, os operadores do algoritmo clássico ED (mutação, cruzamento, seleção) são aplicados.

## 5 ABORDAGEM DE EVOLUÇÃO DIFERENCIAL HÍBRIDA AUTO ADAPTATIVA

Nesta seção, uma nova versão de um algoritmo meta-heurístico populacional para resolução de problemas de otimização mono-objetivo sujeitos apenas a restrições laterais é apresentado.

A abordagem proposta neste trabalho baseia-se nas idéias da evolução diferencial e o algoritmo desenvolvido foi chamado de *Evolução Diferencial Híbrida Auto Adaptativa*, ou em inglês, *Hybrid Self Adaptive Differential Evolution (HSADE)*.

Conforme já detalhado na seção 3.3, o desempenho e a eficiência do algoritmo ED é dependente dos valores utilizados para o fator de mutação  $F$  e a taxa de cruzamento  $CR$ . Além disto, a escolha apropriada destes parâmetros depende da função a ser minimizada e da variante do algoritmo ED utilizada no processo de otimização. Conseqüentemente, para obter um ajuste adequado destes parâmetros, o método de tentativa e erro é geralmente utilizado o que normalmente é uma tarefa tediosa e demorada.

Além de uma boa configuração adequada dos parâmetros de controle, um compromisso adequado entre *exploration* e *exploitation* é importante para aumentar a precisão e a eficiência de um algoritmo estocástico de busca. Conseqüentemente, o algoritmo deve ser capaz de obter bom compromisso entre ser capaz de explorar o espaço de busca o máximo possível para aumentar a probabilidade de encontrar o ótimo global (*exploration*) e ser capaz de explorar a vizinhança de pontos do espaço de busca já visitados anteriormente de forma a aumentar a velocidade de convergência do processo de busca (*exploitation*) (CREPINSEK; LIU; MERNIK, 2013).

Com base nestas considerações, o objetivo da abordagem proposta neste trabalho é a implementação de um algoritmo que seja capaz de cumprir os seguintes pontos:

- Ter um equilíbrio adequado entre a *exploration* e *exploitation*, isto é, garantir que o algoritmo seja capaz de explorar cada região do espaço de busca (*exploration*) ao mesmo tempo em que é capaz de obter soluções ainda melhores a partir de um conjunto de boas soluções a cada geração (*exploitation*);

- Capaz de adaptar os valores dos parâmetros de controle durante o processo de otimização, o que é crucial para aumentar a velocidade de convergência e reduzir o risco de estagnação durante o processo de otimização.

Neste trabalho, duas versões para o algoritmo *HSADE* foram desenvolvidas. Cada uma destas versões será descrita em detalhe nas próximas seções.

A avaliação e análise do desempenho e eficiência de cada uma das versões serão apresentadas e discutidas no capítulo 6.

### 5.1 PRIMEIRA VERSÃO DO ALGORITMO HSADE (HSADE-v1)

O primeiro algoritmo proposto neste trabalho, denominado *HSADE-v1*, foi desenvolvido com o intuito de obter um algoritmo híbrido que fizesse uma sinergia entre um operador de busca voltado à *exploitation* e uma abordagem de evolução diferencial visando obter uma melhor precisão, eficiência e rapidez no processo de otimização. Com relação à abordagem diferencial, o objetivo foi de combinar as melhores características do algoritmo *JADE*, da otimização quase oposicional e de um esquema de mutação *ED* clássico de forma a obter um algoritmo com um melhor balanço entre *exploration* e *exploitation* na geração de novos indivíduos (FACHIN; COELHO, 2014).

De uma maneira geral, o algoritmo *HSADE-v1* pode ser descrito resumidamente da seguinte forma.

Em primeiro lugar, a população inicial é gerada através da aplicação de uma inicialização denominada quase oposicional. Em seguida, a cada geração, os vetores experimentais são gerados através de um método de busca voltado à *exploitation* ou de acordo com uma abordagem de evolução diferencial. Um parâmetro de probabilidade  $\beta$  é usado para decidir qual método será utilizado para a geração dos vetores experimentais. A seleção é feita de acordo com o operador de seleção clássico *ED*. Depois do operador de seleção ser executado, uma abordagem quase oposicional é aplicada sobre o indivíduo da população atual com o pior valor de função objetivo. No final, se uma estagnação da população é detectada, a população é reinicializada na vizinhança de um indivíduo da população atual escolhido aleatoriamente e uniformemente.

O pseudocódigo do algoritmo *HSADE-v1* é apresentado na FIGURA 2.

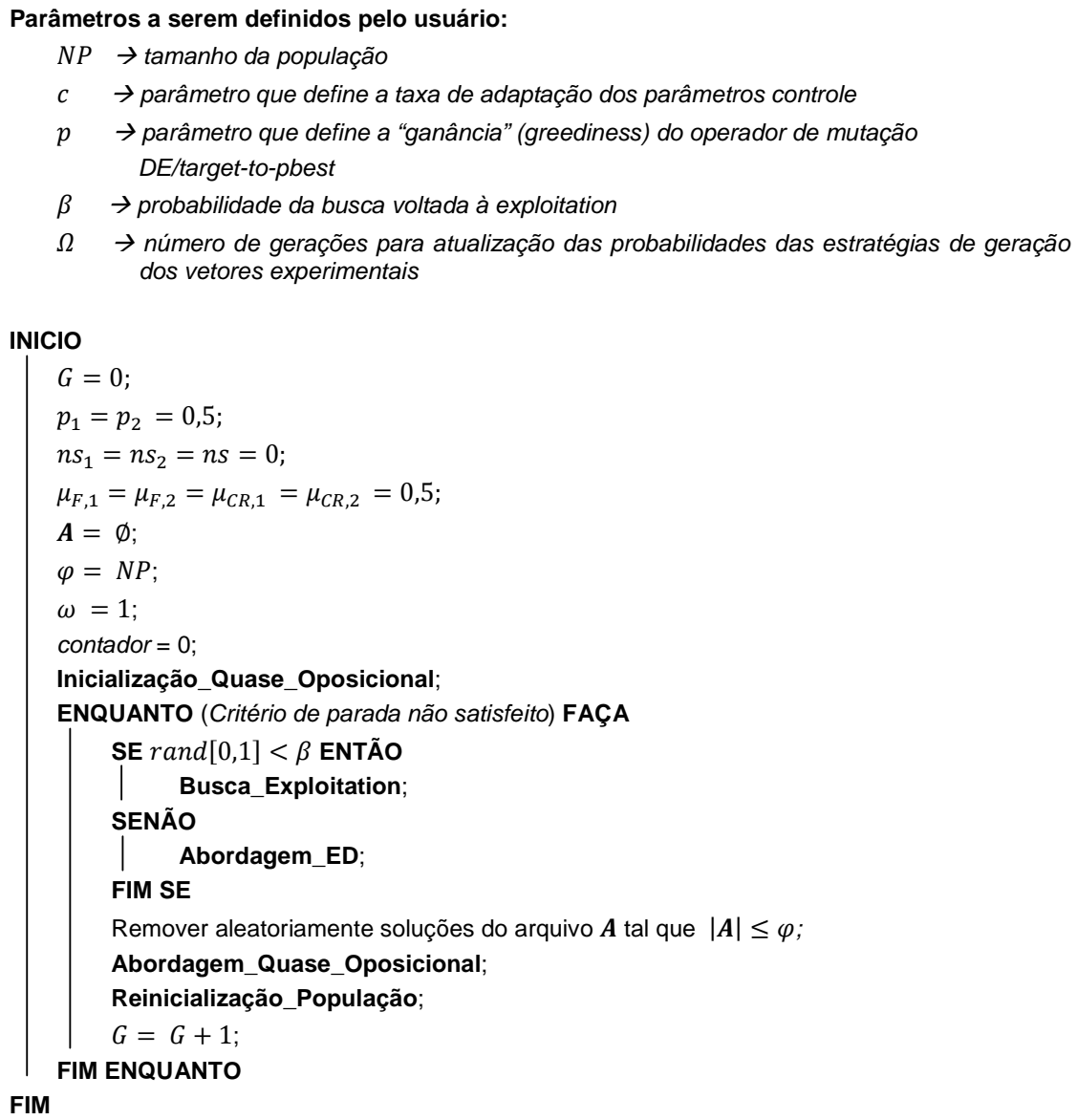


FIGURA 2: Pseudocódigo algoritmo HSADE-v1

FONTE: O autor (2014)

As principais etapas do algoritmo proposto HSADE-v1 serão descritas em detalhes nas seções subsequentes.

### 5.1.1 Inicialização da população

No algoritmo HSADE-v1, a inicialização da população é realizada aplicando-se o procedimento denominado inicialização quase oposicional proposto por



Rahnamayan, Tizhoosh e Salama (2007) que se baseia na utilização de vetores quase opostos para gerar a população inicial.

A grande vantagem de se utilizar a inicialização quase oposicional é devido ao fato de que se podem obter indivíduos iniciais mais aptos, mesmo quando não há conhecimento a priori sobre a solução (RAHNAMAYAN; TIZHOOSH; SALAMA, 2007).

Vetores quase opostos correspondem à generalização da definição de números quase opostos para o espaço  $D$ -dimensional.

Números quase-opostos são definidos da seguinte forma.

Dado  $x$  tal que  $a \leq x \leq b$ , onde  $a, b \in \mathbb{R}$ , o *quase oposto* de  $x$ , identificado por  $\check{x}^q$ , é dado por:

$$\check{x}^q = \begin{cases} M + rand[0,1] \cdot (\check{x} - M), & \text{se } x < M \\ \check{x} + rand[0,1] \cdot (M - \check{x}), & \text{caso contrário} \end{cases} \quad (35)$$

onde

$$\check{x} = a + b - x \quad (36)$$

$$M = \frac{a+b}{2} \quad (37)$$

$rand[0,1]$  corresponde ao resultado de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$ .  $\check{x}$  é denomiando o *oposto* de  $x$ .

Analogamente, a definição de números quase opostos pode ser estendida para dimensões superiores.

Seja  $\vec{X} = [x_1, x_2, \dots, x_D]$  um vetor  $D$ -dimensional tal que  $a_j \leq x_j \leq b_j$  para todo  $j = 1, 2, \dots, D$ , onde  $a_j, b_j \in \mathbb{R}$ . O vetor quase oposto  $\vec{X}^q = [\check{x}_1^q, \check{x}_2^q, \dots, \check{x}_D^q]$  de  $\vec{X}$  é dado por:

$$\check{x}_j^q = \begin{cases} M_j + rand_j[0,1] \cdot (\check{x}_j - M_j), & \text{se } x_j < M_j \\ \check{x}_j + rand_j[0,1] \cdot (M_j - \check{x}_j), & \text{caso contrário} \end{cases}, \quad \text{para } j = 1, 2, \dots, D \quad (38)$$

onde

$$\check{x}_j = a_j + b_j - x_j \quad (39)$$

$$M_j = \frac{a_j + b_j}{2} \quad (40)$$

$\tilde{x}_j$  é o oposto da  $j$ -ésima componente de  $\vec{X}$  e  $rand_j[0,1]$  corresponde ao resultado de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$  avaliado independentemente para cada  $j$ -ésima componente de  $\vec{X}$ .

Este processo é ilustrado na FIGURA 3 para um caso bidimensional. Nesta figura é apresentado o cálculo do vetor quase oposto de um vetor  $\vec{X}$  em um espaço de busca arbitrário  $S$ . Pode-se observar que neste exemplo o vetor quase oposto  $\vec{X}^q$  será localizado em algum ponto sobre a reta que une os vetores  $M$  e  $\vec{X}$ .

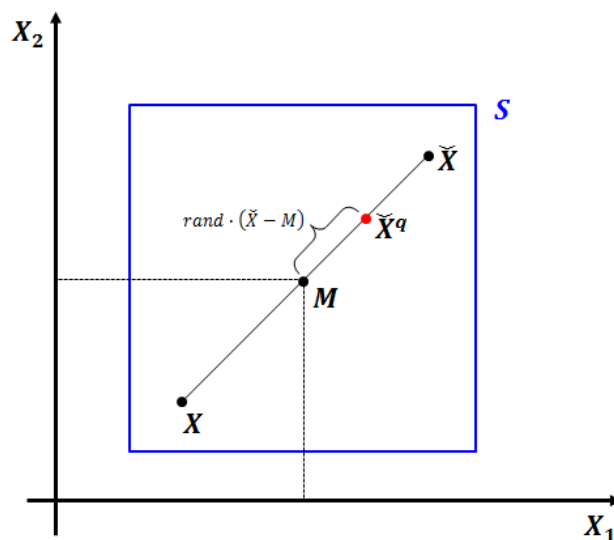


FIGURA 3: Exemplo bidimensional de geração de um vetor quase oposto  
 FONTE: O autor (2014)

Partindo-se da definição de vetores quase opostos, a ideia da inicialização quase oposicional é de gerar duas populações: uma população gerada de forma aleatória e sua respectiva população quase oposta. Em seguida, os  $NP$  melhores indivíduos do conjunto formado pelas duas populações são selecionados para formar a população inicial.

Mais precisamente, a população inicial é gerada de acordo com o seguinte procedimento.

Inicialmente, uma população formada por  $NP$  indivíduos é gerada de forma aleatória e uniformemente distribuída no espaço de busca. Esta população é denotada por  $P_0$ . Em seguida, a população quase oposta correspondente, denotada por  $QOP_0$ , é calculada de acordo com a definição de vetores quase oposto (equações (38), (39) e (40)) da seguinte forma:

$$QOP_{0,j,i} = \begin{cases} M_j + rand_{j,i}[0,1] \cdot (OP_{0,j,i} - M_j) & , \quad \text{se } P_{0,j,i} < M_j \\ OP_{0,j,i} + rand_{j,i}[0,1] \cdot (M_j - OP_{0,j,i}) & , \quad \text{caso contrário} \end{cases} \quad (41)$$

onde

$$OP_{0,j,i} = X_j^{min} + X_j^{max} - P_{0,j,i} \quad (42)$$

$$M_j = \frac{X_j^{min} + X_j^{max}}{2} \quad (43)$$

para  $i = 1, 2, \dots, NP$  e  $j = 1, 2, \dots, D$ .

Na sequência, os melhores  $NP$  indivíduos (aqueles que possuem o menor valor de função objetivo) da união  $\{P_0 \cup QOP_0\}$  são selecionados para formar a população inicial  $\{\vec{X}_{1,0}, \vec{X}_{2,0}, \dots, \vec{X}_{NP,0}\}$ .

O pseudocódigo do processo de inicialização quase oposicional utilizado no algoritmo HSADE-v1 é ilustrado na FIGURA 4.

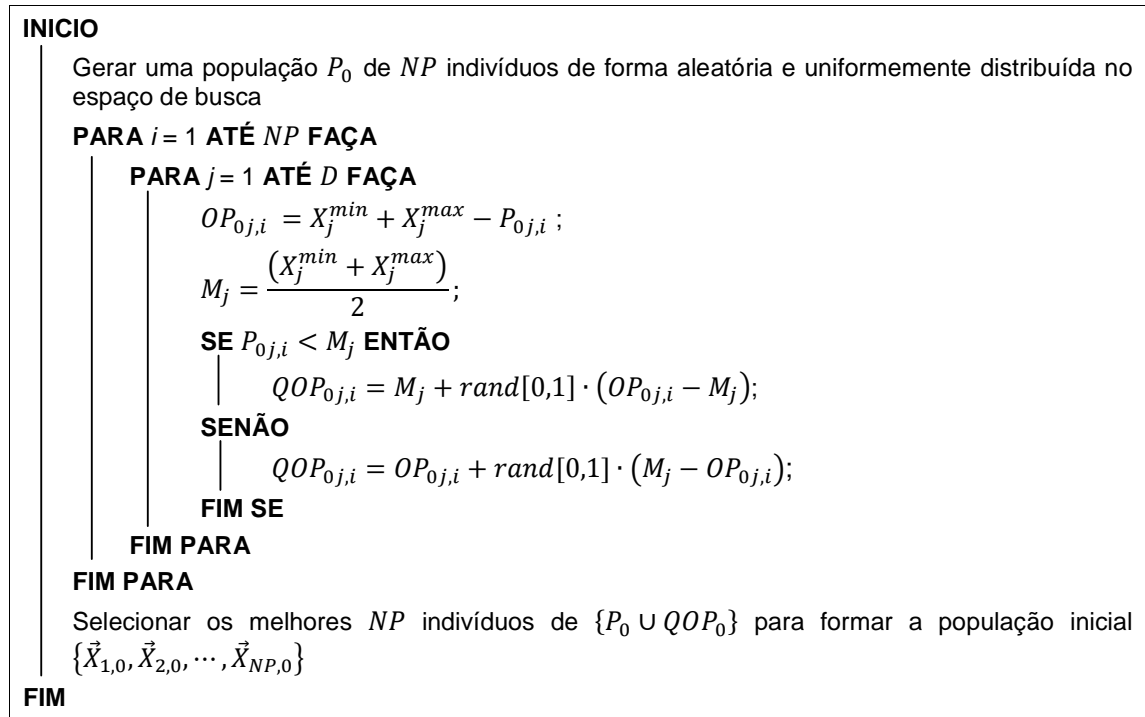


FIGURA 4: Pseudocódigo Inicialização\_Quase\_Oposicional

FONTE: O autor

### 5.1.2 Abordagem de Evolução Diferencial

Com o propósito de gerar uma população de vetores experimentais, uma abordagem de evolução diferencial é utilizada. A abordagem proposta neste trabalho e que é utilizada no algoritmo HSADE-v1 emprega um método em que os parâmetros de controle e a estratégia de geração do vetor experimental são auto-adaptados para os mais adequados durante o processo de otimização.

A população de vetores experimentais é gerada através desta abordagem de evolução diferencial de acordo com uma probabilidade  $(1 - \beta)$ .

#### 5.1.2.1 Adaptação da estratégia de geração do vetor experimental

Com o objetivo de melhorar o equilíbrio entre *exploration* e *exploitation* durante o processo de otimização, ao invés de usar apenas uma estratégia de geração de vetor experimental, o algoritmo proposto HSADE-v1 utiliza duas estratégias de geração de vetor experimental:

- *DE/current-to-pbest/1/bin* com arquivo externo
- *DE/rand/2/bin*

A variante *DE/current-to-pbest/1/bin* com arquivo externo, definida na seção 4.1, foi escolhida com o propósito de promover o *exploitation* e a variante *DE/rand/2/bin*, definida na seção 3.2, foi escolhida para promover o *exploration*.

Na variante *DE/current-to-pbest/1/bin*, os vetores experimentais são atraídos para as melhores soluções, o que aumenta a velocidade de convergência do processo de busca promovendo, dessa forma, a habilidade de *exploitation*. Todavia, isto faz com que haja uma diminuição da diversidade da população potencial de vetores experimentais o que reduz a capacidade de *exploration*.

Por outro lado, na variante *DE/rand/2/bin*, a adição de dois vetores diferenças ao vetor base promove a diversidade da população potencial de vetores experimentais aumentando, conseqüentemente, a capacidade de *exploration*. Porém, como resultado, há uma redução na velocidade de convergência do processo de busca, afetando, desta forma, a capacidade de *exploitation*.

Com relação à variante *DE/current-to-pbest/1/bin*, a versão com arquivo externo foi escolhida devido ao fato de a diversidade da população potencial de vetores experimentais ser melhorada utilizando-se soluções inferiores recentemente exploradas na geração do vetor experimental. Além disso, esta versão apresentou melhores resultados, quando comparada a versão sem arquivo, na resolução de problemas de otimização com dimensões elevadas (ZHANG; SANDERSON, 2009a). Entretanto, a utilização da variante *DE/current-to-pbest/1/bin* com arquivo externo introduz um novo parâmetro ao algoritmo: o tamanho do arquivo externo  $\varphi$ . Sendo assim, para evitar o ajuste de um novo parâmetro, neste trabalho, o tamanho do arquivo externo  $\varphi$  foi considerado igual ao tamanho da população  $NP$ . A formação do arquivo externo é realizada conforme descrito na seção 4.1.

Visto que duas estratégias de geração de vetor experimental são empregadas, na abordagem ED proposta no algoritmo HSADE-v1 um processo de seleção auto-adaptativo é utilizado para determinar qual estratégia será utilizada na geração do vetor experimental.

Neste processo de seleção, a cada geração  $G$  na qual a abordagem ED é aplicada, para cada vetor alvo  $\vec{X}_{i,G}$  é associada aleatoriamente uma das duas estratégias para gerar o vetor experimental de acordo com uma certa probabilidade  $p_m, m = 1, 2$ , onde  $p_m$  representa a probabilidade da  $m$ -ésima estratégia a ser escolhida.

A probabilidade de cada estratégia de geração de vetores experimentais  $p_m, m = 1, 2$  é inicializada a 0,5 (no início as duas estratégias têm a mesma probabilidade de ser escolhidas) e, em seguida, é atualizada, a cada  $\Omega$  gerações na qual a abordagem ED foi aplicada, de acordo com a seguinte equação:

$$p_m = 0,1 + 0,8 \cdot \frac{ns_m}{ns}, \quad \text{para } m = 1, 2 \quad (44)$$

onde  $ns_m$  é o número de vetores experimentais bem sucedidos nas últimas  $\Omega$  gerações, na qual a abordagem ED foi aplicada, gerados pela  $m$ -ésima estratégia e  $ns$  é o número total de vetores experimentais bem sucedidos nas últimas  $\Omega$  gerações, na qual a abordagem ED foi aplicada, gerados por qualquer estratégia ( $ns = ns_1 + ns_2$ ).

Pode ser visto a partir da equação ((44) que  $p_m \geq 0,1$ . Este limite inferior é um valor arbitrário para assegurar que qualquer uma das estratégias de geração do vetor experimental tenha uma probabilidade não nula de ser escolhida durante todo o processo de otimização. Além disso, após cada atualização das probabilidades  $p_m$ , os contadores  $ns_m$  e  $ns$  são reinicializados a 0. Isto é feito para que o cálculo das probabilidades  $p_m$  seja influenciado apenas pelo resultado das últimas  $\Omega$  gerações na qual a abordagem ED foi aplicada. Se após  $\Omega$  gerações  $ns = 0$ ,  $p_m, m = 1, 2$  será ajustado para 0,5.

### 5.1.2.2 Adaptação dos parâmetros de controle $F$ e $CR$

Conforme já mencionado anteriormente, a eficiência do algoritmo ED é dependente dos valores utilizados para o fator de mutação  $F$  e a taxa de cruzamento  $CR$ . Além disso, o ajuste destes parâmetros pode ser demorado, uma vez que valores apropriados dependem do problema a ser resolvido e, geralmente, o método de tentativa e erro é necessário para o ajuste correto destes parâmetros.

Devido a estes fatos, um processo de auto-adaptação dos parâmetros de controle  $F$  e  $CR$  durante o processo de otimização é implementado no algoritmo HSADE-v1 a fim de evitar o ajuste manual dos parâmetros.

O algoritmo HSADE-v1 utiliza um processo de adaptação dos parâmetros de controle  $F$  e  $CR$  semelhante ao descrito na seção 4.1. No entanto, no algoritmo HSADE-v1 são calculados parâmetros  $\mu_{CR}$  e  $\mu_F$  específicos para cada estratégia de geração do vetor experimental com o objetivo de obter parâmetros  $F$  e  $CR$  adaptados para cada estratégia.

No algoritmo proposto, a cada geração  $G$  na qual a abordagem ED é aplicada, e após selecionar a estratégia de geração de vetor experimental ( $m = 1$  ou  $2$ ), um fator de mutação  $F_i$  é gerado e associado a cada vetor alvo  $\vec{X}_{i,G}$ ,  $i = 1, 2, \dots, NP$ , de acordo com uma distribuição de Cauchy com parâmetro de locação  $\mu_{F,m}$ ,  $m = 1, 2$ , e parâmetro de escala de 0,1:

$$F_i = \begin{cases} \text{Cauchy}(\mu_{F,1}, 0,1), & \text{se estratégia } m = 1 \text{ é usada para gerar o } i - \text{ésimo vetor experimental} \\ \text{Cauchy}(\mu_{F,2}, 0,1), & \text{se estratégia } m = 2 \text{ é usada para gerar o } i - \text{ésimo vetor experimental} \end{cases} \quad (45)$$

sendo  $F_i$  truncado para 1 caso  $F_i > 1$  ou recalculado de acordo com a equação (45) se  $F_i \leq 0$ .

Os parâmetros de locação  $\mu_{F,m}$ ,  $m = 1,2$ , são inicializados a 0,5 e, em seguida, atualizados no final de cada geração na qual a abordagem ED é aplicada, caso tenha sido gerado algum vetor experimental bem sucedido, de acordo com a seguinte equação:

$$\mu_{F,m} = (1 - c) \cdot \mu_{F,m} + c \cdot \text{média}_L(S_{F,m}), \quad m = 1,2 \quad (46)$$

onde  $c$  é uma constante real pertencente ao intervalo  $[0,1]$  que define a taxa de adaptação dos parâmetros controle.  $S_{F,m}$  é o conjunto de todos os fatores de mutação bem sucedidos associados à estratégia  $m$  na geração  $G$ , na qual a abordagem ED foi aplicada, e  $\text{média}_L(.)$  corresponde a média Lehmer definida pela equação (25).

Da mesma forma, a cada geração  $G$  na qual a abordagem ED é aplicada, e depois de selecionar a estratégia de geração de vetor experimental ( $m = 1$  ou  $2$ ), uma taxa de cruzamento  $CR_i$  é gerada e associada a cada vetor alvo  $\vec{X}_{i,G}$ ,  $i = 1, 2, \dots, NP$ , de acordo com uma distribuição normal com média  $\mu_{CR,m}$ ,  $m = 1,2$  e desvio padrão 0,1, tal que:

$$CR_i = \begin{cases} N(\mu_{CR,1}, 0,1), & \text{se estratégia } m = 1 \text{ é usada para gerar o } i - \text{ésimo vetor experimental} \\ N(\mu_{CR,2}, 0,1), & \text{se estratégia } m = 2 \text{ é usada para gerar o } i - \text{ésimo vetor experimental} \end{cases} \quad (47)$$

sendo  $CR_i$  diretamente truncado para o intervalo  $[0,1]$  caso  $CR_i > 1$  ou  $CR_i < 0$ .

As médias  $\mu_{CR,m}$ ,  $m = 1,2$  são inicializadas a 0,5 e, em seguida, atualizadas no final de cada geração, na qual a abordagem ED é aplicada, caso tenha sido gerado algum vetor experimental bem sucedido, de acordo com a seguinte equação:

$$\mu_{CR,m} = (1 - c) \cdot \mu_{CR,m} + c \cdot \text{média}_A(S_{CR,m}), \quad m = 1,2 \quad (48)$$

onde  $c$  é a mesma constante definida na equação (46) e  $S_{CR,m}$  é o conjunto de todas as taxas de cruzamento bem sucedidas associadas à estratégia  $m$  na geração  $G$  na qual a abordagem ED foi aplicada, e  $média_A(.)$  representa a média aritmética.

Ao utilizar este processo de adaptação, os parâmetros de controle  $F$  e  $CR$  serão gradualmente adaptados para os valores mais adequados para o problema a ser resolvido e para diferentes fases do processo de otimização.

A idéia por trás deste processo de adaptação dos parâmetros de controle  $F$  e  $CR$  é de que os valores mais adaptados tendem a gerar indivíduos melhores e mais propensos a sobreviver e, portanto, estes valores devem ser propagados (ZHANG; SANDERSON, 2009b).

Com relação ao processo de adaptação do fator de mutação, o primeiro ponto a ser observado está relacionado ao fato da utilização de uma distribuição de Cauchy na geração dos  $F_i$ . Conforme apresentado na seção 3.3.2, o fator de mutação tem um papel importante no controle da diversidade do universo de potenciais vetores experimentais, impactando diretamente, portanto, a chance de se escapar de um mínimo local. A distribuição de Cauchy possui a propriedade de possuir uma calda longa o que reflete em uma maior diversidade dos fatores de mutação gerados, o que minimiza, portanto, o risco de estagnação devido a uma convergência prematura (ZHANG; SANDERSON, 2009a).

No caso do processo de adaptação do fator de cruzamento, uma distribuição normal é utilizada pois a propriedade de cauda longa da distribuição de Cauchy não é necessária.

Outro ponto a ser observado em relação ao processo de adaptação do fator de mutação, é o fato de se utilizar a média Lehmer ao invés da média aritmética. A média Lehmer é útil na propagação de maiores valores do fator de mutação, auxiliando, portanto, em uma maior perturbação dos vetores experimentais, o que reduz o risco de se ficar preso em um mínimo local (ZHANG; SANDERSON, 2009a).

O pseudocódigo da abordagem ED implementada no algoritmo HSADE-v1 é apresentado na FIGURA 5.



**INICIO**

$$S_{F,1} = S_{F,2} = S_{CR,1} = S_{CR,2} = 0;$$

$$\lambda = \max(\lfloor p \cdot NP \rfloor, 2);$$

**PARA  $i = 1$  ATÉ  $NP$  FAÇA**

Selecionar aleatoriamente, de acordo com as probabilidades  $p_1$  e  $p_2$ , uma das duas estratégias de geração do vetor experimental;

**SE Estratégia Seleccionada = 1 ENTÃO**

$$\text{Gerar } F_i = \text{Cauchy}(\mu_{F,1}, 0,1) \text{ e } CR_i = N(\mu_{CR,1}, 0,1);$$

Escolher aleatoriamente um indivíduo  $\vec{X}_{best,G}^p$  a partir dos  $\lambda$  melhores indivíduos da população atual  $\mathbf{P}$ ;

Escolher aleatoriamente  $\vec{X}_{r_1,G}$  a partir da população atual  $\mathbf{P}$  tal que  $\vec{X}_{r_1,G} \neq \vec{X}_{i,G}$ ;

Escolher aleatoriamente  $\vec{X}_{r_2,G}$  a partir da união  $\mathbf{P} \cup \mathbf{A}$ , tal que  $\vec{X}_{r_2,G} \neq \vec{X}_{r_1,G} \neq \vec{X}_{i,G}$

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F_i \cdot (\vec{X}_{best,G}^p - \vec{X}_{i,G}) + F_i \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G});$$

**SENÃO**

$$\text{Gerar } F_i = \text{Cauchy}(\mu_{F,2}, 0,1) \text{ e } CR_i = N(\mu_{CR,2}, 0,1);$$

Escolher aleatoriamente  $r_0, r_1, r_2, r_3, r_4 \in \{1,2,\dots, NP\}$  tal que  $r_0 \neq r_1 \neq r_2 \neq r_3 \neq r_4 \neq i$ ;

$$\vec{V}_{i,G} = \vec{X}_{r_0,G} + F_i \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) + F_i \cdot (\vec{X}_{r_3,G} - \vec{X}_{r_4,G});$$

**FIM SE**

Gerar  $\vec{U}_{i,G}$  aplicando o operador de cruzamento binomial;

Caso alguma componente de  $\vec{U}_{i,G}$  viole as restrições laterais, aplicar o método *bounce back*;

Avaliar  $f(\vec{U}_{i,G})$ ;

**SE  $f(\vec{X}_{i,G}) \leq f(\vec{U}_{i,G})$  ENTÃO**

$$\vec{X}_{i,G+1} = \vec{X}_{i,G};$$

**SENÃO**

$$\vec{X}_{i,G+1} = \vec{U}_{i,G};$$

$$\vec{X}_{i,G} \rightarrow \mathbf{A};$$

**SE Estratégia Seleccionada = 1 ENTÃO**

$$F_i \rightarrow S_{F,1};$$

$$CR_i \rightarrow S_{CR,1};$$

$$ns_1 = ns_1 + 1;$$

**SENÃO**

$$F_i \rightarrow S_{F,2};$$

$$CR_i \rightarrow S_{CR,2};$$

$$ns_2 = ns_2 + 1;$$

**FIM SE**

$$ns = ns + 1;$$

**FIM SE****FIM PARA**

$$\text{contador} = \text{contador} + 1;$$

```

SE contador = Ω ENTÃO
  |
  | SE ns = 0 ENTÃO
  |   |
  |   | p1 = p2 = 0,5 ;
  |   |
  |   | SENÃO
  |   |   |
  |   |   | p1 = 0,1 + 0,8 · (ns1/ns) ;
  |   |   | p2 = 0,1 + 0,8 · (ns2/ns) ;
  |   |   | ns1 = ns2 = ns = 0;
  |   |
  |   | FIM SE
  |   | contador = 0;
  |
  | FIM SE
  |
  | SE SF,1 ≠ ∅ E SCR,1 ≠ ∅ ENTÃO
  |   |
  |   | μF,1 = (1 - c) · μF,1 + c · médiaL(SF,1);
  |   | μCR,1 = (1 - c) · μCR,1 + c · médiaA(SCR,1);
  |   |
  |   | FIM SE
  |   |
  |   | SE SF,2 ≠ ∅ E SCR,2 ≠ ∅ ENTÃO
  |   |   |
  |   |   | μF,2 = (1 - c) · μF,2 + c · médiaL(SF,2);
  |   |   | μCR,2 = (1 - c) · μCR,2 + c · médiaA(SCR,2);
  |   |   |
  |   |   | FIM SE
  |   |
  |   | FIM SE
  |
  | FIM

```

FIGURA 5: Pseudocódigo Abordagem\_ED

FONTE: O autor (2014)

### 5.1.3 Busca voltada à *Exploitation*

A fim de melhorar a característica de *exploitation* do algoritmo proposto, e, conseqüentemente, aumentar a velocidade de convergência do processo de otimização, uma busca voltada à *exploitation* é aplicada de acordo com uma probabilidade  $\beta$ .

A idéia desta busca é a geração de vetores experimentais na vizinhança da população atual.

O operador de busca voltado à *exploitation* implementado no algoritmo HSADE-v1 é definido da seguinte forma: para cada vetor alvo  $\vec{X}_{i,G}$ , um vetor base  $\vec{X}_{r_0,G}$  é selecionado aleatoriamente ( $r_0 \neq i$ ). O vetor experimental correspondente  $\vec{U}_{i,G}$  é então gerado aleatoriamente na vizinhança de  $\vec{X}_{r_0,G}$ .

Mais precisamente, a cada geração  $G$ , com probabilidade  $\beta$ , o vetor experimental  $\vec{U}_{i,G}$  é gerado de acordo com as seguintes equações:

$$U_{j,i,G} = X_{j,r_0,G} - \Delta X_j + 2 \cdot rand_{j,i}[0,1] \cdot \Delta X_j \quad (49)$$

onde

$$\Delta X_j = \omega \cdot rand_{j,i}[0,1] \cdot (MAX_j^G - MIN_j^G) \quad (50)$$

$$MIN_j^G = \min(X_{j,1,G}, X_{j,2,G}, \dots, X_{j,NP,G}) \quad (51)$$

$$MAX_j^G = \max(X_{j,1,G}, X_{j,2,G}, \dots, X_{j,NP,G}) \quad (52)$$

para  $i = 1, 2, \dots, NP$  e  $j = 1, 2, \dots, D$ .  $r_0 \neq i$ .  $MIN_j^G$  e  $MAX_j^G$  representam os valores mínimo e máximo, respectivamente, da  $j$ -ésima componente da população atual que garantem que os vetores experimentais serão localizados na vizinhança da população atual (o conhecimento do espaço reduzido onde se encontra a população atual não é perdido).

O parâmetro  $\omega \in (0, 1]$  da equação (50) é um fator de ajuste com o objetivo de ajustar os movimentos dos vetores experimentais.  $\omega$  é inicializado a 1 e é reduzido cada vez que a busca voltada à *exploitation* não produzir nenhum vetor experimental bem sucedido, ou seja, quando todos os vetores experimentais gerados por esta busca são descartados durante a seleção. A redução é feita através da multiplicação do fator  $\omega$  por 0,75. O objetivo desta redução é de fazer com que a busca voltada à *exploitation* explore o interior de uma vizinhança cada vez menor toda vez que esta busca não é bem sucedida. Este valor de redução foi escolhido empiricamente, pois foi observado que uma redução maior faria com que o tamanho da vizinhança fosse reduzido muito rapidamente enquanto uma redução menor faria com que o tamanho da vizinhança fosse reduzido muito lentamente o que degradaria, em ambos os casos, a velocidade de convergência.

O pseudocódigo do operador de busca voltado à *exploitation* utilizado para gerar os vetores experimentais é ilustrado na FIGURA 6.

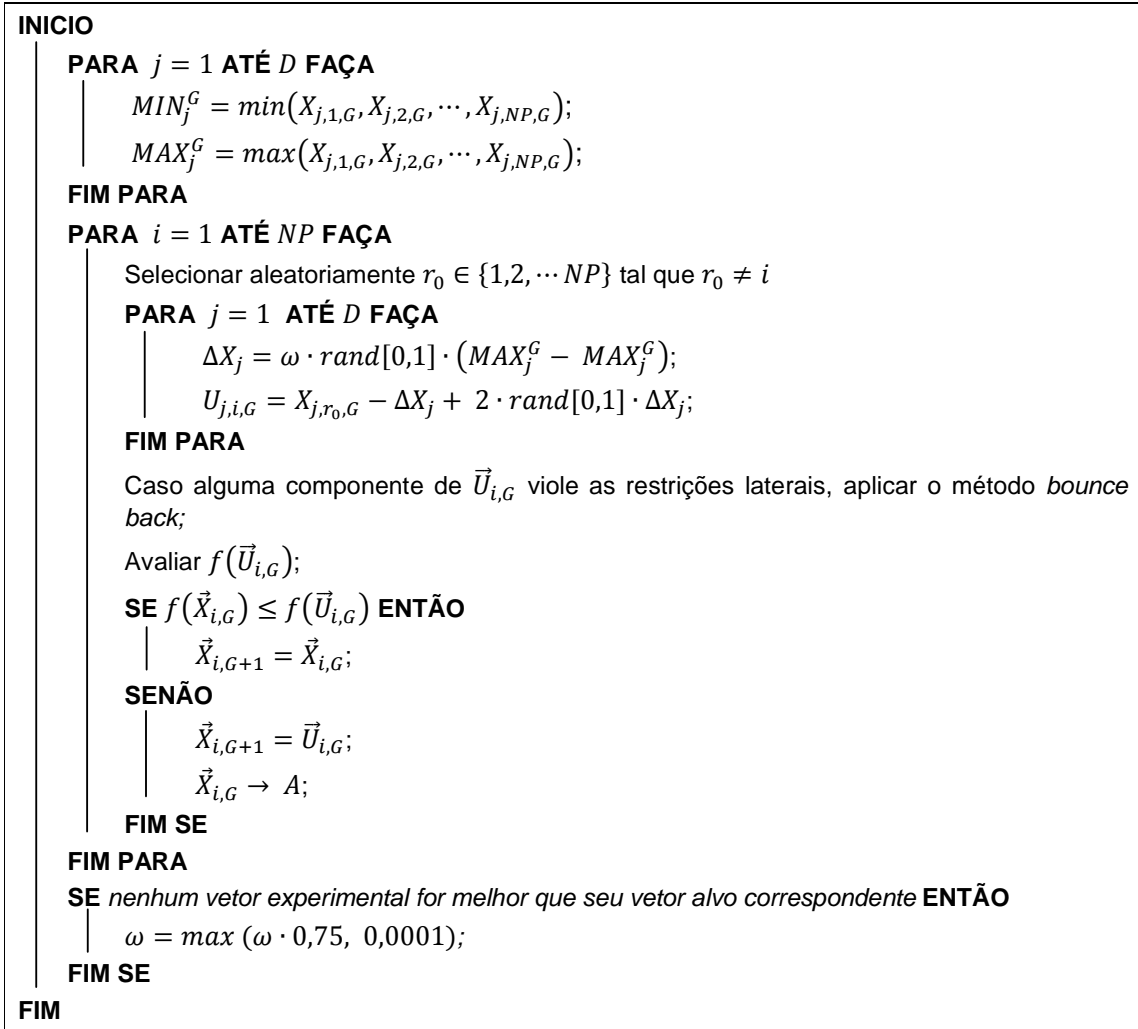


FIGURA 6: Pseudocódigo Busca\_Exploitation

FONTE: O autor (2014)

#### 5.1.4 Abordagem quase oposicional

Rahnamayan, Tizhoosh e Salama (2007) demonstraram matematicamente e também experimentalmente que a aplicação de vetores quase opostos aumenta a velocidade de convergência da evolução diferencial.

Portanto, no algoritmo HSADE-v1 proposto neste trabalho, vetores quase opostos são utilizados com o objetivo de melhorar a velocidade de convergência do processo de otimização.

Entretanto, ao invés de calcular os vetores quase oposto de toda a população com uma pré-determinada probabilidade como é realizado no trabalho de Rahnamayan, Tizhoosh e Salama (2007), na abordagem proposta no algoritmo

HSADE-v1, no final de cada geração  $G$ , após a geração da nova população, calcula-se o vetor quase oposto  $\vec{X}^q$  do indivíduo  $\vec{X}_{h,G+1}$  com o pior valor de função objetivo da população atual (  $f(\vec{X}_{i,G+1}) \leq f(\vec{X}_{h,G+1})$ ,  $\forall i \in \{1,2,\dots, NP\}$  ). Se  $\vec{X}_{h,G+1}$  tiver um valor de função objetivo melhor do que seu correspondente quase oposto  $\vec{X}^q$ ,  $\vec{X}_{h,G+1}$  será mantido na população. Caso contrário,  $\vec{X}_{h,G+1}$  será substituído por  $\vec{X}^q$  para a próxima geração.

No entanto, ao contrário de inicialização quase oposicional, o vetor quase oposto é calculado utilizando-se de limites calculados dinamicamente durante o processo de otimização, conforme definido por Rahnamayan, Tizhoosh e Salama (2007). Em outras palavras, ao invés de usar os limites pré-definidos do problema  $\vec{X}^{max} = [X_1^{max}, X_2^{max}, \dots, X_D^{max}]$  e  $\vec{X}^{min} = [X_1^{min}, X_2^{min}, \dots, X_D^{min}]$ , o vetor quase oposto é calculado com base nos valores mínimos e máximos de cada componente da população atual  $[MIN_j^{G+1}, MAX_j^{G+1}]$ , onde:

$$MIN_j^{G+1} = \min(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1}) \quad (53)$$

$$MAX_j^{G+1} = \max(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1}) \quad (54)$$

A utilização destes limites calculados de forma dinâmica faz com que o espaço de busca seja reduzido ao espaço onde se encontra a população atual, fazendo com que a informação de convergência da população não seja perdida.

Esta abordagem quase oposicional adotada no algoritmo HSADE-v1 pode ser descrita da seguinte forma:

1. O indivíduo  $\vec{X}_{h,G+1}$  com o pior valor de função objetivo é selecionado a partir da geração atual  $\{\vec{X}_{1,G+1}, \vec{X}_{2,G+1}, \dots, \vec{X}_{NP,G+1}\}$ :

$$f(\vec{X}_{h,G+1}) \geq f(\vec{X}_{i,G+1}), \quad \forall i \in [1, NP] \quad (55)$$

2. Na sequência, o vetor quase oposto correspondente  $\vec{X}^q$  é calculado a partir das equações (38), (39) e (40) onde os limites  $[a_j, b_j]$ ,  $j = 1, 2, \dots, D$ , são definidos como:

$$a_j = MIN_j^{G+1} = \min(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1}) \quad (56)$$

$$b_j = MAX_j^{G+1} = \max(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1}) \quad (57)$$

3. Finalmente, o melhor indivíduo entre  $\vec{X}_{h,G+1}$  e  $\check{X}^q$  é selecionado para permanecer na população:

$$\vec{X}_{h,G+1} = \begin{cases} \check{X}^q & , \quad \text{se } f(\check{X}^q) < f(\vec{X}_{h,G+1}) \\ \vec{X}_{h,G+1} & , \quad \text{caso contrário} \end{cases} \quad (58)$$

O pseudocódigo da abordagem quase oposicional implementada no algoritmo HSADE-v1 é apresentado na FIGURA 7.

```

INICIO
  Selecionar o indivíduo  $\vec{X}_{h,G+1}$  da geração atual com o pior valor de função objetivo;
  PARA  $j = 1$  ATÉ  $D$  FAÇA
     $MIN_j^{G+1} = \min(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1})$ 
     $MAX_j^{G+1} = \max(X_{j,1,G+1}, X_{j,2,G+1}, \dots, X_{j,NP,G+1})$ 
     $\check{X}_j = MIN_j^{G+1} + MAX_j^{G+1} - X_{j,h,G+1}$ ;
     $M_j = (MIN_j^{G+1} + MAX_j^{G+1})/2$ ;
    SE  $X_{j,h,G+1} < M_j$  ENTÃO
       $\check{X}_j^q = M_j + rand[0,1] \cdot (\check{X}_j - M_j)$ ;
    SE NÃO
       $\check{X}_j^q = \check{X}_j + rand[0,1] \cdot (M_j - \check{X}_j)$ ;
    FIM SE
  FIM PARA
  Avaliar  $f(\check{X}^q)$ ;
  SE  $f(\check{X}^q) < f(\vec{X}_{h,G+1})$  ENTÃO
     $\vec{X}_{h,G+1} = \check{X}^q$ 
  FIM SE
FIM

```

FIGURA 7: Pseudocódigo Abordagem\_Quase\_Oposicional

FONTE: O autor

### 5.1.5 Reinicialização da população

Caso ocorra uma estagnação da população durante o processo de otimização o algoritmo pára de prosseguir em direção às melhores soluções globais porque nenhuma combinação formada pelos operadores evolutivos de mutação e cruzamento leva a uma melhor solução.

Assim, para garantir que o algoritmo HSADE-v1 possa sempre continuar a encontrar soluções melhores, uma detecção de estagnação é implementada e a população é reinicializada em caso de estagnação.

Quando todos os indivíduos da população atual possuem o mesmo valor de função objetivo, a população é reinicializada na vizinhança de um indivíduo selecionado aleatoriamente de acordo com uma distribuição normal com valor médio 0 e desvio padrão  $\sigma_j$  definido como:

$$\sigma_j = \frac{(X_j^{max} - X_j^{min})}{4}, \quad j = 1, 2, \dots, D \quad (59)$$

onde  $\vec{X}^{max} = [X_1^{max}, X_2^{max}, \dots, X_D^{max}]$  e  $\vec{X}^{min} = [X_1^{min}, X_2^{min}, \dots, X_D^{min}]$  são as restrições laterais do problema a ser resolvido.

Além da reinicialização da população, os parâmetros  $\mu_{CR,m}, m = 1, 2$  e  $\mu_{F,m}, m = 1, 2$  são reinicializados a 0,5. Além disto, as probabilidades  $p_m, m = 1, 2$  também são reinicializadas a 0,5. Isto é feito para que estes parâmetros possam se adaptar à nova população, pois seus valores antigos poderiam não estar adaptados a nova população.

O pseudocódigo da reinicialização da população em caso de estagnação é apresentado na FIGURA 8.

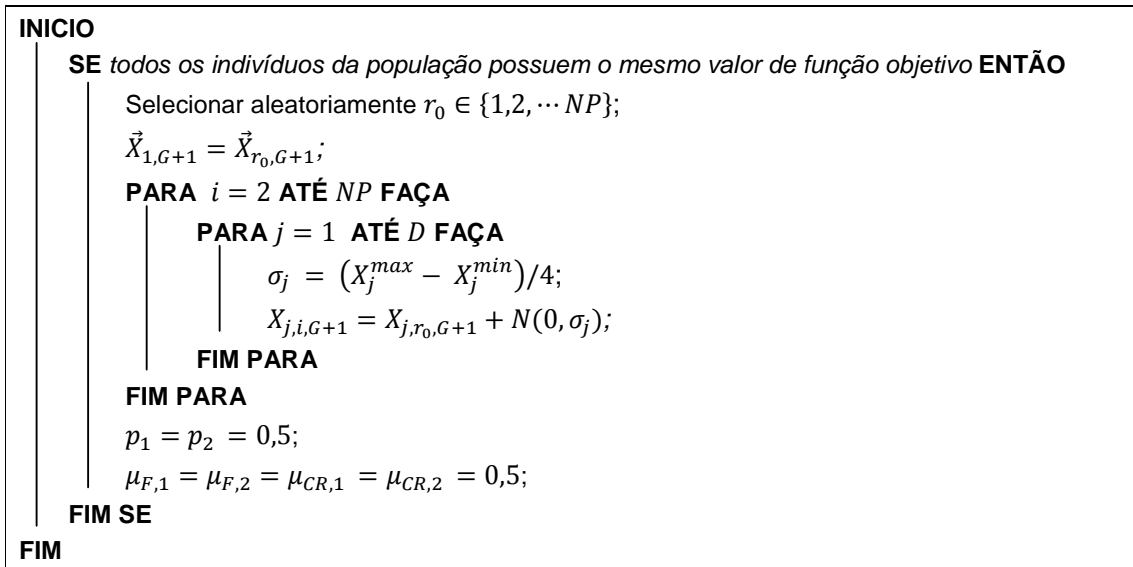


FIGURA 8: Pseudocódigo Reinicialização\_População

FONTE: O autor

## 5.2 SEGUNDA VERSÃO DO ALGORITMO HSADE (HSADE-v2)

O segundo algoritmo proposto neste trabalho, denominado HSADE-v2, foi desenvolvido baseando-se na primeira versão do algoritmo *HSADE*. Entretanto, o objetivo principal da implementação do algoritmo HSADE-v2 foi a de produzir um algoritmo híbrido que também fizesse uma sinergia entre um operador de busca voltado à *exploitation* e uma abordagem de evolução diferencial mas que possuísse o menor número possível de parâmetros de controle a serem ajustados pelo usuário. Além disto, um objetivo secundário foi de criar um algoritmo mais simples.

De uma maneira geral, o algoritmo HSADE-v2 pode ser descrito resumidamente da seguinte forma. Primeiramente, a população inicial é gerada de forma aleatória e uniformemente distribuída no espaço de busca. Em seguida, a cada geração, os vetores experimentais são gerados através de um método de busca voltado à *exploitation* ou de acordo com uma abordagem de evolução diferencial. Um parâmetro de probabilidade  $\beta$ , auto-adaptado durante todo o processo de otimização, é usado para decidir qual método será utilizado para a geração dos vetores experimentais. A seleção é feita de acordo com o operador de seleção clássico ED. No final, se uma estagnação da população é detectada, a população é reinicializada na vizinhança de um indivíduo da população atual escolhido aleatoriamente.



O pseudocódigo do algoritmo HSADE-v2 é apresentado na FIGURA 9.

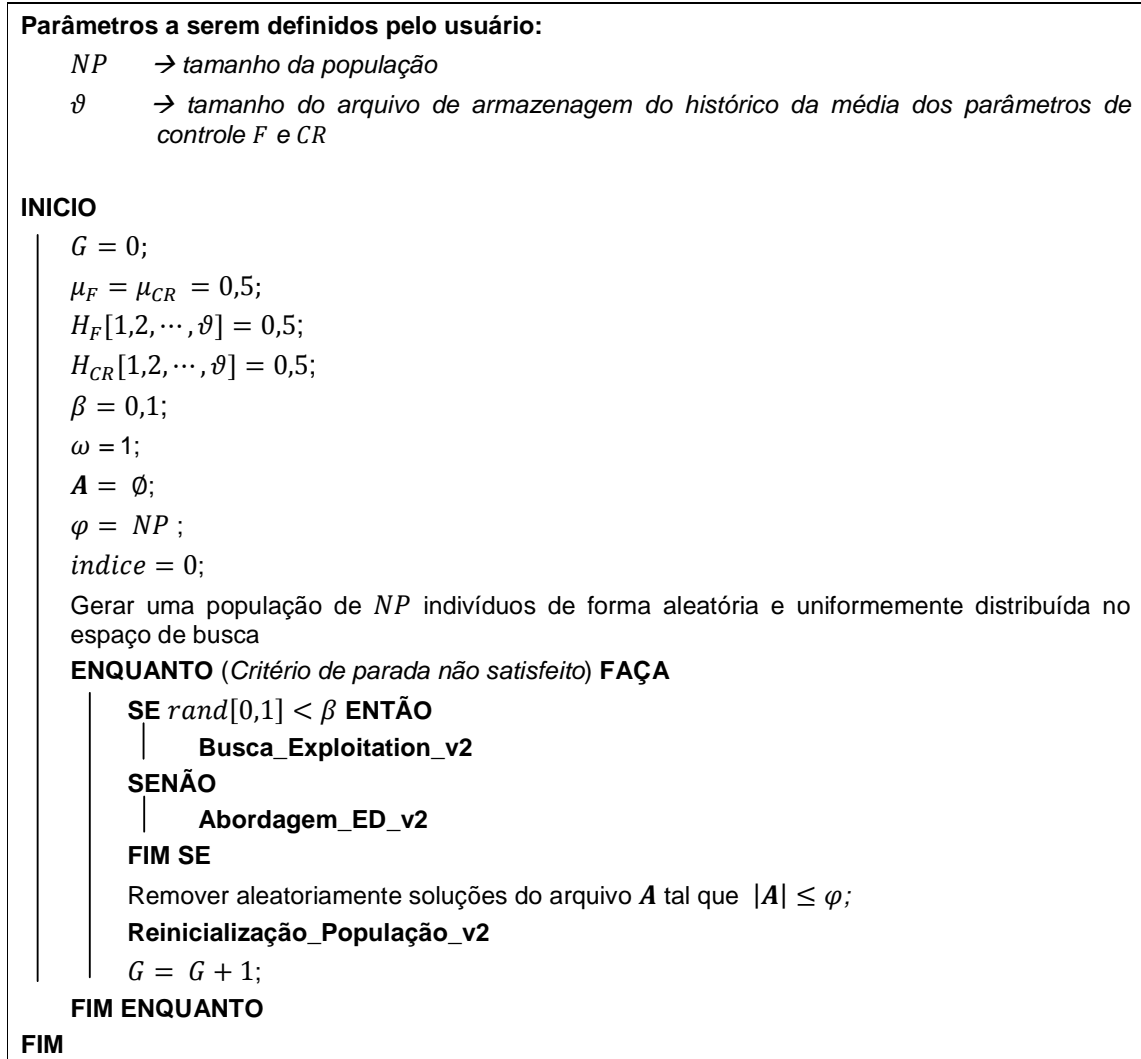


FIGURA 9: Pseudocódigo algoritmo HSADE-v2

FONTE: O autor (2014)

As principais diferenças do algoritmo HSADE-v2, comparativamente a primeira versão HSADE-v1, podem ser brevemente resumidas como:

- Número menor de parâmetros a serem ajustados pelo usuário. No algoritmo HSADE-v2, apenas dois parâmetros devem ser ajustados pelo usuário como pode ser observado na FIGURA 9;
- A geração da população inicial é realizada da maneira clássica, conforme descrito na seção 3.1.1;
- HSADE-v2 não utiliza nenhuma abordagem quase oposicional;
- Utilização de uma nova abordagem de evolução diferencial.

As principais etapas do algoritmo proposto HSADE-v2 serão descritas em maiores detalhes nas seções subsequentes.

### 5.2.1 Abordagem de Evolução Diferencial

Assim como na primeira versão, o algoritmo HSADE-v2 também utiliza uma abordagem de evolução diferencial para gerar uma população de vetores experimentais. Entretanto, na abordagem implementada no algoritmo HSADE-v2 apenas uma estratégia de geração de vetor experimental é utilizada. Além disto, esta abordagem emprega um novo método de auto adaptação dos parâmetros de controle.

A população de vetores experimentais é gerada através desta abordagem de evolução diferencial de acordo com uma probabilidade  $(1 - \beta)$ . O parâmetro  $\beta$  é auto-adaptado durante o processo de otimização. Este processo de adaptação será descrito em maiores detalhes na seção 5.2.2.

#### 5.2.1.1 Estratégia de geração do vetor experimental

Conforme já mencionado anteriormente, o algoritmo proposto HSADE-v2 utiliza apenas uma estratégia de geração de vetor experimental. A estratégia escolhida para ser implementada neste algoritmo baseia-se na estratégia *DE/current-to-pbest/1/bin* com arquivo externo definida na seção 4.1.

O vetor doador continua sendo gerado através das equações (21) e (22). Entretanto uma modificação foi realizada. Na variante *DE/current-to-pbest/1/bin*, o parâmetro  $p$  deve ser definido pelo usuário e é fixo durante todo o processo de otimização. Já no algoritmo HSADE-v2, o parâmetro  $p$  é variável durante todo o processo de otimização. Mais especificamente, no algoritmo proposto HSADE-v2 o parâmetro  $p$  é gerado de forma aleatória e uniforme no início de cada geração  $G$ , quando a abordagem de evolução diferencial é selecionada para gerar os vetores experimentais. Em outras palavras, o parâmetro  $p$  é calculado através da seguinte equação:

$$p = rand[0,1] \quad (60)$$

onde  $rand[0,1]$  correspondem ao resultado de um gerador de números aleatórios com distribuição uniforme no intervalo  $[0,1]$ .

De acordo com a equação (22), quando  $p = 0$  têm-se  $\lambda = 2$  e, conseqüentemente,  $\vec{X}_{best,G}^p$  será escolhido a partir do conjunto formado pelos dois melhores indivíduos da população atual. Por outro lado quando  $p = 1$  têm-se  $\lambda = NP$  e, portanto,  $\vec{X}_{best,G}^p$  será escolhido entre todos os indivíduos da população atual.

Mais precisamente, valores pequenos para o parâmetro  $p$  favorecem o *exploitation* pois  $\vec{X}_{best,G}^p$  poderá ser escolhido somente a partir de um grupo reduzido de indivíduos formado somente pelos melhores indivíduos e, conseqüentemente, os vetores experimentais serão atraídos para as melhores soluções correntes.

Por outro lado, valores altos para o parâmetro  $p$  favorecem o *exploration* pois  $\vec{X}_{best,G}^p$  poderá ser escolhido a partir de um número maior de indivíduos, gerando uma maior diversidade da população potencial de vetores experimentais.

Sendo assim, o objetivo de gerar valores aleatórios foi de melhorar o equilíbrio entre *exploration* e *exploitation* pois desta forma durante todo o processo de otimização, tanto valores menores quanto maiores podem ser selecionados para o parâmetro  $p$ .

#### 5.2.1.2 Adaptação dos parâmetros de controle $F$ e $CR$

O princípio do processo de adaptação do fator de mutação e taxa de cruzamento utilizado no algoritmo HSADE-v2 é o mesmo do processo de adaptação utilizado na primeira versão do algoritmo. Entretanto, o algoritmo proposto HSADE-v2 possui duas diferenças principais quando comparado com a primeira versão. A primeira diferença está relacionada ao fato de se utilizar apenas um conjunto de parâmetros  $\mu_{CR}$  e  $\mu_F$ , pois apenas uma estratégia de geração dos vetores experimental está disponível. A segunda diferença está relacionada ao fato de que no algoritmo HSADE-v2 uma nova metodologia de atualização dos parâmetros  $\mu_{CR}$  e  $\mu_F$  é implementada.

No algoritmo HSADE-v2, em cada geração  $G$  na qual a abordagem ED é aplicada, um fator de mutação  $F_i$  e uma taxa de cruzamento  $CR_i$  são gerados para cada vetor alvo  $\vec{X}_{i,G}$  da população atual de acordo com uma distribuição de Cauchy,

com parâmetro de locação  $\mu_F$  e parâmetro de escala de 0,1, e de acordo com uma distribuição normal, com média  $\mu_{CR}$  e desvio padrão 0,1, respectivamente.

$$F_i = \text{Cauchy}(\mu_F, 0,1) \quad (61)$$

$$CR_i = N(\mu_{CR}, 0,1) \quad (62)$$

$F_i$  é truncado para 1 caso  $F_i > 1$  ou recalculado de acordo com a equação (61) se  $F_i \leq 0$  enquanto  $CR_i$  é diretamente truncado para o intervalo [0,1] caso  $CR_i > 1$  ou  $CR_i < 0$ .

Os parâmetros  $\mu_{CR}$  e  $\mu_F$  são inicializados a 0,5 no início do processo de otimização e atualizados no final de cada geração, na qual a abordagem ED é aplicada, de acordo com as seguintes equações:

$$\mu_F = \text{média}_A(H_F) \quad (63)$$

$$\mu_{CR} = \text{média}_A(H_{CR}) \quad (64)$$

onde  $\text{média}_A(.)$  representa a média aritmética.  $H_F$  e  $H_{CR}$  são arquivos que armazenam o histórico da média dos parâmetros de controle  $F$  e  $CR$ , respectivamente, utilizados nas últimas  $\vartheta$  gerações na qual a abordagem ED foi aplicada.

Os arquivos  $H_F$  e  $H_{CR}$  possuem tamanho  $\vartheta$  e são responsáveis por armazenar os valores médios dos parâmetros de controle bem sucedidos de cada geração. O processo de criação e atualização destes arquivos é realizado da seguinte forma.

Inicialmente todos os elementos de  $H_F$  e  $H_{CR}$  são inicializados a 0,5. Na sequência, a cada geração na qual a abordagem ED é aplicada, os valores médios dos parâmetros de controle bem sucedidos de cada geração, denominados  $m_{S_F}$  e  $m_{S_{CR}}$ , são adicionados aos seus respectivos arquivos no lugar dos elementos mais antigos.

As médias  $m_{S_F}$  e  $m_{S_{CR}}$  armazenadas, são calculadas após o processo de seleção, de acordo com as seguintes equações:

$$m_{S_F} = \text{média}_{Gen}(S_F, 3) \quad (65)$$

$$m_{S_{CR}} = média_{Gen}(S_{CR}, 2) \quad (66)$$

onde  $S_F$  é o conjunto de todos os fatores de mutação bem sucedidos na geração  $G$  e  $S_{CR}$  é o conjunto de todas as taxas de cruzamento bem sucedidas na geração  $G$ .  $média_{Gen}(\cdot, \alpha)$  corresponde a média generalizada com parâmetro  $\alpha$  que é definida por :

$$média_{Gen}(B, \alpha) = \sqrt[\alpha]{\frac{1}{n} \cdot \sum_{b \in B} b^\alpha}, \quad \alpha \neq 0 \quad (67)$$

onde  $n$  corresponde a cardinalidade do conjunto  $B$ .

O objetivo do processo de adaptação dos parâmetros de controle implementado no algoritmo HSADE-v2 é de promover gradualmente a adaptação dos parâmetros de controle  $F$  e  $CR$  para os valores mais adequados para o problema a ser resolvido e para diferentes fases do processo de otimização.

Assim como na primeira versão no algoritmo HSADE-v2, os parâmetros  $F_i$  e  $CR_i$  são gerados a partir de uma distribuição de Cauchy e a partir de uma distribuição normal respectivamente.

Conforme já mencionado na seção 5.1.2.2, o motivo de se utilizar a distribuição de Cauchy na geração do fator de mutação é devido ao fato desta distribuição ter a propriedade de calda longa o que reflete em uma maior diversidade dos fatores de mutação gerados. Isto é útil principalmente quando o ponto de ótimo global se encontra longe da região de busca corrente, pois fatores de mutação podem ser gerados de tal forma que gerem perturbação suficiente tal que o risco de uma convergência prematura seja evitado.

No caso do processo de adaptação do fator de cruzamento, uma distribuição normal é utilizada pois a propriedade de cauda longa da distribuição de Cauchy não é necessária.

Com relação aos parâmetros  $\mu_{CR}$  e  $\mu_F$ , o algoritmo HSADE-v2 implementa um novo processo de adaptação destes parâmetros, diferentemente daquele descrito na seção 5.1.2.2.

O primeiro ponto a ser observado relativo a este processo está relacionado às equações (63) e (64). A atualização dos parâmetros  $\mu_{CR}$  e  $\mu_F$  nada mais é do que a aplicação de uma média móvel simples calculada sobre  $\vartheta$  observações.

O objetivo deste cálculo é a realização de uma filtragem sobre os valores de  $m_{SF}$  e  $m_{SCR}$  devido às interferências estatísticas que podem ocorrer a cada cálculo de  $m_{SF}$  e  $m_{SCR}$ . Isto faz com que haja, conseqüentemente, uma melhor determinação da tendência do fator de mutação e taxa de cruzamento durante todo o processo de otimização.

Outro ponto a ser observado em relação a este processo de adaptação é a utilização da média generalizada no cálculo dos valores médios dos parâmetros de controle bem sucedidos de cada geração. Como pode ser observado nas equações (65) e (66), o cálculo dos parâmetros  $m_{SF}$  e  $m_{SCR}$  utilizam a média generalizada com parâmetro  $\alpha = 2$  e  $\alpha = 3$  respectivamente ao invés da média aritmética.

Para se ter uma idéia de como o parâmetro  $\alpha$  impacta o cálculo da média generalizada, a FIGURA 10 apresenta o cálculo da média generalizada em função do parâmetro  $\alpha$  para um conjunto exemplo  $B = \{1,2,3,4,5,6,7,8,9,10\}$ . O ponto identificado pela letra **A** representa o valor da média aritmética. A partir desta figura pode-se concluir que  $média_A(\cdot) = média_{Gen}(\cdot, 1) \leq média_{Gen}(\cdot, 2) \leq média_{Gen}(\cdot, 3)$ .

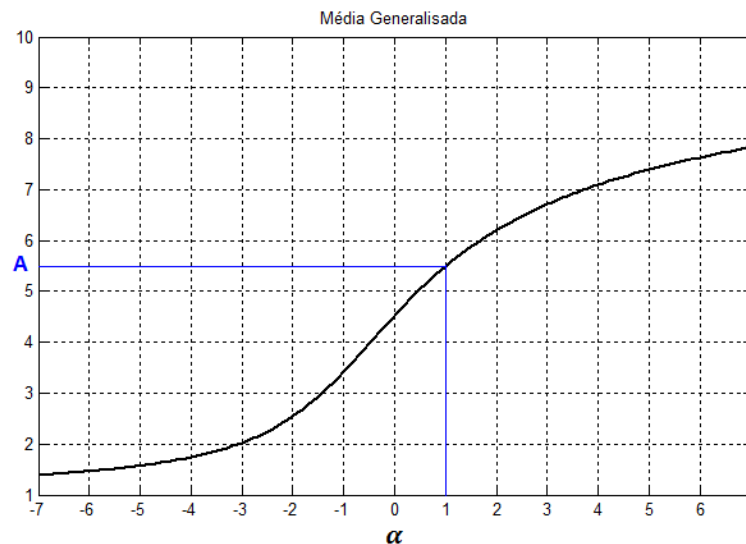


FIGURA 10: Exemplo de cálculo da média generalizada em função do parâmetro  $\alpha$  para o conjunto  $B = \{1,2,3,4,5,6,7,8,9,10\}$ .

FONTE: O autor (2014)

Com relação ao processo de adaptação do fator de mutação, a média generalizada com parâmetro  $\alpha = 3$  é útil na propagação de maiores valores do fator de mutação, auxiliando, portanto, em uma maior perturbação dos vetores experimentais, o que reduz o risco de se ficar preso em um mínimo local.

Já com relação ao processo de adaptação da taxa de cruzamento, sabe-se que o cálculo de  $\mu_{CR}$  através do processo definido na seção 4.1 tem uma tendência implícita para valores pequenos durante o processo de auto-adaptação (PENG et al., 2009). Sendo assim, para diminuir esta tendência, a média generalizada com parâmetro  $\alpha = 2$  é utilizada para que haja uma propagação de maiores valores da taxa de cruzamento, mas não tanto quanto a do fator de mutação.

O pseudocódigo da abordagem ED implementada no algoritmo HSADE-v2 é apresentado na FIGURA 11.

<p><b>INICIO</b></p> <p><math>S_F = S_{CR} = \emptyset;</math>  <math>p = rand[0,1];</math>  <math>\lambda = max(\lfloor p \cdot NP \rfloor, 2);</math></p> <p><b>PARA</b> <math>i = 1</math> <b>ATÉ</b> <math>NP</math> <b>FAÇA</b></p> <p>    Gerar <math>F_i = Cauchy(\mu_F, 0,1)</math> e <math>CR_i = N(\mu_{CR}, 0,1);</math>      Escolher aleatoriamente um indivíduo <math>\vec{X}_{best,G}^p</math> a partir dos <math>\lambda</math> melhores indivíduos da população atual <math>\mathbf{P}</math>;      Escolher aleatoriamente <math>\vec{X}_{r_1,G}</math> a partir da população atual <math>\mathbf{P}</math> tal que <math>\vec{X}_{r_1,G} \neq \vec{X}_{i,G}</math>;      Escolher aleatoriamente <math>\vec{X}_{r_2,G}</math> a partir da união <math>\mathbf{P} \cup \mathbf{A}</math>, tal que <math>\vec{X}_{r_2,G} \neq \vec{X}_{r_1,G} \neq \vec{X}_{i,G}</math>      <math>\vec{V}_{i,G} = \vec{X}_{i,G} + F_i \cdot (\vec{X}_{best,G}^p - \vec{X}_{i,G}) + F_i \cdot (\vec{X}_{r_1,G} - \vec{X}_{r_2,G});</math>      Gerar <math>\vec{U}_{i,G}</math> aplicando o operador de cruzamento binomial;      Caso alguma componente de <math>\vec{U}_{i,G}</math> viole as restrições laterais, aplicar o método <i>bounce back</i>;      Avaliar <math>f(\vec{U}_{i,G});</math>      <b>SE</b> <math>f(\vec{X}_{i,G}) \leq f(\vec{U}_{i,G})</math> <b>ENTÃO</b>        <math>\vec{X}_{i,G+1} = \vec{X}_{i,G};</math>      <b>SENÃO</b>        <math>\vec{X}_{i,G+1} = \vec{U}_{i,G};</math>        <math>\vec{X}_{i,G} \rightarrow \mathbf{A};</math>        <math>F_i \rightarrow S_F;</math>        <math>CR_i \rightarrow S_{CR};</math>      <b>FIM SE</b></p> <p><b>FIM PARA</b></p> <p><math>m_{S_F} = média_{Gen}(S_F, 3);</math></p>
---

```

 $m_{SCR} = média_{Gen}(S_{CR}, 2);$ 
 $indice = indice + 1;$ 
SE  $indice > \vartheta$  ENTÃO
    |  $indice = 1;$ 
FIM SE
 $H_F[indice] = m_{SF};$ 
 $H_{CR}[indice] = m_{SCR};$ 
 $\mu_F = media_A(H_F);$ 
 $\mu_{CR} = media_A(H_{CR});$ 
FIM

```

FIGURA 11: Pseudocódigo Abordagem\_ED\_v2

FONTE: O autor (2014)

### 5.2.2 Busca voltada à *Exploitation*

Esta segunda versão do algoritmo *HSADE* também utiliza uma busca voltada à *exploitation*. Da mesma forma que na primeira versão, o objetivo desta busca é aumentar a velocidade de convergência do processo de otimização através de um aprimoramento da capacidade de *exploitation*.

A busca voltada à *exploitation* implementada no algoritmo *HSADE-v2* utiliza o mesmo operador de busca implementado na primeira versão do algoritmo e definido pelas equações (49), (50), (51) e (52) para a geração do vetor experimental  $\vec{U}_{i,G}$ . Esta busca voltada à *exploitation* também é aplicada a cada geração  $G$  de acordo com uma probabilidade  $\beta$ .

A principal diferença está no fato de que a probabilidade  $\beta$  da busca voltada à *exploitation* é adaptada durante o processo de otimização, enquanto na primeira versão este parâmetro é fixo devendo ser definido pelo usuário. Além disto, o fator de ajuste  $\omega$  da equação (50) também é adaptado de uma forma diferente durante o processo de otimização.

A probabilidade  $\beta$  é inicializada a 0,1 e em seguida este parâmetro é aumentado ou reduzido de acordo com o resultado da busca voltada à *exploitation*. A probabilidade  $\beta$  é aumentada toda vez que a busca voltada à *exploitation* produzir ao menos um vetor experimental bem sucedido, ou seja, quando pelo menos um vetor experimental for melhor que seu respectivo vetor alvo. Este aumento é feito através da multiplicação da probabilidade  $\beta$  por 1,05. Por outro lado, a probabilidade  $\beta$  é reduzida cada vez que a busca voltada à *exploitation* não produzir nenhum vetor



experimental bem sucedido, ou seja, quando todos os vetores experimentais gerados por esta busca são descartados durante a seleção. A redução é feita através da multiplicação do fator  $\beta$  por 0,75.

A probabilidade  $\beta$  é limitada a um valor mínimo de 0,1 e a um valor máximo de 0,9. Estes limites são valores arbitrários para assegurar que tanto a busca voltada à *exploitation*, quanto a abordagem ED tenham uma probabilidade não nula de serem escolhidas durante todo o processo de otimização.

Os valores de redução e aumento são valores arbitrários cuja finalidade é de fazer com que a probabilidade  $\beta$  seja aumentada gradativa e lentamente e que seja reduzida rapidamente. Isto é necessário para dar maior estabilidade ao processo de adaptação e melhorar a velocidade de convergência pois esta adaptação possui uma tendência implícita para valores maiores o que aumentaria a probabilidade da busca voltada à *exploitation* e, conseqüentemente, aumentaria a probabilidade de estagnação do algoritmo visto que a busca voltada à *exploitation* não é capaz de escapar de mínimos locais.

A adaptação do fator de ajuste  $\omega$  é feita da mesma forma que a adaptação da probabilidade  $\beta$ . Sendo assim,  $\omega$  é inicializado a 1 e é aumentado toda vez que busca voltada à *exploitation* produzir ao menos um vetor experimental bem sucedido e reduzido cada vez que esta busca não produzir nenhum vetor experimental bem sucedido. O aumento é feito pela multiplicação do fator  $\omega$  por 1,05 e a redução é feita através da multiplicação do fator  $\omega$  por 0,75. Além disto, o fator de ajuste  $\omega$  é limitado a um valor mínimo de 0,0001 e a um valor máximo de 1.

O objetivo da redução do fator de ajuste  $\omega$  é de fazer com que a busca voltada à *exploitation* explore o interior de uma vizinhança menor toda vez que esta busca não é bem sucedida. Entretanto, uma vez que a busca voltada à *exploitation* é bem sucedida, aumenta-se gradativamente e lentamente a vizinhança de busca.

A adaptação de ambos os parâmetros  $\beta$  e  $\omega$  é realizada da mesma forma. Basicamente, os parâmetros  $\beta$  e  $\omega$  são reduzidos até o momento em que a busca voltada à *exploitation* é capaz de produzir vetores experimentais bem sucedidos. Ou seja, enquanto a busca voltada à *exploitation* não for bem sucedida, reduz-se o fator de ajuste  $\omega$  para que esta busca explore o interior de uma vizinhança menor e, ao mesmo tempo, reduz-se a probabilidade  $\beta$  de forma a aumentar a quantidade de

vetores experimentais produzidos pela abordagem ED para não prejudicar a convergência do algoritmo.

Por outro lado, uma vez que a busca voltada à *exploitation* é bem sucedida, os parâmetros  $\beta$  e  $\omega$  são aumentados gradativamente. Ou seja, enquanto a busca voltada à *exploitation* for bem sucedida, aumenta-se o fator de ajuste  $\omega$  para que esta busca explore o interior de uma vizinhança cada vez maior de forma a aumentar a velocidade de convergência do algoritmo. Ao mesmo tempo, aumenta-se a probabilidade  $\beta$ , aumentando-se, conseqüentemente, a quantidade de vetores experimentais produzidos pela busca voltada à *exploitation* também com o intuito de aumentar a velocidade de convergência do algoritmo.

O pseudocódigo do operador de busca voltada à *exploitation* utilizado para gerar os vetores experimentais é ilustrado na FIGURA 12.

```

INICIO
  PARA  $j = 1$  ATÉ  $D$  FAÇA
     $MIN_j^G = \min(X_{j,1,G}, X_{j,2,G}, \dots, X_{j,NP,G});$ 
     $MAX_j^G = \max(X_{j,1,G}, X_{j,2,G}, \dots, X_{j,NP,G});$ 
  FIM PARA
  PARA  $i = 1$  ATÉ  $NP$  FAÇA
    Selecionar aleatoriamente  $r_0 \in \{1, 2, \dots, NP\}$  tal que  $r_0 \neq i$ ;
    PARA  $j = 1$  ATÉ  $D$  FAÇA
       $\Delta X_j = \omega \cdot \text{rand}[0,1] \cdot (MAX_j^G - MIN_j^G);$ 
       $U_{j,i,G} = X_{j,r_0,G} - \Delta X_j + 2 \cdot \text{rand}[0,1] \cdot \Delta X_j;$ 
    FIM PARA
    Caso alguma componente de  $\vec{U}_{i,G}$  viole as restrições laterais, aplicar o método bounce back;
    Avaliar  $f(\vec{U}_{i,G});$ 
    SE  $f(\vec{X}_{i,G}) \leq f(\vec{U}_{i,G})$  ENTÃO
       $\vec{X}_{i,G+1} = \vec{X}_{i,G};$ 
    SENÃO
       $\vec{X}_{i,G+1} = \vec{U}_{i,G};$ 
       $\vec{X}_{i,G} \rightarrow \mathbf{A};$ 
    FIM SE
  FIM PARA
  SE nenhum vetor experimental for melhor que seu vetor alvo correspondente ENTÃO
     $\omega = \max(\omega \cdot 0,75, 0,0001);$ 
     $\beta = \max(\beta \cdot 0,75, 0,1);$ 
  SENÃO
     $\omega = \min(\omega \cdot 1,05, 1);$ 

```

```

| | |  $\beta = \min(\beta \cdot 1,05, 0,9);$ 
| | | FIM SE
| | FIM

```

FIGURA 12: Pseudocódigo Busca\_Exploitation\_v2

FONTE: O autor (2014)

### 5.2.3 Reinicialização da população

Assim como na primeira versão, o algoritmo HSADE-v2 aplica uma reinicialização da população caso uma estagnação seja detectada. A detecção de estagnação e a reinicialização da população é realizada da mesma forma que é realizada na primeira versão conforme descrito na seção 5.1.5.

Além da reinicialização da população, os parâmetros  $\mu_F$  e  $\mu_{CR}$  são reinicializados a 0,5, sendo isto feito para que estes parâmetros possam se adaptar à nova população.

O pseudocódigo da reinicialização da população em caso de estagnação é apresentado na FIGURA 13.

```

INICIO
| SE todos os indivíduos da população possuem o mesmo valor de função objetivo ENTÃO
| | Selecionar aleatoriamente  $r_0 \in \{1,2, \dots NP\};$ 
| |  $\vec{X}_{1,G+1} = \vec{X}_{r_0,G+1};$ 
| | PARA  $i = 2$  ATÉ  $NP$  FAÇA
| | | PARA  $j = 1$  ATÉ  $D$  FAÇA
| | | |  $\sigma_j = (X_j^{max} - X_j^{min})/4;$ 
| | | |  $X_{j,i,G+1} = X_{j,r_0,G+1} + N(0, \sigma_j);$ 
| | | FIM PARA
| | FIM PARA
| |  $\mu_F = \mu_{CR} = 0,5;$ 
| FIM SE
FIM

```

FIGURA 13: Pseudocódigo Reinicialização\_População\_v2

FONTE: O autor (2014)

## 5.3 RESUMO COMPARATIVO

Para um melhor entendimento das diferenças entre as duas versões do algoritmo HSADE apresentadas nesta dissertação, a TABELA 1 apresenta um resumo comparativo das principais diferenças entre os algoritmos HSADE-v1 e HSADE-v2

TABELA 1 – RESUMO COMPARATIVO ENTRE AS DUAS VERSÕES DO ALGORITMO HSADE

(continua)

	HSADE-v1	HSADE-v2
<i>Geração da população inicial</i>	Geração de indivíduos através de uma inicialização quase oposicional	Geração de indivíduos aleatoriamente e uniformemente distribuídos no espaço de busca
<i>Geração dos vetores experimentais</i>	Vetores experimentais podem ser gerados através de uma abordagem de evolução diferencial ou através de um operador de busca voltado à <i>exploitation</i> de acordo com uma probabilidade $(1-\beta)$ e $\beta$ respectivamente	
	Parâmetro $\beta$ deve ser definido pelo usuário	Parâmetro $\beta$ auto-adaptado durante o processo de otimização de acordo com o resultado da busca voltada à <i>exploitation</i>
<i>Busca voltada à exploitation</i>	Geração de vetores experimentais de forma aleatória na vizinhança da população atual	
	Fator de ajuste $\omega$ reduzido toda vez que a busca voltada à <i>exploitation</i> não for bem sucedida	Fator de ajuste $\omega$ é reduzido ou aumentado de acordo com o resultado da busca voltada à <i>exploitation</i>
<i>Abordagem ED:</i>  <i>Estratégias de geração do vetor experimental</i>	Utilização de duas estratégias de geração de vetores experimentais: - <i>DE/rand/2/bin</i> - <i>DE/target-to-pbest/1/bin</i>	Utilização de apenas uma estratégia de geração de vetores experimentais: - <i>DE/target-to-pbest/1/bin</i>
	Parâmetro $p$ que define a “ganância” ( <i>greediness</i> ) do operador de mutação <i>DE/target-to-pbest</i> deve ser definido pelo usuário	Parâmetro $p$ que define a “ganância” ( <i>greediness</i> ) do operador de mutação <i>DE/target-to-pbest</i> gerado aleatoriamente no início de cada geração

TABELA 1 – RESUMO COMPARATIVO ENTRE AS DUAS VERSÕES DO ALGORITMO HSADE

(continuação)

	<p>Fator de mutação <math>F_i</math> gerado de acordo com uma distribuição de cauchy e de acordo com a estratégia de geração do vetor experimental:</p> <p>Se estratégia <math>m = 1</math> é usada para gerar o <math>i</math>-ésimo vetor experimental, então:</p> $F_i = \text{Cauchy}(\mu_{F,1}, 0,1)$ <p>Se estratégia <math>m = 2</math> é usada para gerar o <math>i</math>-ésimo vetor experimental, então:</p> $F_i = \text{Cauchy}(\mu_{F,2}, 0,1)$	<p>Fator de mutação <math>F_i</math> gerado de acordo com uma distribuição de cauchy:</p> $F_i = \text{Cauchy}(\mu_F, 0,1)$
<p>Abordagem ED:</p> <p>Geração do Fator de mutação</p>	<p>Parâmetro <math>\mu_{F,m}</math>, <math>m = 1,2</math> atualizado no final de cada geração através da seguinte fórmula:</p> $\mu_{F,m} = (1 - c) \cdot \mu_{F,m} + c \cdot \text{média}_L(S_{F,m})$ <p>onde <math>S_{F,m}</math> é o conjunto de todos os fatores de mutação bem sucedidos associados à estratégia <math>m</math> na geração <math>G</math></p>	<p>Parâmetro <math>\mu_F</math> atualizado no final de cada geração através da seguinte fórmula:</p> $\mu_F = \text{média}_A(H_F)$ <p>onde <math>H_F</math> corresponde a um arquivo responsável por armazenar o histórico da média dos fatores de mutação bem sucedidos, <math>m_{S_F}</math>, utilizado nas últimas <math>\vartheta</math> gerações onde :</p> $m_{S_F} = \text{média}_{Gen}(S_F, 3)$ <p>sendo <math>S_F</math> o conjunto de todos os fatores de mutação bem sucedidos na geração <math>G</math></p>

TABELA 1 – RESUMO COMPARATIVO ENTRE AS DUAS VERSÕES DO ALGORITMO HSADE

(conclusão)

	<p>Taxa de cruzamento <math>CR_i</math> gerada de acordo com uma distribuição normal e de acordo com a estratégia de geração do vetor experimental:</p> <p>Se estratégia <math>m = 1</math> é usada para gerar o <math>i</math>-ésimo vetor experimental, então:</p> $CR_i = N(\mu_{CR,1}, 0,1)$ <p>Se estratégia <math>m = 2</math> é usada para gerar o <math>i</math>-ésimo vetor experimental, então:</p> $CR_i = N(\mu_{CR,2}, 0,1)$	<p>Taxa de cruzamento <math>CR_i</math> gerada de acordo com uma distribuição normal:</p> $CR_i = N(\mu_{CR}, 0,1)$
<p>Abordagem ED</p> <p>Geração da taxa de cruzamento</p>	<p>Parâmetro <math>\mu_{CR,m}</math>, <math>m = 1,2</math> atualizado no final de cada geração através da seguinte fórmula:</p> $\mu_{CR,m} = (1 - c) \cdot \mu_{CR,m} + c \cdot média_A(S_{CR,m})$ <p><math>S_{CR,m}</math> é o conjunto de todas as taxas de cruzamento bem sucedidas associadas à estratégia <math>m</math> na geração <math>G</math></p>	<p>Parâmetro <math>\mu_{CR}</math> atualizado no final de cada geração através da seguinte fórmula:</p> $\mu_{CR} = média_A(H_{CR})$ <p>onde <math>H_{CR}</math> corresponde a um arquivo responsável por armazenar o histórico da média da taxa de cruzamento, <math>m_{SCR}</math>, utilizado nas últimas <math>\vartheta</math> gerações onde:</p> $m_{SCR} = média_{Gen}(S_{CR}, 2)$ <p>sendo <math>S_{CR}</math> o conjunto de todas as taxas de cruzamento bem sucedidas na geração <math>G</math></p>
Seleção	Seleção realizada da mesma forma que o algoritmo clássico ED	
Abordagem quase oposicional	Após o operador seleção, uma abordagem quase oposicional é aplicada com o objetivo de melhorar a velocidade de convergência do processo de otimização	--
Reinicialização da população	Se uma estagnação da população é detectada, a população é reinicializada na vizinhança de um indivíduo da população atual escolhido aleatoriamente e uniformemente	

FONTE: O autor (2014)

NOTA:  $média_{Gen}(\cdot, \alpha)$  representa a média generalizada com parâmetro  $\alpha$ ,  $média_A(\cdot)$  representa a média aritmética e  $média_L(\cdot)$  representa a média Lehmer.

## 6 AVALIAÇÃO DO DESEMPENHO DO ALGORITMO PROPOSTO

Para avaliar o desempenho e eficiência da abordagem apresentada nesta dissertação, cada uma das versões do algoritmo *HSADE* será comparada a oito algoritmos estado da arte baseados na evolução diferencial: JADE com e sem arquivo (ZHANG; SANDERSON, 2009a), jDE (BREST et al., 2006), CoDE (WANG, CAI; ZHANG, 2011), EPSDE (MALLIPEDDI et al., 2011), SaDE (QIN; HUANG; SUGANTHAN, 2009), DEGL (DAS et al., 2009) e DEahcSPX (NOMAN; IBA, 2008). A seção 4 apresenta uma breve descrição destes oito algoritmos.

Cada uma das duas versões do algoritmo *HSADE*, juntamente com os oito algoritmos acima citados, serão aplicados a três estudos de caso. Para cada um dos estudos de caso, será realizada uma avaliação do desempenho dos algoritmos propostos neste trabalho através da comparação dos resultados obtidos por cada um dos algoritmos *HSADE-v1* e *HSADE-v2* com os resultados obtidos pelos algoritmos de comparação JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX.

O primeiro estudo de caso abordado nesta dissertação corresponde a um conjunto de funções testes proposto no *Special Session & Competition on Real-Parameter Optimization* do *IEEE Congress on Evolutionary Computation* de 2005 (CEC2005).

O segundo estudo de caso corresponde a um conjunto de problemas proposto no *Competition on Testing Evolutionary Algorithms on Real-world Numerical Optimization Problems* do *IEEE Congress on Evolutionary Computation* de 2011 (CEC2011). Este estudo de caso diferencia-se dos demais pelo fato de ser formado de problemas reais de otimização.

O terceiro estudo de caso corresponde a um conjunto de funções testes proposto no *Special Session & Competition on Real-Parameter Single Objective Optimization* do *IEEE Congress on Evolutionary Computation* de 2013 (CEC2013).

Estes estudos de caso foram selecionados, pois cada um deles é formado por um conjunto de funções que correspondem a problemas de minimização mono-objetivos sujeito apenas a restrições laterais, especificamente propostas para avaliar o desempenho de algoritmos evolutivos.

## 6.1 GERAÇÃO DOS DADOS PARA ANÁLISE

Os algoritmos evolutivos baseados na evolução diferencial são métodos de otimização estocásticos. Sendo assim, visando reduzir o impacto da natureza estocástica dos algoritmos durante a avaliação de suas performances, é necessário que cada algoritmo seja executado várias vezes para uma mesma função.

Consequentemente, cada um dos dez algoritmos será executado um número definido de vezes para cada função dos estudos de caso. Cada uma destas execuções será denominada *experimento*. O número de experimentos para cada função será definido de acordo com o estudo de caso.

Além disso, será utilizado um critério único de parada dos experimentos para todos os algoritmos. Este critério de parada é baseado em um número máximo de avaliações da função objetivo e será definido especificamente para cada estudo de caso.

Para os estudos de caso referentes ao CEC2005 e CEC2013, o desempenho será avaliado de acordo com o valor do erro da função objetivo em cada experimento que é definido como  $f(\vec{X}_{Melhor}) - f(\vec{X}^*)$ , onde  $\vec{X}_{Melhor}$  representa a melhor solução encontrada pelo algoritmo de otimização em um experimento e  $\vec{X}^*$  é o ponto de mínimo global da função de referência a ser otimizada. Para cada algoritmo e para cada função, o valor do erro da função objetivo foi gravado ao final de cada experimento.

Para o estudo de caso referente ao CEC2011, o desempenho será avaliado diretamente de acordo com o valor da função objetivo obtido em cada experimento que é definido como  $f(\vec{X}_{Melhor})$  onde  $\vec{X}_{Melhor}$  representa a melhor solução encontrada pelo algoritmo de otimização em um experimento. Para cada algoritmo e para cada função, o valor da função objetivo foi gravado ao final de cada experimento.

Além disso, para uma comparação mais justa, a mesma inicialização foi utilizada para todos os algoritmos. Isto foi alcançado através da utilização de um mesmo *seed* para o gerador de números aleatórios para todos os algoritmos. Entretanto, é importante salientar que para cada experimento um *seed* diferente foi utilizado.



Com relação à codificação dos algoritmos, tanto as duas versões do algoritmo *HSADE*, como os oito algoritmos de comparação, foram codificados em *MATLAB*. O código fonte dos algoritmos *SaDE*, *jDE*, *JADE* com e sem arquivo, *EPSDE* e *CoDE* foram obtidos em (<http://dces.essex.ac.uk/staff/zhang/code/codealgorithm/>). Já o algoritmo *DEGL* foi codificado baseando-se em Das et al. (2009) e *DEahcSPX* foi codificado baseando-se em Noman e Iba (2008).

## 6.2 CRITÉRIO DE DESEMPENHO

Para ser possível analisar os resultados obtidos, e conseqüentemente, determinar se a abordagem proposta neste trabalho é melhor ou não do que os outros algoritmos de uma forma coerente, devido à aleatoriedade presente nos algoritmos testados, testes não paramétricos serão utilizados. Testes não paramétricos são inferências estatísticas úteis na análise de dados dos quais não se conhece a forma de distribuição, o que é o caso dos resultados obtidos nos estudos de caso considerado nesta dissertação

Os testes não paramétricos são utilizados para testar hipóteses estatísticas que são afirmações, podendo ou não ser verdadeiras, acerca de um ou mais conjuntos de dados (WALPOLE et al, 2011; MONTGOMERY, RUNGER, 2010).

Em um teste de hipóteses, duas hipóteses relativas aos conjuntos de dados a serem analisados são definidas: a *hipótese nula*, representada por  $H_0$  e a *hipótese alternativa* representada por  $H_1$ .

O procedimento de análise destas hipóteses é realizado da seguinte maneira. Inicialmente a hipótese nula é considerada verdadeira. O teste não paramétrico é então aplicado ao conjunto de dados para decidir sobre rejeitar ou não a hipótese nula. Caso a hipótese nula  $H_0$  não seja rejeitada pelo teste, considera-se ela como verdadeira. Em caso de rejeição da hipótese nula, considera-se hipótese alternativa  $H_1$  como verdadeira (WALPOLE et al, 2011; MONTGOMERY, RUNGER, 2010).

A decisão sobre rejeitar ou não uma hipótese está sujeita a erros, pois se está tomando a decisão em condições de incerteza. Sendo assim, para tomar uma decisão é utilizado um nível de significância  $\alpha$  que corresponde à probabilidade de

se rejeitar a hipótese quando ela é verdadeira, ou seja, a probabilidade de se cometer um erro. Este erro é também conhecido como erro do *Tipo I*.

A tomada de decisão para rejeitar a hipótese nula será baseada no método do *valor-p*. O *valor-p* é uma estatística obtida através do teste não paramétrico aplicado aos dados em análise. O *valor-p* pode ser definido como a probabilidade de se obter uma estatística de teste igual ou mais extrema que aquela observada em uma amostra, dado que a hipótese nula é verdadeira. Basicamente, quanto menor o *valor-p*, maior é a evidência para rejeitar a hipótese nula (WALPOLE et al, 2011; MONTGOMERY, RUNGER, 2010).

No método do *valor-p*, a hipótese nula será rejeitada se o *valor-p* é menor ou igual ao nível de significância  $\alpha$ .

Para analisar os resultados obtidos em cada um dos estudos de caso, dois testes não paramétricos serão utilizados: *teste de Mann-Whitney* e o *teste post hoc de Friedman* (DERRAC et al., 2011).

#### 6.2.1 Teste de *Mann-Whitney*

O teste de *Mann-Whitney* é um teste não paramétrico utilizado para comparar dois grupos de dados não pareados e independentes. Uma hipótese sobre estes dados é formulada e o teste de *Mann-Whitney* é então aplicado para rejeitar ou não esta hipótese.

A partir destas considerações, o teste de *Mann-Whitney* será utilizado para realizar uma análise do desempenho entre dois algoritmos em cada função teste. Mais especificamente, este teste será utilizado para comparar o desempenho de cada um dos algoritmos HSADE-v1 e HSADE-v2 com cada um dos oito algoritmos (JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX) em cada uma das funções testes dos estudos de caso.

Sendo assim, para cada função teste, o conjunto de dados provenientes de cada um dos algoritmos HSADE-v1 e HSADE-v2, o qual será denominado *algoritmo de controle*, será comparado individualmente com cada conjunto de dados proveniente dos algoritmos restantes, os quais serão chamados de *algoritmos de comparação*.

Para os estudos de casos referentes ao CEC2005 e CEC3013, o conjunto de dados de cada algoritmo utilizado no teste de *Mann-Whitney* será formado pelo

respectivo conjunto de valores de erro da função objetivo de todos os experimentos. Já para o estudo de caso referente ao CEC2011, o conjunto de dados de cada algoritmo utilizado no teste de *Mann-Whitney* será formado diretamente pelo respectivo conjunto de valores da função objetivo de todos os experimentos

Para cada função será inicialmente realizado um teste bilateral, ou seja, as seguintes hipóteses serão verificadas:  $H_0: A = B$  e  $H_1: A \neq B$ . Neste teste, será verificado se o conjunto de valores de erro da função objetivo provenientes do algoritmo de controle  $A$  possui a mesma distribuição que o conjunto de valores de erro da função objetivo provenientes do algoritmo de comparação  $B$ . Caso a hipótese nula não seja rejeitada considera-se que os algoritmos  $A$  e  $B$  possuem desempenho similar no processo de otimização da função em análise.

Caso a hipótese nula seja rejeitada, considera-se que os algoritmos  $A$  e  $B$  possuem desempenho diferentes. Entretanto, não se pode afirmar qual deles possui o melhor desempenho. Sendo assim, dois testes adicionais serão aplicados caso seja identificado que os algoritmos que estão sendo comparados possuem desempenhos diferentes.

Primeiramente será realizado um teste unilateral à esquerda:  $H_0: A = B$  e  $H_1: A < B$ . Neste teste será verificado se o conjunto de valores de erro da função objetivo provenientes do algoritmo de controle  $A$  está deslocado para a esquerda em relação à distribuição do conjunto de valores de erro da função objetivo provenientes do algoritmo de comparação  $B$ . Caso a hipótese nula seja rejeitada, considera-se que o algoritmo de controle  $A$  possui desempenho melhor que o algoritmo de comparação  $B$ .

Caso a hipótese nula do teste unilateral à esquerda não seja rejeitada, será então realizado um teste unilateral à direita:  $H_0: A = B$  e  $H_1: A > B$ . Neste teste será verificado se o conjunto de valores de erro da função objetivo provenientes do algoritmo de controle  $A$  está deslocado para a direita em relação à distribuição do conjunto de valores de erro da função objetivo provenientes do algoritmo de comparação  $B$ . Caso a hipótese nula seja rejeitada, considera-se que o algoritmo de comparação  $B$  possui desempenho melhor que o algoritmo de controle  $A$ .

O nível de significância utilizado para rejeitar as hipóteses nulas no teste de *Mann-Whitney* foi de  $\alpha = 0,05$ .

### 6.2.2 Teste *post hoc* de *Friedman*

O teste de *Mann-Whitney* permite realizar apenas uma análise comparativa entre dois algoritmos. Entretanto, ele não permite, com um determinado grau de certeza, extrair uma conclusão de uma comparação envolvendo mais de dois algoritmos.

O teste de *Friedman* é um teste não paramétrico utilizado para comparar um conjunto de  $k$  grupos de dados ( $k \geq 2$ ) pareados. Assim como no teste de *Mann-Whitney*, o teste de *Friedman* também avalia uma hipótese nula. Neste caso, a hipótese nula  $H_0$  considera igualdade das medianas entre os grupos de dados dos algoritmos comparados. Quando a hipótese nula  $H_0$  é rejeitada, tem-se que, ao menos um dos algoritmos possui desempenho diferente dos demais.

Entretanto o teste de *Friedman* possui uma desvantagem. Embora seja possível determinar a existência de uma diferença de desempenho entre os algoritmos avaliados, o teste de *Friedman* não é capaz de fornecer a informação de quais algoritmos possuem desempenho diferente dos demais.

Sendo assim, para obter esta resposta, um procedimento denominado *post hoc* é aplicado.

Neste procedimento *post hoc* considera-se uma comparação do tipo *um contra todos* na qual um dos  $k$  algoritmos é escolhido como algoritmo de controle e este é comparado com cada um dos algoritmos restantes. Para cada comparação uma hipótese nula é definida. Consequentemente, uma família composta de  $k - 1$  hipóteses é gerada.

De acordo com Derrac et al. (2011), o valor-p de cada hipótese da família pode ser obtido utilizando-se uma aproximação normal através de uma conversão das classificações dos algoritmos obtidas pelo teste de *Friedman*.

Para isto, primeiramente calcula-se o valor-z para cada hipótese utilizando-se a seguinte equação:

$$z = \frac{(R_c - R_i)}{\sqrt{\frac{k \cdot (k + 1)}{6 \cdot n}}} \quad (68)$$

onde  $R_c$  e  $R_i$  correspondem ao valor médio da classificação do algoritmo de controle e ao valor médio da classificação do  $i$  –ésimo algoritmo de comparação respectivamente, obtidos através do teste de *Friedman*.  $k$  é o número de algoritmos que estão sendo avaliados e  $n$  é o tamanho do grupo de dados de cada algoritmo.

O valor-z obtido na equação (68) pode ser então utilizado para determinar o correspondente valor-p a partir de uma tabela de distribuição normal  $N(0, 1)$ .

O valor-p obtido através deste procedimento reflete apenas a probabilidade de erro de uma determinada comparação. Ele não leva em conta todas as comparações restantes da família. Consequentemente, este valor-p não é adequado quando se realizam múltiplas comparações (DERRAC et al., 2011).

O problema relacionado a múltiplas comparações se deve ao fato de que à medida que se aumenta o número de testes de hipóteses, aumenta-se também a probabilidade de rejeitar uma hipótese nula quando ela é verdadeira (erro Tipo I). Esta probabilidade é também chamada de taxa de erro da família de testes (em inglês *family-wise error rate* - FWER). A FWER nada mais é do que é a probabilidade de rejeitarmos incorretamente ao menos uma das hipóteses nulas que compõem a família.

Para controlar a FWER um ajuste nos valores-p é necessário. O procedimento escolhido neste trabalho de ajuste dos valores-p é o procedimento de Holm (DERRAC et al., 2011). Neste método, o valor-p ajustado (VPA) é calculado da seguinte maneira.

Seja  $p_{(1)}, p_{(2)}, \dots, p_{(k)}$  a ordenação dos valores-p  $p_1, p_2, \dots, p_k$  da família de hipótese tal que  $p_{(1)} \leq p_{(2)} \leq \dots \leq p_{(k)}$  e sejam  $H_{(1)}, H_{(2)}, \dots, H_{(k)}$  as hipóteses correspondentes. O valor-p ajustado, conforme o procedimento de Holm, associado a cada hipótese  $H_{(1)}, H_{(2)}, \dots, H_{(k)}$  é determinado através da seguinte fórmula

$$VPA_i = \min\{\vartheta, 1\} \quad (69)$$

onde

$$\vartheta = \max\{(k - j) \cdot p_{(j)} : 1 \leq j \leq i\} \quad (70)$$

O valor-p ajustado de cada hipótese calculado através das equações (69) e (70) pode ser então comparado diretamente com o nível de significância desejado. Caso o valor-p ajustado  $VPA_i$  seja menor que o nível de significância especificado, a

hipótese nula  $H_{(i)}$  é rejeitada e considera-se que o algoritmo de controle possui um desempenho melhor que o respectivo algoritmo de comparação.

A partir destas considerações, o teste *post hoc* de *Friedman* será utilizado para comparar o desempenho de cada um dos algoritmos propostos neste trabalho em relação aos outros oito algoritmos JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX. Para isto, dois conjuntos de algoritmos serão avaliados pelo teste *post hoc* de *Friedman*:

- HSADE-v1, JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL, DEahcSPX, onde HSADE-v1 será o algoritmo de controle;
- HSADE-v2, JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL, DEahcSPX, onde HSADE-v2 será o algoritmo de controle;

Nota-se que nos dois casos  $k = 9$ .

Visto que os dados utilizados pelo teste de *Friedman* possuem a característica de serem pareados, o grupo de dados de cada algoritmo utilizado no teste de *Friedman* será formado, conforme proposto em Derrac et al. (2011), pelo conjunto das médias dos valores de erro de cada função teste obtido em cada experimento para os estudos de casos referentes ao CEC2005 e CEC2013 e pelo conjunto das médias dos valores de cada função teste obtido em cada experimento para o estudo de caso referente ao CEC2011. O valor de  $n$  utilizado na equação (68) corresponderá ao número de funções testes do estudo de caso.

### 6.3 ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2005

Um conjunto de 25 funções testes propostas no *Special Session & Competition on Real-Parameter Optimization* do CEC2005 foi escolhido para a verificação do desempenho das duas versões do algoritmo *HSADE* apresentadas nesta dissertação.

As características deste conjunto de funções testes podem ser brevemente resumidas como se segue:

- $F_1$  a  $F_5$  funções unimodais;
- $F_6$  a  $F_{12}$  funções multimodais básicas;
- $F_{13}$  a  $F_{14}$  funções multimodais expandidas;
- $F_{15}$  a  $F_{25}$  funções multimodais de composição híbrida.

A formulação matemática de cada uma destas funções, bem como suas respectivas propriedades, são descritas em Suganthan et al. (2005) e não serão apresentadas nesta dissertação.

Para este estudo de caso, todas as funções foram avaliadas nas dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ . Para reduzir o impacto da natureza estocástica dos algoritmos, cada algoritmo foi executado 50 vezes para cada função teste. Além disto, cada experimento foi finalizado após um número máximo de  $D \cdot 10000$  (100000 para  $D = 10$ , 300000 para  $D = 30$  e 500000 para  $D = 50$ ) avaliações da função objetivo (SUGANTHAN et al., 2005).

### 6.3.1 Definição dos parâmetros

Com relação aos parâmetros relativos aos algoritmos propostos nesta dissertação, a mesma configuração de parâmetros foi utilizada na otimização de todas as funções testes com o intuito de obter uma comparação justa.

Os valores de parâmetros de cada versão do algoritmo *HSADE* utilizados para este estudo de caso são apresentados nas TABELAS 2 e 3.

TABELA 2 – VALORES DOS PARÂMETROS DO ALGORITMO HSADE-V1

<b>Parâmetro</b>	<b>Descrição</b>	<b>Valor utilizado</b>	<b>Comentário</b>
$NP$	Tamanho da população	$NP=30$ para $D=10$ $NP=100$ para $D=30$ $NP=200$ para $D=50$	-
$\varphi$	Tamanho do arquivo externo	$NP$	Escolha feita para evitar o ajuste de um novo parâmetro
$c$	Parâmetro que define a taxa de adaptação dos parâmetros de controle	0,1	Escolha baseada nas orientações de Zhang e Sanderson (2009a)
$p$	Parâmetro que define a “ganância” (greediness) do operador de mutação DE/target-to-pbest	0,05	
$\beta$	Probabilidade da busca voltada à exploitation	0,1	Escolha baseada em resultados empíricos
$\Omega$	Número de gerações para atualização das probabilidades das estratégias de geração dos vetores experimentais	20	

FONTE: O autor (2014)

TABELA 3 – VALORES DOS PARÂMETROS DO ALGORITMO HSADE-V2

<b>Parâmetro</b>	<b>Descrição</b>	<b>Valor utilizado</b>	<b>Comentário</b>
$NP$	Tamanho da população	$NP=30$ para $D=10$ $NP=100$ para $D=30$ $NP=200$ para $D=50$	-
$\varphi$	Tamanho do arquivo externo	$NP$	Escolha feita para evitar o ajuste de um novo parâmetro
$\vartheta$	Tamanho do arquivo de armazenagem do histórico da média dos parâmetros de controle $F$ e $CR$	20	Escolha baseada nas orientações de Zhang e Sanderson (2009a)

FONTE: O autor (2014)

Os parâmetros dos outros oito algoritmos utilizados para comparação foram definidos segundo a orientação dos autores dos mesmos. Apenas em alguns casos, o tamanho da população teve que ser ajustado devido ao fato de não haver orientações dos autores originais. Mais especificamente, os parâmetros de cada um dos oito algoritmos foram definidos da seguinte forma.

No caso do algoritmo JADE, o tamanho da população foi definido como  $NP = 200$  para  $D = 50$ . O resto da configuração dos parâmetros foi definido de acordo com Zhang e Sanderson (2009a).

Para o algoritmo jDE o tamanho da população foi definido como  $NP = 30$  e  $200$  para  $D = 10$  e  $D = 50$ , respectivamente. O resto da configuração dos parâmetros foi definido de acordo com o Brest et al. (2006).

Para o algoritmo CoDE, o tamanho da população foi definido como  $NP = 10$ ,  $50$  para  $D = 10$  e  $D = 50$ , respectivamente. O resto da configuração dos parâmetros foi definido de acordo com Wang, Cai e Zangh (2011).

O ajuste dos parâmetros para o algoritmo EPSDE foi definido de acordo com Mallipeddi et al. (2011).

Para o algoritmo SaDE, o tamanho da população foi definido como  $NP = 200$  para  $D = 50$ . O resto da configuração dos parâmetros foi definido de acordo com Qin, Huang e Suganthan (2009).

E, finalmente, o ajuste dos parâmetros para o algoritmo DEGL foi definido de acordo com Das et al. (2009) e para o algoritmo DEahcSPX o ajuste dos parâmetros foi definido de acordo com Noman e Iba (2008).



### 6.3.2 Resultados experimentais e discussão

As TABELAS 12, 15 e 18 apresentam a média e o desvio padrão do erro da função objetivo calculados sobre 50 experimentos para cada uma das 25 funções testes do CEC2005 para as dimensões 10, 30 e 50, respectivamente, obtidos para cada um dos algoritmos: JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, HSADE-v1 e HSADE-v2.

As TABELAS 13, 16 e 19 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando HSADE-v1 como algoritmo de controle enquanto as TABELAS 14, 17 e 20 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando HSADE-v2 como algoritmo de controle. Em todas estas tabelas, os sinais "+", "≈" e "-" indicam se o algoritmo de controle apresentou um desempenho melhor, semelhante ou pior, respectivamente, do que o algoritmo de comparação para a função teste correspondente. No final de cada tabela é apresentado o número total de funções testes nas quais o algoritmo de controle apresentou um desempenho melhor (+), similar (≈) ou pior (-) quando comparado com cada um dos oito algoritmos estado da arte baseados na evolução diferencial.

Nas seções subseqüentes, uma análise mais detalhada dos resultados experimentais obtidos na solução das funções testes propostas no CEC2005 será apresentada para cada um dos algoritmos HSADE-v1 e HSADE-v2.

#### 6.3.2.1 Análise do desempenho do algoritmo HSADE-v1

Nesta seção, uma análise dos resultados obtidos no teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v1 como algoritmo de controle é realizada.

##### 6.3.2.1.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados nas TABELAS 13, 16 e 19. Uma análise dos resultados obtidos para cada grupo de função teste é detalhada na seqüência.

- Funções unimodais

Em geral, o algoritmo HSADE-v1 teve um desempenho melhor, ou pelo menos comparável, do que as oito variantes DE para as cinco funções testes unimodais.

Nota-se que nenhum dos quatro algoritmos CoDE, SaDE, jDE e DEahcSPX apresentaram um resultado melhor do HSADE-v1 nas três dimensões ( $D = 10, 30$  e  $50$ ). HSADE-v1 sempre obteve um desempenho melhor ou similar em todas as cinco funções quando comparado a estes quatro algoritmos.

Já o algoritmo JADE *sem arquivo* superou HSADE-v1 apenas na função  $F_2$  na dimensão  $D = 30$ . Em todos os outros casos HSADE-v1 apresentou um desempenho superior.

Desempenho semelhante é observado quando se compara HSADE-v1 com EPSDE. O algoritmo EPSDE superou HSADE-v1 apenas na função  $F_2$  na dimensão  $D = 50$ . Em todos os outros casos HSADE-v1 apresentou um desempenho superior.

No caso de uma comparação com o algoritmo DEGL, HSADE-v1 foi superado apenas na função  $F_3$  na dimensão  $D = 10$ . Em todos os outros casos HSADE-v1 apresentou um desempenho similar ou superior.

Quando se compara o desempenho de HSADE-v1 com o algoritmo JADE *com arquivo* não se observa uma clara vantagem no desempenho de HSADE-v1. HSADE-v1 foi melhor em alguns casos e JADE *com arquivo* em outros. Isto se deve provavelmente ao fato de o algoritmo JADE *com arquivo* possuir características gulosas, que são vantajosas na solução de problemas unimodais, devido à utilização apenas da estratégia *DE/current-to-pbest/1/bin* o que direciona a geração dos vetores experimentais em direção a melhor solução corrente.

- Funções multimodais básicas

HSADE-v1 apresentou um desempenho em geral muito bom para as sete funções básicas multimodais.

HSADE-v1 possui desempenho semelhante ou melhor do que todos os oito algoritmos de comparação em cinco ( $F_6, F_8, F_9, F_{10}, F_{12}$ ), seis ( $F_6, F_7, F_9, F_{10}, F_{11},$

$F_{12}$ ), quatro ( $F_7, F_8, F_9, F_{10}$ ) funções para o caso de dimensão 10, 30 e 50 respectivamente.

HSADE-v1 só é superado por JADE *com arquivo* na função  $F_7$  e por SaDE e DEGL na função  $F_{11}$  para o caso de 10 dimensões. No caso de 30 dimensões, HSADE-v1 só é superado por CoDE na função  $F_8$ . No caso de 50 dimensões, HSADE-v1 apresentou um pior desempenho na função  $F_6$  quando comparado com JADE *com arquivo* e EPSDE, na função  $F_{11}$  quando comparado com CoDE e na função  $F_{12}$  quando comparado com JADE *com arquivo* e CoDE.

- Funções multimodais expandidas

Para as funções de 30 e 50 dimensões, HSADE-v1 supera todos os oito algoritmos de comparação em ambas as funções. Nenhum outro algoritmo apresentou um melhor desempenho ou até mesmo semelhante do que HSADE-v1 em qualquer uma das duas funções teste.

Para as funções de 10 dimensões, HSADE-v1 consegue um desempenho semelhante ou melhor do que todos os oito algoritmos de comparação para  $F_{14}$  e exibe um desempenho semelhante ou melhor do que o CoDE, SaDE, DEGL e DEahcSPX para  $F_{13}$ .

- Funções multimodais de composição híbrida

O conjunto de funções híbridas é formado por funções multimodais que têm um grande número de ótimos locais (SUGANTHAN et al., 2005) o que faz com que estas funções sejam muito mais difíceis de serem resolvidas. Mesmo assim, HSADE-v1 obteve um desempenho significativo.

Para as funções de 10 dimensões, HSADE-v1 obteve um desempenho melhor ou semelhante do que seis algoritmos de comparação em todas as funções: JADE com e sem arquivo, CoDE, SaDE, jDE, e DEGL. Com relação aos outros dois algoritmos de comparação, EPSDE superou HSADE-v1 apenas nas funções  $F_{15}$  e  $F_{23}$  e DEahcSPX superou HSADE-v1 nas funções  $F_{18}, F_{19}, F_{20}$  e  $F_{23}$ . Em todos os outros casos HSADE-v1 apresentou um desempenho melhor ou similar que EPSDE e DEahcSPX.

Para o caso de 30 dimensões, HSADE-v1 apresentou um desempenho excepcional. HSADE-v1 superou ou apresentou em alguns poucos casos desempenho semelhante do que todos os algoritmos de comparação em nove das onze funções:  $F_{15}$ ,  $F_{16}$ ,  $F_{17}$ ,  $F_{18}$ ,  $F_{19}$ ,  $F_{20}$ ,  $F_{22}$ ,  $F_{23}$  e  $F_{24}$ . Para a função  $F_{21}$  HSADE-v1 foi superado apenas por JADE com *arquivo* e CoDE e no caso da função  $F_{25}$  HSADE-v1 foi superado apenas por jDE, DEGL e DEahcSPX.

Para o caso de 50 dimensões, HSADE-v1 superou ou apresentou em alguns poucos casos desempenho semelhante do que todos os algoritmos de comparação em sete funções:  $F_{16}$ ,  $F_{17}$ ,  $F_{21}$ ,  $F_{22}$ ,  $F_{23}$ ,  $F_{24}$  e  $F_{25}$ . HSADE-v1 foi superado na função  $F_{15}$  apenas por JADE com e sem *arquivo* e jDE. Para as funções  $F_{18}$ ,  $F_{19}$  e  $F_{01}$  HSADE-v1 foi superado por CoDE, jDE e DEahcSPX.

#### 6.3.2.1.2 Teste de *Post hoc* de *Friedman*

Os resultados obtidos através do teste de *Mann-Whitney* permitem apenas realizar um comparativo de desempenho do HSADE-v1 com apenas um algoritmo por vez. Por este motivo, para ser possível de realizar uma análise global sobre a eficiência do algoritmo HSADE-v1, o teste de *post hoc* de *Friedman* também foi aplicado aos resultados experimentais.

A TABELA 4 apresenta a classificação média dos algoritmos considerados e o valor-p obtidos através do teste de *Friedman* para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ . Os baixos valores-p obtidos sugerem fortemente a existência de diferenças significativas no desempenho entre os algoritmos comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada para as três dimensões avaliadas com um nível de significância de  $\alpha = 0,0001$ .

Pode-se observar também através dos dados da TABELA 4, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v1 foi o que apresentou o melhor resultado nas dimensões 10 e 30. Na dimensão 50, apenas o algoritmo JADE *com arquivo* apresentou uma classificação melhor.

TABELA 4 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2005

	<i>D</i> = 10	<i>D</i> = 30	<i>D</i> = 50
JADE <i>com arquivo</i>	4,46	3,98	2,70
JADE <i>sem arquivo</i>	5,06	5,00	3,54
CoDE	7,88	4,18	4,10
SaDE	4,66	5,64	6,34
jDE	5,24	5,42	5,10
EPSDE	4,80	5,88	7,20
DEGL	4,96	6,20	5,92
DEahcSPX	4,82	5,88	7,24
HSADE-v1	3,12	2,82	2,86
<i>Valor-p</i>	8,61E-07	2,53E-05	3,77E-15

FONTE: O autor (2014)

Tendo em mãos os dados da classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. As TABELAS 5, 6 e 7 apresentam os valores-p e seus respectivo valores-p ajustados através do procedimento de Holm para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$  respectivamente considerando o algoritmo HSADE-v1 como algoritmo de controle.

A partir destes resultados, algumas considerações podem ser feitas.

Primeiramente, pode-se observar na TABELA 5 que o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v1 sobre CoDE e jDE para o caso de 10 dimensões considerando-se um nível de significância  $\alpha = 0,05$ . Já para os algoritmos JADE *sem arquivo* e DEGL, HSADE-v1 pode ser considerado melhor para um nível de significância  $\alpha = 0,1$ . Para os outros quatro algoritmos, HSADE-v1 é considerado melhor para um nível de significância  $\alpha = 0,12$ .

A partir dos resultados apresentados na

TABELA 6, nota-se que o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v1 sobre DEGL, EPSDE, DEahcSPX, SaDE, jDE e JADE *sem arquivo*. Para todos estes, HSADE-v1 foi melhor considerando-se um nível de significância  $\alpha = 0,02$ . Para os outros dois algoritmos, CoDE e JADE *com arquivo*, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v1, pois o valor-p obtido foi maior que 0,15.

Como pode ser observado na TABELA 7, o teste *post hoc* de *Friedman* mostrou uma melhoria do algoritmo HSADE-v1 sobre DEahcSPX, EPSDE, SaDE, DEGL e jDE considerando-se um nível de significância  $\alpha = 0,02$ . Para os outros três algoritmos, CoDE e JADE com e sem arquivo, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v1, pois o valor-p obtido foi maior que 0,3.

TABELA 5 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
CoDE	7,99E-10	6,39E-09
jDE	0,006	0,043
JADE <i>sem arquivo</i>	0,012	0,074
DEGL	0,018	0,088
DEahcSPX	0,028	0,113
EPSDE	0,030	0,113
SaDE	0,047	0,113
JADE <i>com arquivo</i>	0,084	0,113

FONTE: O autor (2014)

TABELA 6 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEGL	1,28E-05	1,02E-04
EPSDE	7,80E-05	5,46E-04
DEahcSPX	7,80E-05	0,001
SaDE	2,72E-04	0,001
jDE	0,001	0,003
JADE <i>sem arquivo</i>	0,005	0,015
CoDE	0,079	0,158
JADE <i>com arquivo</i>	0,134	0,158

FONTE: O autor (2014)

TABELA 7 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	1,56E-08	1,25E-07
EPSDE	2,11E-08	1,48E-07
SaDE	7,03E-06	4,22E-05
DEGL	7,80E-05	3,90E-04
jDE	0,004	0,015
CoDE	0,109	0,328
JADE <i>sem arquivo</i>	0,380	0,760
JADE <i>com arquivo</i>	1	1

FONTE: O autor (2014)

### 6.3.2.2 Análise do desempenho do algoritmo HSADE-v2

Nesta seção uma análise dos resultados obtidos no teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v2 como algoritmo de controle é realizada.

#### 6.3.2.2.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados nas TABELAS 14, 17 e 20. Uma análise dos resultados obtidos para cada grupo de função teste é detalhada na sequência.

- *Funções unimodais*

Em geral, o algoritmo HSADE-v2 teve um desempenho melhor, ou pelo menos comparável, do que sete variantes DE (JADE *sem arquivo*, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX) para as cinco funções testes unimodais. Apenas quando se compara o desempenho do algoritmo HSADE-v2 com o algoritmo JADE *com arquivo* não se observa uma vantagem de HSADE-v2, tendência esta que também já havia sido observada na primeira versão HSADE-v1. Isto se deve provavelmente ao fato do algoritmo JADE *com arquivo* possuir mais características gulosas que o algoritmo HSADE-v2, o que é muito vantajoso para funções

unimodais, pois no algoritmo HSADE-v2 têm-se  $\lambda$  variando entre 2 e  $NP$  durante o processo de otimização enquanto no algoritmo JADE *com arquivo* têm-se  $\lambda = 2$ , o que faz com que a geração dos vetores experimentais seja sempre direcionada para a melhor solução corrente.

Com relação aos algoritmos CoDE, SaDE, jDE e DEahcSPX, nota-se que nenhum deles apresentaram um resultado melhor do que HSADE-v2 nas três dimensões consideradas. HSADE-v2 sempre obteve um desempenho melhor ou similar em todas as cinco funções quando comparado a estes quatro algoritmos.

Já o algoritmo JADE *sem arquivo* superou HSADE-v2 apenas na função  $F_2$  na dimensão  $D = 30$  e  $D = 50$ . Em todos os outros casos HSADE-v2 apresentou um desempenho similar ou superior.

Quando se compara HSADE-v2 com EPSDE, verifica-se que o algoritmo EPSDE supera HSADE-v2 apenas na função  $F_2$  na dimensão  $D = 50$ . Em todos os outros casos HSADE-v2 apresentou um desempenho similar ou superior.

No caso de uma comparação com o algoritmo DEGL, HSADE-v2 foi superado apenas nas funções  $F_4$  e  $F_5$  em ambas as dimensões  $D = 30$  e  $D = 50$ . Em todos os outros casos HSADE-v2 apresentou um desempenho similar ou superior.

- Funções multimodais básicas

Com relação às funções multimodais básicas, HSADE-v2 apresentou em geral um bom desempenho.

HSADE-v2 possui desempenho semelhante ou melhor do que todos os oito algoritmos de comparação em cinco das sete funções analisadas:  $F_6, F_8, F_9, F_{10}, F_{12}$  para o caso de dimensão 10,  $F_6, F_9, F_{10}, F_{11}, F_{12}$  para o caso de dimensão 30 e  $F_7, F_8, F_{10}, F_{11}, F_{12}$  para o caso de dimensão 50.

Para o caso de dimensão 10, HSADE-v2 é superado apenas por JADE *com arquivo* na função  $F_7$  e por SaDE e DEGL na função  $F_{11}$ . No caso de 30 dimensões, HSADE-v2 é superado apenas por CoDE nas funções  $F_7$  e  $F_8$ . Já no caso de 50 dimensões, HSADE-v2 apresentou um pior desempenho somente na função  $F_6$  quando comparado com JADE *com* e *sem arquivo* e EPSDE e pior desempenho somente na função  $F_9$  quando comparado com JADE *com* e *sem arquivo*, CoDE, SaDE, jDE e EPSDE.



- Funções multimodais expandidas

Para a função  $F_{14}$ , HSADE-v2 supera, ou apresenta no mínimo desempenho semelhante, todos os oito algoritmos de comparação nas três dimensões consideradas. Nenhum outro algoritmo apresentou um melhor desempenho do que HSADE-v2 nesta função teste.

No caso da função  $F_{13}$ , HSADE-v2 apresentou um desempenho melhor do que CoDE, SaDE, DEGL e DEahcSPX para o caso de 10 dimensões, um desempenho semelhante ou melhor do que SaDE, EPSDE, DEGL e DEahcSPX para o caso de 30 dimensões e para o caso de 50 dimensões, HSADE-v2 é superado apenas por JADE *com arquivo*.

- Funções multimodais de composição híbrida

Para as funções de 10 dimensões, HSADE-v2 obteve um desempenho melhor ou semelhante do que quatro algoritmos de comparação (JADE com e sem arquivo, CoDE e jDE) em todas as funções. Com relação aos outros algoritmos de comparação, EPSDE superou HSADE-v2 somente nas funções  $F_{15}$ ,  $F_{21}$  e  $F_{23}$ , SaDE superou HSADE-v2 apenas na função  $F_{15}$  e o algoritmo DEGL superou HSADE-v2 apenas na função  $F_{16}$ . Com relação ao algoritmo DEahcSPX, HSADE-v2 foi superado nas funções  $F_{18}$ ,  $F_{19}$ ,  $F_{20}$ ,  $F_{21}$ ,  $F_{23}$  e  $F_{24}$ . Em todos os outros casos HSADE-v2 apresentou um desempenho melhor ou similar que SaDE, EPSDE, DEGL e DEahcSPX.

Para o caso de 30 dimensões, HSADE-v2 apresentou um excelente desempenho. HSADE-v2 superou ou apresentou, em alguns poucos casos desempenho semelhante, do que todos os algoritmos de comparação em sete das onze funções:  $F_{15}$ ,  $F_{16}$ ,  $F_{17}$ ,  $F_{19}$ ,  $F_{22}$ ,  $F_{23}$  e  $F_{24}$ . Para a função  $F_{21}$ , HSADE-v2 foi superado apenas por JADE *com arquivo* e CoDE e no caso da função  $F_{25}$  HSADE-v2 foi superado apenas por DEGL e DEahcSPX. Já para as funções  $F_{18}$  e  $F_{20}$ , HSADE-v2 foi superado apenas por DEahcSPX. Em todas as outras funções HSADE-v2 foi superior ou obteve desempenho semelhante que JADE *com arquivo*, CoDE, DEGL e DEahcSPX.

Para o caso de 50 dimensões, HSADE-v2 possui desempenho semelhante ou melhor do que SaDE, EPSDE e DEGL em todas as onze funções. Já com relação aos outros algoritmos HSADE-v2 possui desempenho semelhante ou melhor do que JADE com e sem arquivo, CoDE, jDE e DEahcSPX em seis ( $F_{16}$ ,  $F_{17}$ ,  $F_{21}$ ,  $F_{22}$ ,  $F_{23}$ ,  $F_{25}$ ), sete ( $F_{15}$ ,  $F_{16}$ ,  $F_{17}$ ,  $F_{21}$ ,  $F_{22}$ ,  $F_{23}$ ,  $F_{25}$ ), sete ( $F_{16}$ ,  $F_{17}$ ,  $F_{21}$ ,  $F_{22}$ ,  $F_{23}$ ,  $F_{24}$ ,  $F_{25}$ ) e oito ( $F_{15}$ ,  $F_{16}$ ,  $F_{17}$ ,  $F_{21}$ ,  $F_{22}$ ,  $F_{23}$ ,  $F_{24}$ ,  $F_{25}$ ) funções respectivamente.

### 6.3.2.2.2 Teste *post hoc* de *Friedman*

A TABELA 8 apresenta a classificação média dos algoritmos considerados e o valor-p obtidos através do teste de *Friedman* para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ .

Os baixos valores-p obtidos sugerem fortemente a existência de diferenças significativas no desempenho entre os algoritmos comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada para as três dimensões avaliadas com um nível de significância de  $\alpha = 0,0001$ .

Pode-se observar também através dos dados da TABELA 8, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v2 foi o que apresentou o melhor resultado nas dimensões 10 e 30. Na dimensão 50, apenas o algoritmo JADE *com arquivo* apresentou uma classificação melhor.

TABELA 8 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2005

	<i>D = 10</i>	<i>D = 30</i>	<i>D = 50</i>
JADE <i>com arquivo</i>	4,50	3,82	2,56
JADE <i>sem arquivo</i>	5,06	5,00	3,42
CoDE	7,88	4,22	4,12
SaDE	4,58	5,68	6,24
jDE	5,16	5,38	5,06
EPSDE	4,80	5,92	7,20
DEGL	4,92	6,16	5,80
DEahcSPX	4,84	5,80	7,24
HSADE-v2	3,26	3,02	3,36

<i>Valor-p</i>	1,87E-06	6,83E-05	3,31E-14
----------------	----------	----------	----------

FONTE: O autor (2014)

Uma vez calculada a classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. As TABELAS 9, 10 e 11 apresentam os valores-p e seus respectivos valores-p ajustados através do procedimento de Holm para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$  respectivamente considerando o algoritmo HSADE-v2 como algoritmo de controle.

Como pode ser verificado na TABELA 9, o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v2 sobre CoDE, jDE e JADE *sem arquivo* para o caso de 10 dimensões considerando-se um nível de significância  $\alpha = 0,12$ . Para os outros cinco algoritmos, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v2 pois o valor-p obtido foi maior que 0,15.

A partir dos resultados apresentados na TABELA 10, para o caso de 30 dimensões, nota-se que o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v2 sobre DEGL, EPSDE, DEahcSPX, SaDE, jDE e JADE *sem arquivo*. Para todos estes casos, HSADE-v2 foi melhor considerando-se um nível de significância  $\alpha = 0,04$ . Para os outros dois algoritmos, CoDE e JADE *com arquivo*, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v2 pois o valor-p obtido foi maior que 0,24.

Com relação à dimensão 50, observa-se na TABELA 11, que o teste *post hoc* de *Friedman* mostrou uma melhoria do algoritmo HSADE-v2 sobre DEahcSPX, EPSDE, SaDE, DEGL considerando-se um nível de significância  $\alpha = 0,01$ . Já para o algoritmo jDE, HSADE-v2 foi superior considerando um nível de significância  $\alpha = 0,12$ . Para os outros três algoritmos, CoDE e JADE *com* e *sem* arquivo, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado a HSADE-v2.

TABELA 9 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
CoDE	2,46E-09	1,96E-08
jDE	0,014	0,099
JADE <i>sem arquivo</i>	0,020	0,120
DEGL	0,032	0,161
DEahcSPX	0,041	0,165
EPSDE	0,047	0,165
SaDE	0,088	0,177
JADE <i>com arquivo</i>	0,109	0,177

FONTE: O autor (2014)

TABELA 10 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEGL	5,04E-05	4,03E-04
EPSDE	1,81E-04	0,001
DEahcSPX	3,32E-04	0,002
SaDE	0,001	0,003
jDE	0,002	0,009
JADE <i>sem arquivo</i>	0,011	0,032
CoDE	0,121	0,243
JADE <i>com arquivo</i>	0,302	0,302

FONTE: O autor (2014)

TABELA 11 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	5,47E-07	4,38E-06
EPSDE	7,14E-07	5,00E-06
SaDE	2,01E-04	0,001
DEGL	0,002	0,008
jDE	0,028	0,113
CoDE	0,327	0,980
JADE <i>sem arquivo</i>	0,938	1
JADE <i>com arquivo</i>	1	1

FONTE: O autor (2014)

TABELA 12 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSAD-1	HSAD-2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	9,91E-30 ± 4,03E-29	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	9,34E-30 ± 2,87E-29	5,68E-29 ± 9,76E-29	1,71E-27 ± 6,30E-27	1,62E-29 ± 4,61E-29	9,72E-29 ± 1,03E-28	7,04E-29 ± 8,49E-29	0,00E+00 ± 0,00E+00	1,40E-25 ± 6,71E-25	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_3$	9,90E-26 ± 6,54E-26	6,08E-03 ± 3,10E-02	5,25E+03 ± 6,34E+03	6,22E+03 ± 1,50E+04	2,79E+03 ± 5,66E+03	1,91E+01 ± 9,14E+01	1,52E-25 ± 5,57E-26	8,76E-07 ± 2,70E-06	3,91E-25 ± 7,09E-25	1,60E-25 ± 1,00E-25
$F_4$	7,07E-30 ± 2,89E-29	4,72E-29 ± 8,16E-29	1,50E-27 ± 5,77E-27	2,37E-29 ± 5,26E-29	1,12E-28 ± 1,82E-28	5,93E-29 ± 9,54E-29	0,00E+00 ± 0,00E+00	6,81E-23 ± 3,07E-22	6,06E-30 ± 2,42E-29	2,02E-30 ± 1,43E-29
$F_5$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	4,37E-13 ± 1,19E-12	1,46E-13 ± 7,20E-13	2,55E-13 ± 9,01E-13	4,18E-09 ± 2,03E-08	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_6$	4,26E-01 ± 1,15E+00	1,04E+00 ± 3,50E+00	1,04E+00 ± 1,77E+00	1,55E+00 ± 1,34E+00	4,78E-01 ± 1,31E+00	9,31E-27 ± 2,64E-26	3,19E-01 ± 1,09E+00	7,97E-02 ± 5,64E-01	2,40E-01 ± 9,56E-01	7,97E-02 ± 5,64E-01
$F_7$	1,31E-02 ± 9,80E-03	2,90E-02 ± 3,07E-02	1,42E-01 ± 1,02E-01	2,84E-02 ± 2,59E-02	5,68E-02 ± 3,57E-02	3,54E-02 ± 1,76E-02	3,68E-02 ± 5,96E-02	8,77E-02 ± 8,75E-02	3,33E-02 ± 2,20E-02	2,05E-02 ± 1,66E-02
$F_8$	2,03E+01 ± 1,15E-01	2,03E+01 ± 1,79E-01	2,04E+01 ± 1,88E-01	2,04E+01 ± 7,67E-02	2,03E+01 ± 6,69E-02	2,03E+01 ± 6,89E-02	2,04E+01 ± 8,03E-02	2,03E+01 ± 8,38E-02	2,03E+01 ± 8,83E-02	2,03E+01 ± 1,29E-01
$F_9$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,10E+00 ± 1,16E+00	0,00E+00 ± 0,00E+00	3,98E-02 ± 1,97E-01	0,00E+00 ± 0,00E+00	1,27E+01 ± 7,39E+00	9,84E-01 ± 1,02E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_{10}$	5,22E+00 ± 1,84E+00	5,57E+00 ± 2,09E+00	1,45E+01 ± 6,76E+00	5,99E+00 ± 2,61E+00	6,90E+00 ± 2,77E+00	6,00E+00 ± 1,62E+00	4,61E+00 ± 3,43E+00	7,41E+00 ± 4,41E+00	4,78E+00 ± 1,91E+00	5,09E+00 ± 1,91E+00
$F_{11}$	5,06E+00 ± 8,63E-01	4,68E+00 ± 8,22E-01	4,69E+00 ± 1,61E+00	9,38E-01 ± 8,94E-01	2,27E+00 ± 1,34E+00	5,40E+00 ± 6,35E-01	6,72E-01 ± 7,82E-01	2,64E+00 ± 2,87E+00	1,74E+00 ± 8,55E-01	1,65E+00 ± 1,14E+00
$F_{12}$	3,21E+02 ± 1,18E+03	1,45E+02 ± 4,33E+02	3,45E+02 ± 6,40E+02	3,99E+01 ± 2,39E+02	1,77E+02 ± 4,14E+02	6,38E+01 ± 7,99E+01	5,38E+02 ± 1,37E+03	9,82E+01 ± 3,33E+02	6,68E+01 ± 3,07E+02	5,50E+01 ± 2,40E+02
$F_{13}$	2,38E-01 ± 5,58E-02	2,61E-01 ± 4,50E-02	5,73E-01 ± 1,97E-01	5,84E-01 ± 7,66E-02	2,73E-01 ± 6,26E-02	3,07E-01 ± 4,17E-02	1,46E+00 ± 3,23E-01	7,95E-01 ± 4,06E-01	3,82E-01 ± 1,21E-01	4,20E-01 ± 1,03E-01
$F_{14}$	2,78E+00 ± 3,88E-01	2,84E+00 ± 2,66E-01	3,50E+00 ± 2,86E-01	2,86E+00 ± 2,67E-01	3,00E+00 ± 3,55E-01	3,14E+00 ± 2,09E-01	2,81E+00 ± 2,72E-01	3,03E+00 ± 5,45E-01	2,75E+00 ± 4,28E-01	2,67E+00 ± 4,32E-01

TABELA 12 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{15}$	1,83E+02 ± 1,89E+02	1,18E+02 ± 1,67E+02	2,53E+02 ± 1,72E+02	2,78E+01 ± 6,18E+01	1,13E+02 ± 1,32E+02	1,96E+01 ± 6,04E+01	1,89E+02 ± 1,47E+02	2,30E+02 ± 1,86E+02	1,05E+02 ± 1,64E+02	1,13E+02 ± 1,60E+02
$F_{16}$	1,02E+02 ± 7,66E+00	1,01E+02 ± 6,83E+00	1,26E+02 ± 1,60E+01	9,89E+01 ± 7,59E+00	1,05E+02 ± 6,02E+00	9,93E+01 ± 6,58E+00	1,02E+02 ± 1,61E+01	1,07E+02 ± 1,27E+01	9,87E+01 ± 5,93E+00	1,01E+02 ± 5,23E+00
$F_{17}$	1,07E+02 ± 8,13E+00	1,04E+02 ± 7,37E+00	1,30E+02 ± 1,92E+01	9,98E+01 ± 8,93E+00	1,07E+02 ± 1,09E+01	1,14E+02 ± 8,11E+00	1,00E+02 ± 8,35E+00	1,08E+02 ± 1,08E+01	1,03E+02 ± 8,13E+00	1,02E+02 ± 6,12E+00
$F_{18}$	6,43E+02 ± 2,70E+02	7,00E+02 ± 2,44E+02	7,26E+02 ± 2,27E+02	7,62E+02 ± 1,30E+02	6,20E+02 ± 2,49E+02	7,48E+02 ± 1,45E+02	6,94E+02 ± 2,43E+02	4,00E+02 ± 2,02E+02	5,65E+02 ± 2,68E+02	6,27E+02 ± 2,58E+02
$F_{19}$	6,42E+02 ± 2,74E+02	6,91E+02 ± 2,50E+02	7,01E+02 ± 2,34E+02	7,39E+02 ± 1,76E+02	5,84E+02 ± 2,61E+02	7,53E+02 ± 1,59E+02	6,90E+02 ± 2,31E+02	4,00E+02 ± 2,02E+02	6,54E+02 ± 2,60E+02	6,30E+02 ± 2,58E+02
$F_{20}$	6,41E+02 ± 2,73E+02	6,81E+02 ± 2,56E+02	6,98E+02 ± 2,35E+02	7,55E+02 ± 1,59E+02	5,76E+02 ± 2,42E+02	7,40E+02 ± 1,70E+02	7,39E+02 ± 2,25E+02	4,31E+02 ± 2,23E+02	6,30E+02 ± 2,64E+02	6,32E+02 ± 2,57E+02
$F_{21}$	6,30E+02 ± 2,58E+02	7,00E+02 ± 2,66E+02	7,43E+02 ± 2,79E+02	5,66E+02 ± 1,78E+02	6,57E+02 ± 2,05E+02	4,95E+02 ± 1,93E+02	5,50E+02 ± 2,55E+02	4,96E+02 ± 2,83E+01	5,27E+02 ± 2,43E+02	6,22E+02 ± 2,50E+02
$F_{22}$	7,69E+02 ± 3,41E+01	7,56E+02 ± 7,09E+01	7,76E+02 ± 7,79E+01	7,23E+02 ± 1,26E+02	7,63E+02 ± 2,46E+01	7,20E+02 ± 1,25E+02	7,72E+02 ± 2,26E+01	7,48E+02 ± 9,46E+01	7,22E+02 ± 1,10E+02	7,42E+02 ± 1,83E+01
$F_{23}$	8,06E+02 ± 2,50E+02	8,31E+02 ± 2,20E+02	8,19E+02 ± 2,31E+02	7,47E+02 ± 1,91E+02	7,94E+02 ± 2,41E+02	6,87E+02 ± 1,46E+02	7,56E+02 ± 2,24E+02	5,69E+02 ± 3,88E+01	7,83E+02 ± 1,88E+02	8,07E+02 ± 2,07E+02
$F_{24}$	2,68E+02 ± 1,48E+02	2,78E+02 ± 1,33E+02	2,38E+02 ± 1,26E+02	2,06E+02 ± 4,24E+01	2,12E+02 ± 5,94E+01	2,12E+02 ± 5,94E+01	2,48E+02 ± 1,11E+02	2,00E+02 ± 0,00E+00	2,18E+02 ± 7,20E+01	2,30E+02 ± 9,09E+01
$F_{25}$	4,18E+02 ± 1,55E+02	3,94E+02 ± 1,06E+02	3,81E+02 ± 6,20E+00	3,77E+02 ± 3,20E+00	3,76E+02 ± 3,30E+00	3,75E+02 ± 2,99E+00	4,14E+02 ± 1,27E+02	3,76E+02 ± 2,67E+00	3,93E+02 ± 1,06E+02	3,89E+02 ± 1,05E+02

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 100.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324

TABELA 13 – RESULTADOS DO TESTE DE MANN-WHITNEY PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
Funções Unimodais	$F_1$	≈	≈	+	≈	≈	≈	≈	≈
	$F_2$	+	+	+	+	+	+	≈	+
	$F_3$	-	+	+	+	+	+	-	+
	$F_4$	≈	+	+	+	+	+	≈	+
	$F_5$	≈	≈	+	≈	+	+	≈	≈
Funções Multimodais Básicas	$F_6$	+	+	+	+	+	≈	≈	+
	$F_7$	-	≈	+	≈	+	≈	≈	+
	$F_8$	≈	≈	+	≈	≈	≈	≈	≈
	$F_9$	≈	≈	+	≈	≈	≈	+	+
	$F_{10}$	+	+	+	+	+	+	≈	+
	$F_{11}$	+	+	+	-	+	+	-	≈
	$F_{12}$	+	+	+	+	+	+	+	+
Funções Multimodais Expandidas	$F_{13}$	-	-	+	+	-	-	+	+
	$F_{14}$	≈	≈	+	≈	+	+	≈	+
Funções multimodais de composição híbrida	$F_{15}$	+	≈	+	≈	+	-	+	+
	$F_{16}$	≈	≈	+	≈	+	≈	≈	+
	$F_{17}$	+	≈	+	≈	≈	+	≈	+
	$F_{18}$	≈	+	+	+	+	+	+	-
	$F_{19}$	≈	≈	≈	≈	≈	+	≈	-
	$F_{20}$	≈	+	≈	+	≈	+	+	-
	$F_{21}$	+	+	+	≈	+	≈	≈	≈
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	≈	≈	≈	≈	≈	-	≈	-
	$F_{24}$	+	+	≈	≈	≈	≈	≈	≈
$F_{25}$	≈	≈	+	≈	≈	≈	≈	≈	
<b>Total</b>	≈	12	12	4	14	9	9	16	7
	-	3	1	0	1	1	3	2	4
	+	10	12	21	10	15	13	7	14

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 14 – RESULTADOS DO TESTE DE MANN-WHITNEY PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
Funções Unimodais	$F_1$	≈	≈	+	≈	≈	≈	≈	≈
	$F_2$	+	+	+	+	+	+	≈	+
	$F_3$	-	+	+	+	+	+	≈	+
	$F_4$	≈	+	+	+	+	+	≈	+
	$F_5$	≈	≈	+	≈	+	+	≈	≈
Funções Multimodais Básicas	$F_6$	+	+	+	+	+	+	≈	+
	$F_7$	-	≈	+	≈	+	+	≈	+
	$F_8$	≈	≈	+	+	≈	≈	+	+
	$F_9$	≈	≈	+	≈	≈	≈	+	+
	$F_{10}$	≈	+	+	≈	+	+	≈	+
	$F_{11}$	+	+	+	-	+	+	-	≈
	$F_{12}$	+	+	+	+	+	+	+	+
Funções Multimodais Expandidas	$F_{13}$	-	-	+	+	-	-	+	+
	$F_{14}$	≈	+	+	+	+	+	≈	+
Funções multimodais de composição híbrida	$F_{15}$	≈	≈	+	-	≈	-	+	+
	$F_{16}$	≈	≈	+	≈	+	≈	-	+
	$F_{17}$	+	≈	+	≈	≈	+	≈	+
	$F_{18}$	≈	+	+	+	≈	+	≈	-
	$F_{19}$	≈	+	≈	+	≈	+	≈	-
	$F_{20}$	≈	+	≈	+	≈	+	+	-
	$F_{21}$	≈	≈	+	≈	≈	-	≈	-
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	≈	≈	≈	≈	≈	-	≈	-
	$F_{24}$	≈	+	≈	≈	≈	≈	≈	-
$F_{25}$	≈	+	+	+	+	+	+	+	
<b>Total</b>	≈	16	10	4	10	11	6	15	3
	-	3	1	0	2	1	4	2	6
	+	6	14	21	13	13	15	8	16

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.



TABELA 15 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADE-v1	HSADE-v2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,01E-30 ± 7,14E-30	0,00E+00 ± 0,00E+00	1,54E-28 ± 1,87E-28	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	9,87E-29 ± 8,76E-29	1,85E-28 ± 1,44E-28	3,21E-15 ± 1,05E-14	5,08E-06 ± 9,73E-06	5,59E-07 ± 6,95E-07	5,61E-26 ± 1,17E-25	6,13E-23 ± 1,13E-22	2,18E-05 ± 4,37E-05	3,02E-28 ± 2,01E-28	4,84E-28 ± 3,13E-28
$F_3$	9,13E+03 ± 6,92E+03	1,57E+04 ± 1,05E+04	1,31E+05 ± 9,56E+04	5,01E+05 ± 2,13E+05	1,90E+05 ± 9,41E+04	2,11E+06 ± 4,92E+06	5,92E+04 ± 4,24E+04	3,91E+05 ± 2,17E+05	5,56E+03 ± 4,78E+03	7,18E+03 ± 5,03E+03
$F_4$	1,50E-14 ± 1,01E-13	4,29E-09 ± 1,29E-08	1,36E-02 ± 5,48E-02	1,58E+02 ± 3,09E+02	2,32E-02 ± 3,44E-02	2,54E+01 ± 1,08E+02	1,42E-15 ± 2,71E-15	4,87E-01 ± 1,02E+00	1,87E-16 ± 6,78E-16	1,53E-12 ± 6,06E-12
$F_5$	2,10E-08 ± 1,30E-07	4,06E-02 ± 1,89E-01	4,76E+02 ± 4,23E+02	3,32E+03 ± 5,83E+02	5,42E+02 ± 4,56E+02	2,06E+03 ± 1,09E+03	1,16E-06 ± 2,97E-06	1,06E+02 ± 1,83E+02	5,21E-06 ± 1,57E-05	1,24E-02 ± 8,04E-02
$F_6$	9,39E+00 ± 2,58E+01	9,13E+00 ± 2,43E+01	1,59E-01 ± 7,89E-01	4,30E+01 ± 3,22E+01	2,39E+01 ± 2,54E+01	6,38E-01 ± 1,48E+00	1,04E+00 ± 1,77E+00	1,78E+00 ± 2,32E+00	2,75E+00 ± 1,21E+01	8,77E-01 ± 1,67E+00
$F_7$	9,02E-03 ± 7,18E-03	1,64E-02 ± 1,36E-02	5,81E-03 ± 7,70E-03	1,77E-02 ± 1,46E-02	1,36E-02 ± 9,08E-03	1,92E-02 ± 1,50E-02	9,80E-03 ± 9,43E-03	8,22E-03 ± 9,66E-03	7,49E-03 ± 7,50E-03	8,96E-03 ± 7,90E-03
$F_8$	2,09E+01 ± 1,43E-01	2,09E+01 ± 2,61E-01	2,01E+01 ± 1,30E-01	2,09E+01 ± 5,87E-02	2,09E+01 ± 5,46E-02	2,09E+01 ± 4,26E-02	2,09E+01 ± 5,80E-02	2,10E+01 ± 4,31E-02	2,09E+01 ± 4,78E-02	2,09E+01 ± 5,02E-02
$F_9$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,99E-02 ± 1,41E-01	1,39E-01 ± 4,03E-01	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	7,84E+01 ± 5,85E+01	2,00E+01 ± 5,94E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_{10}$	2,43E+01 ± 4,43E+00	2,57E+01 ± 5,24E+00	3,99E+01 ± 1,32E+01	4,70E+01 ± 1,16E+01	5,58E+01 ± 9,22E+00	4,93E+01 ± 9,93E+00	1,66E+02 ± 1,18E+01	6,10E+01 ± 6,06E+01	2,17E+01 ± 5,41E+00	1,92E+01 ± 4,59E+00
$F_{11}$	2,52E+01 ± 1,86E+00	2,54E+01 ± 1,60E+00	1,17E+01 ± 3,12E+00	1,68E+01 ± 2,52E+00	2,79E+01 ± 1,73E+00	2,88E+01 ± 3,36E+00	3,77E+01 ± 5,62E+00	3,55E+01 ± 8,06E+00	1,09E+01 ± 2,12E+00	7,61E+00 ± 2,43E+00
$F_{12}$	5,88E+03 ± 4,57E+03	5,94E+03 ± 4,16E+03	3,06E+03 ± 4,41E+03	3,68E+03 ± 3,35E+03	8,47E+03 ± 7,71E+03	2,17E+04 ± 5,55E+03	3,96E+03 ± 5,54E+03	2,84E+03 ± 2,69E+03	3,02E+03 ± 2,55E+03	1,81E+03 ± 1,97E+03
$F_{13}$	1,46E+00 ± 1,19E-01	1,47E+00 ± 1,03E-01	1,66E+00 ± 3,85E-01	3,94E+00 ± 3,79E-01	1,71E+00 ± 1,24E-01	1,93E+00 ± 1,57E-01	1,29E+01 ± 1,26E+00	2,99E+00 ± 9,02E-01	1,15E+00 ± 2,46E-01	1,93E+00 ± 3,54E-01
$F_{14}$	1,23E+01 ± 3,23E-01	1,22E+01 ± 3,58E-01	1,22E+01 ± 6,22E-01	1,27E+01 ± 2,33E-01	1,30E+01 ± 2,07E-01	1,29E+01 ± 3,04E-01	1,29E+01 ± 2,82E-01	1,31E+01 ± 2,00E-01	1,18E+01 ± 4,60E-01	1,16E+01 ± 4,78E-01

TABELA 15 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{15}$	3,78E+02 ± 7,08E+01	3,78E+02 ± 7,62E+01	4,06E+02 ± 5,86E+01	3,60E+02 ± 6,41E+01	3,56E+02 ± 8,37E+01	3,57E+02 ± 1,11E+02	3,58E+02 ± 1,01E+02	3,50E+02 ± 1,09E+02	3,80E+02 ± 6,06E+01	3,76E+02 ± 6,87E+01
$F_{16}$	1,31E+02 ± 1,44E+02	1,14E+02 ± 1,35E+02	7,51E+01 ± 3,22E+01	7,68E+01 ± 5,23E+01	7,48E+01 ± 1,88E+01	1,12E+02 ± 6,82E+01	2,13E+02 ± 7,00E+01	1,02E+02 ± 9,16E+01	4,90E+01 ± 2,23E+01	5,17E+01 ± 5,53E+01
$F_{17}$	1,09E+02 ± 8,16E+01	1,21E+02 ± 1,07E+02	7,12E+01 ± 2,57E+01	8,60E+01 ± 7,24E+01	1,33E+02 ± 2,24E+01	1,47E+02 ± 5,94E+01	2,59E+02 ± 8,92E+01	1,80E+02 ± 9,63E+01	8,26E+01 ± 8,83E+01	5,16E+01 ± 2,90E+01
$F_{18}$	9,04E+02 ± 8,24E-01	9,03E+02 ± 1,48E+01	9,05E+02 ± 9,98E-01	8,84E+02 ± 5,84E+01	9,04E+02 ± 8,78E-01	9,00E+02 ± 3,38E+01	9,02E+02 ± 2,12E+01	9,04E+02 ± 6,33E-01	9,04E+02 ± 5,03E-01	9,04E+02 ± 8,76E-01
$F_{19}$	9,04E+02 ± 7,37E-01	9,05E+02 ± 1,27E+00	9,05E+02 ± 1,09E+00	8,88E+02 ± 5,56E+01	9,04E+02 ± 9,18E-01	8,92E+02 ± 4,08E+01	9,04E+02 ± 1,52E+01	9,04E+02 ± 5,97E-01	9,04E+02 ± 8,44E-01	9,04E+02 ± 1,08E+00
$F_{20}$	9,04E+02 ± 8,80E-01	9,05E+02 ± 1,32E+00	9,05E+02 ± 1,13E+00	8,88E+02 ± 5,58E+01	9,04E+02 ± 1,13E+00	8,91E+02 ± 4,29E+01	9,04E+02 ± 1,52E+01	9,04E+02 ± 4,95E-01	9,04E+02 ± 7,12E-01	9,04E+02 ± 1,05E+00
$F_{21}$	5,00E+02 ± 3,41E-13	5,24E+02 ± 8,22E+01	5,00E+02 ± 3,34E-13	5,84E+02 ± 2,16E+02	5,00E+02 ± 3,18E-13	5,12E+02 ± 5,94E+01	5,48E+02 ± 1,12E+02	5,00E+02 ± 3,65E-13	5,00E+02 ± 3,17E-13	5,00E+02 ± 3,15E-13
$F_{22}$	8,62E+02 ± 2,14E+01	8,69E+02 ± 2,19E+01	8,64E+02 ± 2,28E+01	9,36E+02 ± 1,83E+01	8,75E+02 ± 1,81E+01	9,16E+02 ± 1,39E+01	8,91E+02 ± 1,68E+01	8,88E+02 ± 1,21E+01	8,35E+02 ± 2,11E+01	8,34E+02 ± 2,15E+01
$F_{23}$	5,42E+02 ± 5,70E+01	5,50E+02 ± 7,97E+01	5,34E+02 ± 4,59E-04	5,42E+02 ± 5,70E+01	5,34E+02 ± 2,70E-04	5,42E+02 ± 5,69E+01	5,68E+02 ± 1,12E+02	5,35E+02 ± 4,91E+00	5,34E+02 ± 1,12E-04	5,34E+02 ± 4,26E-13
$F_{24}$	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14	2,00E+02 ± 6,14E-13	2,00E+02 ± 9,44E-13	2,00E+02 ± 2,87E-14	2,00E+02 ± 2,87E-14
$F_{25}$	2,11E+02 ± 7,97E-01	2,11E+02 ± 9,77E-01	2,11E+02 ± 8,05E-01	2,13E+02 ± 1,46E+00	2,11E+02 ± 5,73E-01	2,13E+02 ± 1,66E+00	2,16E+02 ± 4,10E+01	2,10E+02 ± 8,29E-01	2,11E+02 ± 8,49E-01	2,11E+02 ± 7,28E-01

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 300.000

A precisão dos dados apresentados nesta tabela é de 4,9407E-324

TABELA 16 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	≈	≈	+
	$F_2$	-	-	+	+	+	+	+	+
	$F_3$	+	+	+	+	+	+	+	+
	$F_4$	+	+	+	+	+	+	+	+
	$F_5$	-	+	+	+	+	+	-	+
<i>Funções Multimodais Básicas</i>	$F_6$	≈	+	+	+	+	+	+	+
	$F_7$	≈	+	≈	≈	≈	≈	≈	≈
	$F_8$	≈	≈	-	≈	≈	≈	≈	≈
	$F_9$	≈	≈	≈	+	≈	≈	+	+
	$F_{10}$	+	+	+	+	+	+	+	+
	$F_{11}$	+	+	≈	+	+	+	+	+
<i>Funções Multimodais Expandidas</i>	$F_{12}$	+	+	≈	≈	+	+	≈	≈
	$F_{13}$	+	+	+	+	+	+	+	+
<i>Funções multimodais de composição híbrida</i>	$F_{14}$	+	+	+	+	+	+	+	+
	$F_{15}$	≈	≈	+	≈	≈	≈	≈	≈
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	+	+	+	+	+	+	+	+
	$F_{18}$	+	+	+	+	+	+	+	+
	$F_{19}$	+	+	+	+	+	+	+	+
	$F_{20}$	+	+	+	+	+	+	+	+
	$F_{21}$	-	≈	-	+	≈	+	≈	+
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	≈	≈	≈	≈	≈	≈	≈	+
$F_{25}$	≈	+	≈	+	-	+	-	-	
<b>Total</b>	≈	8	6	7	5	6	5	7	4
	-	3	1	2	0	1	0	2	1
	+	14	18	16	20	18	25	16	20

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 17 – RESULTADOS DO TESTE DE MANN-WHITNEY PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
Funções Unimodais	$F_1$	≈	≈	≈	≈	≈	≈	≈	+
	$F_2$	-	-	+	+	+	+	+	+
	$F_3$	≈	+	+	+	+	+	+	+
	$F_4$	-	+	+	+	+	+	-	+
	$F_5$	-	≈	+	+	+	+	-	+
Funções Multimodais Básicas	$F_6$	≈	+	+	+	+	+	+	+
	$F_7$	≈	+	-	≈	+	+	≈	≈
	$F_8$	≈	≈	-	≈	≈	≈	≈	+
	$F_9$	≈	≈	≈	+	≈	≈	+	+
	$F_{10}$	+	+	+	+	+	+	+	+
	$F_{11}$	+	+	+	+	+	+	+	+
	$F_{12}$	+	+	≈	+	+	+	+	+
Funções Multimodais Expandidas	$F_{13}$	-	-	-	+	-	≈	+	+
	$F_{14}$	+	+	+	+	+	+	+	+
Funções multimodais de composição híbrida	$F_{15}$	≈	≈	+	≈	≈	≈	≈	≈
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	+	+	+	+	+	+	+	+
	$F_{18}$	≈	≈	+	+	≈	+	+	-
	$F_{19}$	≈	+	≈	+	≈	+	+	≈
	$F_{20}$	≈	+	+	+	≈	+	+	-
	$F_{21}$	-	≈	-	+	≈	+	≈	+
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	≈	≈	≈	≈	≈	≈	≈	+
	$F_{25}$	+	+	≈	+	≈	+	-	-
<b>Total</b>	≈	11	8	6	4	10	6	6	3
	-	5	2	4	0	1	0	3	3
	+	9	15	15	21	14	19	16	19

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 18 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADe-v1	HSADe-v2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	2,75E-29 ± 7,78E-29	5,70E-30 ± 2,16E-29	5,62E-10 ± 7,86E-10	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	1,07E-23 ± 1,88E-23	7,02E-16 ± 1,83E-15	4,98E-04 ± 5,26E-04	7,94E-01 ± 7,43E-01	9,94E-01 ± 7,02E-01	1,05E+03 ± 3,35E+03	1,18E-08 ± 9,42E-09	9,44E+02 ± 6,14E+02	9,86E-21 ± 1,92E-20	2,19E-14 ± 4,77E-14
$F_3$	2,77E+04 ± 1,45E+04	3,80E+04 ± 1,77E+04	5,15E+05 ± 2,14E+05	1,11E+06 ± 5,08E+05	7,45E+05 ± 3,11E+05	6,64E+06 ± 1,55E+07	4,32E+05 ± 1,75E+05	4,42E+06 ± 1,81E+06	2,59E+04 ± 1,47E+04	3,26E+04 ± 1,23E+04
$F_4$	5,35E-03 ± 2,49E-02	2,33E-01 ± 4,90E-01	4,17E+02 ± 2,73E+02	2,21E+03 ± 1,49E+03	3,31E+02 ± 2,98E+02	6,52E+03 ± 1,01E+04	1,35E-02 ± 1,66E-02	6,87E+03 ± 2,53E+03	1,11E-03 ± 1,61E-03	5,58E-02 ± 9,69E-02
$F_5$	3,32E+02 ± 3,37E+02	1,21E+03 ± 5,29E+02	2,61E+03 ± 5,87E+02	5,22E+03 ± 9,12E+02	2,72E+03 ± 5,92E+02	4,58E+03 ± 1,38E+03	1,03E+03 ± 5,55E+02	3,06E+03 ± 4,81E+02	7,17E+02 ± 4,67E+02	1,31E+03 ± 4,95E+02
$F_6$	7,01E+00 ± 2,71E+01	5,27E+00 ± 1,81E+01	3,20E+01 ± 2,33E+01	1,07E+02 ± 4,15E+01	5,00E+01 ± 2,57E+01	1,93E+00 ± 6,42E+00	9,74E+00 ± 1,11E+01	7,73E+01 ± 5,14E+01	7,92E+00 ± 2,34E+01	2,36E+00 ± 2,63E+00
$F_7$	4,09E-03 ± 7,56E-03	2,66E-03 ± 5,94E-03	1,92E-03 ± 4,60E-03	4,93E-03 ± 7,48E-03	1,38E-03 ± 3,80E-03	7,37E-03 ± 1,30E-02	1,37E-02 ± 1,52E-02	1,19E-02 ± 1,03E-02	3,54E-03 ± 7,22E-03	2,32E-03 ± 5,41E-03
$F_8$	2,11E+01 ± 1,85E-01	2,10E+01 ± 3,16E-01	2,11E+01 ± 5,19E-02	2,11E+01 ± 3,30E-02	2,11E+01 ± 4,45E-02	2,11E+01 ± 4,03E-02	2,11E+01 ± 3,93E-02	2,11E+01 ± 3,85E-02	2,11E+01 ± 3,91E-02	2,11E+01 ± 4,50E-02
$F_9$	9,45E-12 ± 4,25E-12	4,87E-12 ± 2,86E-12	5,33E-01 ± 1,70E+00	1,05E-13 ± 3,60E-13	0,00E+00 ± 0,00E+00	1,79E-01 ± 9,99E-01	2,36E+02 ± 1,07E+02	4,32E+01 ± 1,36E+01	7,11E-17 ± 5,02E-16	2,16E+01 ± 1,87E+00
$F_{10}$	7,23E+01 ± 9,76E+00	7,60E+01 ± 7,83E+00	6,62E+01 ± 1,91E+01	1,63E+02 ± 7,09E+01	1,51E+02 ± 1,91E+01	1,66E+02 ± 3,24E+01	3,43E+02 ± 1,78E+01	3,25E+02 ± 3,26E+01	3,54E+01 ± 7,61E+00	3,11E+01 ± 4,80E+00
$F_{11}$	5,22E+01 ± 1,39E+00	5,16E+01 ± 2,37E+00	2,12E+01 ± 3,81E+00	5,88E+01 ± 3,22E+00	5,61E+01 ± 1,95E+00	6,02E+01 ± 6,74E+00	7,18E+01 ± 1,99E+00	7,14E+01 ± 2,05E+00	2,44E+01 ± 4,31E+00	1,80E+01 ± 4,23E+00
$F_{12}$	8,72E+03 ± 8,18E+03	1,25E+04 ± 1,45E+04	8,68E+03 ± 7,55E+03	1,97E+04 ± 1,20E+04	7,49E+04 ± 3,73E+04	1,20E+05 ± 3,63E+04	3,58E+04 ± 5,32E+04	2,00E+04 ± 1,12E+04	1,36E+04 ± 1,08E+04	1,29E+04 ± 1,06E+04
$F_{13}$	4,12E+00 ± 2,64E-01	4,18E+00 ± 1,70E-01	1,33E+01 ± 2,29E+00	1,10E+01 ± 6,12E-01	4,18E+00 ± 3,29E-01	6,01E+00 ± 4,80E-01	2,76E+01 ± 1,56E+00	9,54E+00 ± 5,92E+00	2,21E+00 ± 3,38E-01	4,34E+00 ± 6,97E-01
$F_{14}$	2,18E+01 ± 2,72E-01	2,18E+01 ± 2,79E-01	2,24E+01 ± 4,68E-01	2,27E+01 ± 1,84E-01	2,28E+01 ± 2,60E-01	2,27E+01 ± 3,33E-01	2,28E+01 ± 2,62E-01	2,29E+01 ± 2,09E-01	2,09E+01 ± 5,91E-01	2,11E+01 ± 4,23E-01

TABELA 18 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{15}$	2,96E+02 ± 9,89E+01	3,07E+02 9,70E+01	3,84E+02 ± 5,48E+01	3,84E+02 ± 5,48E+01	3,15E+02 ± 9,81E+01	3,28E+02 ± 9,70E+01	3,28E+02 ± 8,65E+01	3,22E+02 ± 9,73E+01	3,64E+02 ± 7,76E+01	3,46E+02 ± 8,79E+01
$F_{16}$	6,49E+01 ± 1,66E+01	6,86E+01 ± 5,07E+01	5,53E+01 ± 1,30E+01	6,95E+01 ± 3,35E+01	1,15E+02 ± 1,04E+01	1,22E+02 ± 2,93E+01	2,63E+02 ± 4,71E+01	2,31E+02 ± 4,64E+01	4,30E+01 ± 5,19E+01	2,99E+01 ± 1,24E+01
$F_{17}$	1,30E+02 ± 2,02E+01	1,34E+02 ± 4,60E+01	5,79E+01 ± 9,95E+00	1,53E+02 ± 6,83E+01	2,09E+02 ± 1,89E+01	1,93E+02 ± 4,42E+01	2,95E+02 ± 5,70E+01	2,66E+02 ± 4,02E+01	4,02E+01 ± 7,99E+00	2,94E+01 ± 5,47E+00
$F_{18}$	9,19E+02 ± 5,18E+00	9,22E+02 ± 6,54E+00	9,15E+02 ± 2,83E+00	9,32E+02 ± 4,04E+01	9,15E+02 ± 2,78E+00	9,41E+02 ± 3,82E+01	9,13E+02 ± 6,66E+01	9,16E+02 ± 1,86E+00	9,18E+02 ± 8,58E+00	9,21E+02 ± 1,76E+01
$F_{19}$	9,06E+02 ± 8,75E+01	9,19E+02 ± 1,74E+01	9,15E+02 ± 2,72E+00	9,23E+02 ± 4,76E+01	9,16E+02 ± 3,01E+00	9,48E+02 ± 3,35E+01	9,03E+02 ± 9,15E+01	9,16E+02 ± 1,32E+00	9,20E+02 ± 3,60E+00	9,21E+02 ± 1,78E+01
$F_{20}$	9,06E+02 ± 8,75E+01	9,21E+02 ± 2,60E+00	9,15E+02 ± 2,65E+00	9,22E+02 ± 5,11E+01	9,16E+02 ± 3,23E+00	9,51E+02 ± 2,52E+01	9,20E+02 ± 2,02E+01	9,16E+02 ± 1,32E+00	9,18E+02 ± 7,61E+00	9,22E+02 ± 1,75E+00
$F_{21}$	5,10E+02 ± 7,13E+01	5,00E+02 ± 4,26E-13	7,42E+02 ± 2,55E+02	5,27E+02 ± 1,34E+02	8,01E+02 ± 2,49E+02	5,33E+02 ± 1,40E+02	5,33E+02 ± 1,15E+02	9,45E+02 ± 1,66E+02	5,10E+02 ± 7,12E+01	5,00E+02 ± 4,03E-13
$F_{22}$	8,93E+02 ± 1,12E+01	9,04E+02 ± 1,66E+01	9,25E+02 ± 1,15E+01	9,67E+02 ± 6,33E+00	9,02E+02 ± 8,67E+00	9,72E+02 ± 1,25E+01	9,40E+02 ± 2,36E+01	9,35E+02 ± 6,13E+00	8,57E+02 ± 1,78E+01	8,79E+02 ± 1,17E+01
$F_{23}$	5,53E+02 ± 7,38E+01	5,39E+02 ± 6,26E-03	8,13E+02 ± 2,36E+02	5,65E+02 ± 1,29E+02	9,26E+02 ± 1,83E+02	5,57E+02 ± 7,33E+01	5,99E+02 ± 1,37E+02	1,00E+03 ± 6,70E+01	5,58E+02 ± 9,32E+01	5,39E+02 ± 1,42E-01
$F_{24}$	2,00E+02 ± 5,74E-14	2,00E+02 ± 6,46E-13	2,00E+02 ± 5,74E-14	2,00E+02 ± 1,32E-12	2,00E+02 ± 1,54E-12	4,13E+02 ± 4,05E+02	2,16E+02 ± 1,16E+02	2,16E+02 ± 1,12E+02	2,00E+02 ± 5,74E-14	2,00E+02 ± 1,31E-12
$F_{25}$	2,16E+02 ± 1,21E+00	2,16E+02 ± 1,44E+00	2,15E+02 ± 1,07E+00	2,17E+02 ± 1,17E+00	2,16E+02 ± 1,25E+00	2,46E+02 ± 1,98E+01	2,16E+02 ± 1,77E+00	2,19E+02 ± 1,76E+00	2,16E+02 ± 1,43E+00	2,15E+02 ± 9,05E-01

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 500.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324

TABELA 19 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	+	+	+
	$F_2$	-	+	+	+	+	-	+	+
	$F_3$	≈	+	+	+	+	+	+	+
	$F_4$	≈	+	+	+	+	+	+	+
	$F_5$	-	+	+	+	+	+	+	+
<i>Funções Multimodais Básicas</i>	$F_6$	-	+	+	+	+	-	+	+
	$F_7$	≈	+	+	+	+	+	+	+
	$F_8$	≈	≈	≈	≈	≈	≈	≈	≈
	$F_9$	+	+	+	+	≈	+	+	+
	$F_{10}$	+	+	+	+	+	+	+	+
	$F_{11}$	+	+	-	+	+	+	+	+
	$F_{12}$	-	≈	-	+	+	+	+	+
<i>Funções Multimodais Expandidas</i>	$F_{13}$	+	+	+	+	+	+	+	+
	$F_{14}$	+	+	+	+	+	+	+	+
<i>Funções multimodais de composição híbrida</i>	$F_{15}$	-	-	≈	≈	-	≈	≈	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	+	+	+	+	+	+	+	+
	$F_{18}$	≈	+	-	+	-	+	+	-
	$F_{19}$	-	+	-	+	-	+	+	-
	$F_{20}$	≈	+	-	+	-	+	+	-
	$F_{21}$	≈	≈	+	+	+	+	+	+
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	≈	≈	≈	+	+	+	+	+
$F_{25}$	≈	≈	≈	+	≈	+	≈	+	
<b>Total</b>	≈	10	6	5	3	4	2	3	1
	-	6	1	5	0	4	2	0	3
	+	9	18	15	22	17	21	22	21

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 20 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	+	+	+
	$F_2$	-	-	+	+	+	-	+	+
	$F_3$	-	≈	+	+	+	+	+	+
	$F_4$	-	+	+	+	+	+	-	+
	$F_5$	-	≈	+	+	+	+	-	+
<i>Funções Multimodais Básicas</i>	$F_6$	-	-	+	+	+	-	+	+
	$F_7$	≈	+	+	+	+	+	+	+
	$F_8$	≈	≈	≈	≈	≈	≈	≈	≈
	$F_9$	-	-	-	-	-	-	+	+
	$F_{10}$	+	+	+	+	+	+	+	+
	$F_{11}$	+	+	+	+	+	+	+	+
	$F_{12}$	≈	≈	≈	+	+	+	+	+
<i>Funções Multimodais Expandidas</i>	$F_{13}$	-	≈	+	+	≈	+	+	+
	$F_{14}$	+	+	+	+	+	+	+	+
<i>Funções multimodais de composição híbrida</i>	$F_{15}$	-	-	+	+	-	≈	≈	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	+	+	+	+	+	+	+	+
	$F_{18}$	-	-	-	+	-	+	≈	-
	$F_{19}$	-	-	-	+	-	+	≈	-
	$F_{20}$	-	-	-	+	-	+	+	-
	$F_{21}$	≈	≈	+	+	+	+	+	+
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	-	-	-	≈	+	+	+	+
	$F_{25}$	+	+	+	+	+	+	+	+
<b>Total</b>	≈	5	7	3	3	3	2	4	1
	-	12	8	5	1	5	3	2	3
	+	8	10	17	21	17	20	19	21

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.



## 6.4 ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2011

Um conjunto de 22 problemas reais de otimização propostos no *Competition on Testing Evolutionary Algorithms on Real-world Numerical Optimization Problems* do CEC2011 foi escolhido como o segundo estudo de caso para a verificação do desempenho das duas versões do algoritmo *HSADE* apresentadas nesta dissertação.

Este estudo de caso foi escolhido, pois, diferentemente das funções propostas no CEC2005 e CEC2013, trata-se de problemas de otimização do mundo real especificamente escolhidos para avaliar o desempenho de algoritmos de otimização estocásticos.

A definição de cada um destes problemas (formulação matemática e restrições) é descrita em detalhes em Swagatam e Suganthan (2010) e não será apresentada nesta dissertação. Apenas a dimensão  $D$  de cada um dos problemas proposto no CEC2011 é apresentada na TABELA 21.

TABELA 21 – DIMENSÃO DOS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011

<b>Problema</b>	$P_1$	$P_2$	$P_3$	$P_4$	$P_5$	$P_6$	$P_7$	$P_8$	$P_9$	$P_{10}$	$P_{11}$
<i>Dimensão</i>	6	30	1	1	30	30	20	7	126	12	120
<b>Problema</b>	$P_{12}$	$P_{13}$	$P_{14}$	$P_{15}$	$P_{16}$	$P_{17}$	$P_{18}$	$P_{19}$	$P_{20}$	$P_{21}$	$P_{22}$
<i>Dimensão</i>	216	6	13	15	40	140	96	96	96	26	22

FONTE: O autor (2014)

Para reduzir o impacto da natureza estocástica dos algoritmos, cada algoritmo foi executado 25 vezes para cada problema. Além disto, cada experimento foi finalizado após um número máximo de 150000 avaliações da função objetivo (SWAGATAM; SUGANTHAN, 2010).

### 6.4.1 Definição dos parâmetros

Com relação aos parâmetros utilizados pelos algoritmos propostos nesta dissertação, a mesma configuração de parâmetros foi utilizada na resolução de todos os problemas de otimização deste estudo de caso com o intuito de obter uma comparação justa.

Os valores de parâmetros de cada versão do algoritmo *HSADE*, utilizados para este estudo de caso são os mesmos definidos na seção 6.3.1, exceto pelo tamanho da população. Para as duas versões *HSADE-v1* e *HSADE-v2*, o tamanho da população foi escolhido como  $NP = 120$  em todos os problemas de otimização deste estudo de caso.

Com exceção ao tamanho da população, para os algoritmos JADE com e sem arquivo, jDE, SaDE, EPSDE, DEGL, DEahcSPX e CoDE, todos os outros parâmetros foram definidos segundo a orientação dos autores dos mesmos. Com relação ao tamanho da população, este foi definido como  $NP = 120$  para JADE com e sem arquivo, jDE, SaDE, EPSDE, DEGL, DEahcSPX e  $NP = 40$  para CoDE. O valor de  $NP = 40$  para CoDE foi escolhido devido ao fato de que este algoritmo calcula três vetores experimentais para cada vetor alvo, o que seria equivalente a uma população de 120 indivíduos quando comparado aos outros algoritmos.

#### 6.4.2 Resultados experimentais e discussão

A TABELA 26 apresenta a média e o desvio padrão do valor da função objetivo calculados sobre 25 experimentos para cada um dos 22 problemas do CEC2011 obtidos para cada um dos algoritmos: JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, *HSADE-v1* e *HSADE-v2*.

As TABELAS 27 e 28 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando *HSADE-v1* e *HSADE-v2* como algoritmo de controle respectivamente. Em todas estas tabelas, os sinais "+", "≈" e "-" indicam se o algoritmo de controle apresentou um desempenho melhor, semelhante ou pior, respectivamente, do que o algoritmo de comparação para o problema correspondente. No final de cada tabela é apresentado o número total de problemas nos quais o algoritmo de controle apresentou um desempenho melhor (+), similar (≈) ou pior (-) quando comparado com cada um dos oito algoritmos estado da arte baseados na evolução diferencial.

Nas seções subsequentes, uma análise mais detalhada dos resultados experimentais obtidos na solução dos problemas reais propostos no CEC2011 será apresentada para cada uma das duas versões do algoritmo *HSADE*.

#### 6.4.2.1 Análise do desempenho do algoritmo HSADE-v1

Nesta seção uma análise dos resultados obtidos no teste de teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v1 como algoritmo de controle é realizada.

##### 6.4.2.1.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados na TABELA 27.

Nota-se que nenhum dos quatro algoritmos JADE *com* e *sem arquivo*, EPSDE e DEahcSPX apresentaram um resultado melhor do HSADE-v1 em nenhum dos vinte e dois problemas do CEC2011. Em especial verifica-se que HSADE-v1 teve desempenho superior a EPSDE e DEahcSPX em vinte e um problemas e desempenho superior a JADE *com* e *sem arquivo* em 10 problemas.

Já os algoritmos jDE e DEGL superaram HSADE-v1 em apenas um problema ( $P_{15}$ ). Em todos os outros vinte e um problemas, o algoritmo HSADE-v1 apresentou um desempenho superior ou similar.

Com relação ao algoritmo SaDE, este apresentou desempenho superior a HSADE-v1 em apenas três problemas ( $P_9$ ,  $P_{15}$  e  $P_{16}$ ). Em todos os outros dezenove problemas, o algoritmo HSADE-v1 apresentou um desempenho superior ou similar.

O algoritmo CoDE foi o único algoritmo que conseguiu superar HSADE-v1 em um maior número de problemas ( $P_1$ ,  $P_7$ ,  $P_{10}$ ,  $P_{15}$ ,  $P_{16}$  e  $P_{22}$ ). Por outro lado, HSADE-v1 conseguiu superar algoritmo CoDE em nove problemas.

Em resumo, analisando os dados de uma maneira geral, o algoritmo HSADE-v1 apresentou um bom desempenho na solução dos problemas de otimização propostos no CEC2011.

##### 6.4.2.1.2 Teste *post hoc* de *Friedman*

A TABELA 22 apresenta a classificação média dos algoritmos considerados e o valor-p obtido através do teste de *Friedman*. O baixo valor-p obtido comprova a existência de diferenças significativas no desempenho entre os algoritmos

comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada com um nível de significância de  $\alpha = 1E-14$ .

Pode-se observar também através dos dados da TABELA 22, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v1 foi o que apresentou o melhor resultado.

TABELA 22 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011

JADE <i>com arquivo</i>	3,77
JADE <i>sem arquivo</i>	4,00
CoDE	3,73
SaDE	4,23
jDE	5,77
EPSDE	7,14
DEGL	5,82
DEahcSPX	8,27
HSADE-v1	2,27

Valor-p	1,08E-15
---------	----------

FONTE: O autor (2014)

Uma vez calculada a classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. A TABELA 23 apresenta os valores-p e seus respectivos valores-p ajustados através do procedimento de Holm considerando o algoritmo HSADE-v1 como algoritmo de controle.

Com pode-se ser verificado na TABELA 23, o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v1 sobre DEahcSPX, EPSDE, DEGL e jDE considerando-se um nível de significância  $\alpha = 0,001$ . Já para os algoritmos JADE *sem arquivo* e SaDE, HSADE-v1 pode ser considerado melhor para um nível de significância  $\alpha = 0,11$ . Para os outros dois algoritmos, CoDE e JADE *com arquivo*, o teste *post hoc* de *Friedman* não detectou uma diferença significativa quando comparado a HSADE-v1.

TABELA 23 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	3,69E-13	2,95E-12
EPSDE	3,86E-09	2,70E-08
DEGL	1,76E-05	1,05E-04
jDE	2,25E-05	1,12E-04
SaDE	0,018	0,072
JADE <i>sem arquivo</i>	0,036	0,109
JADE <i>com arquivo</i>	0,069	0,139
CoDE	0,078	0,139

FONTE: O autor (2014)

#### 6.4.2.2 Análise do desempenho do algoritmo HSADE-v2

Nesta seção uma análise dos resultados obtidos no teste de teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v2 como algoritmo de controle é realizada.

##### 6.4.2.2.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados na TABELA 28.

Nota-se que nenhum dos seis algoritmos JADE *com* e *sem arquivo*, jDE, EPSDE, DEGL e DEahcSPX apresentaram um resultado melhor do HSADE-v2 em nenhum dos vinte e dois problemas do CEC2011. Quando comparado a estes seis algoritmos, o algoritmo HSADE-v2 apresentou um desempenho superior ou similar em todos os vinte e dois problemas de otimização.

Apenas os algoritmos SaDE e CoDE que superaram HSADE-v2 em apenas alguns problemas. No caso do algoritmo SaDE, HSADE-v2 foi superado em apenas dois problemas ( $P_9$  e  $P_{10}$ ). Já o algoritmo CoDE conseguiu superar HSADE-v2 em seis problemas ( $P_1$ ,  $P_6$ ,  $P_7$ ,  $P_{10}$ ,  $P_{21}$  e  $P_{22}$ ). Por outro lado, HSADE-v2 conseguiu superar algoritmo SaDE e CoDE em dezesseis e dez problemas respectivamente.

De uma maneira global, os resultados obtidos teste de *Mann-Whitney* evidenciam o excelente desempenho do algoritmo HSADE-v2.

#### 6.4.2.2.2 Teste *post hoc* de *Friedman*

A TABELA 24 apresenta a classificação média dos algoritmos considerados e o valor-p obtido através do teste de *Friedman*. O baixo valor-p obtido comprova a existência de diferenças significativas no desempenho entre os algoritmos comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada com um nível de significância de  $\alpha = 1E-16$ .

Pode-se observar também através dos dados da TABELA 24, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v2 foi o que apresentou o melhor resultado.

TABELA 24 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011

JADE <i>com arquivo</i>	3,77
JADE <i>sem arquivo</i>	4,00
CoDE	3,68
SaDE	4,41
jDE	5,77
EPSDE	7,18
DEGL	5,86
DEahcSPX	8,32
HSADE-v2	2,00

<i>Valor-p</i>	6,03E-17
----------------	----------

FONTE: O autor (2014)

Uma vez calculada a classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. A TABELA 25 apresenta os valores-p e seus respectivos valores-p ajustados através do procedimento de Holm considerando o algoritmo HSADE-v2 como algoritmo de controle.

Com pode-se ser verificado na TABELA 25, o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v2 sobre todos os oito algoritmos considerando-se um nível de significância  $\alpha = 0,07$  confirmando também

o excelente desempenho do algoritmo HSADE-v2 na solução dos problemas de otimização do CEC2011.

TABELA 25 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	1,98E-14	1,58E-13
EPSDE	3,49E-10	2,44E-09
DEGL	2,88E-06	1,73E-05
jDE	4,90E-06	2,45E-05
SaDE	0,004	0,014
JADE <i>sem arquivo</i>	0,015	0,046
JADE <i>com arquivo</i>	0,032	0,064
CoDE	0,042	0,064

FONTE: O autor (2014)

TABELA 26 – RESULTADOS OBTIDOS PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$P_1$	3,84E-01 ± 4,74E-01	5,94E-01 ± 9,50E-01	4,07E-01 ± 2,03E+00	7,36E-01 ± 9,69E-01	6,52E-01 ± 9,18E-01	2,64E+00 ± 2,12E+00	4,06E+00 ± 5,55E+00	3,96E+00 ± 5,21E+00	2,78E-04 ± 6,98E-04	5,80E-03 ± 1,11E-02
$P_2$	-2,40E+01 ± 5,33E-01	-2,39E+01 ± 6,11E-01	-2,49E+01 ± 2,08E+00	-2,22E+01 ± 5,54E-01	-2,24E+01 ± 8,56E-01	-2,24E+01 ± 7,87E-01	-2,55E+01 ± 2,31E+00	-1,19E+01 ± 1,61E+00	-2,70E+01 ± 9,01E-01	-2,60E+01 ± 1,02E+00
$P_3$	1,15E-05 ± 1,69E-19	1,15E-05 ± 2,50E-19	1,15E-05 ± 1,78E-19	1,15E-05 ± 1,35E-19	1,15E-05 ± 1,55E-19	1,15E-05 ± 1,26E-19	1,15E-05 ± 1,63E-19	1,15E-05 ± 1,63E-19	1,15E-05 ± 1,24E-19	1,15E-05 ± 1,23E-19
$P_4$	1,44E+01 ± 1,40E+00	1,48E+01 ± 1,86E+00	1,77E+01 ± 3,35E+00	1,44E+01 ± 1,42E+00	1,60E+01 ± 3,10E+00	1,49E+01 ± 2,28E+00	1,54E+01 ± 2,85E+00	1,70E+01 ± 3,29E+00	1,39E+01 ± 2,07E-01	1,38E+01 ± 1,85E-01
$P_5$	-3,51E+01 ± 5,94E-01	-3,48E+01 ± 6,63E-01	-3,50E+01 ± 1,14E+00	-3,40E+01 ± 5,02E-01	-3,40E+01 ± 4,63E-01	-3,39E+01 ± 5,48E-01	-3,32E+01 ± 2,96E+00	-2,28E+01 ± 1,87E+00	-3,58E+01 ± 7,73E-01	-3,49E+01 ± 7,12E-01
$P_6$	-2,88E+01 ± 4,84E-01	-2,85E+01 ± 6,07E-01	-2,87E+01 ± 7,69E-01	-2,82E+01 ± 6,25E-01	-2,66E+01 ± 1,28E+00	-2,57E+01 ± 2,42E+00	-2,55E+01 ± 2,85E+00	-1,62E+01 ± 1,46E+00	-2,86E+01 ± 1,71E+00	-2,88E+01 ± 4,74E-01
$P_7$	1,15E+00 ± 9,45E-02	1,19E+00 ± 8,72E-02	7,19E-01 ± 1,00E-01	1,35E+00 ± 1,02E-01	1,34E+00 ± 8,16E-02	1,28E+00 ± 8,03E-02	1,14E+00 ± 2,90E-01	1,71E+00 ± 1,18E-01	8,72E-01 ± 1,15E-01	8,62E-01 ± 1,55E-01
$P_8$	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00	2,20E+02 ± 0,00E+00
$P_9$	2,13E+03 ± 3,29E+02	2,01E+03 ± 6,16E+02	2,23E+03 ± 5,50E+02	1,28E+03 ± 5,92E+02	2,44E+03 ± 9,75E+02	5,94E+04 ± 6,56E+03	3,09E+03 ± 1,22E+03	1,23E+06 ± 7,25E+04	1,96E+03 ± 3,54E+02	2,28E+03 ± 4,15E+02
$P_{10}$	-2,15E+01 ± 1,72E-01	-2,15E+01 ± 1,43E-01	-2,17E+01 ± 1,22E-01	-2,16E+01 ± 1,57E-01	-2,16E+01 ± 1,44E-01	-2,11E+01 ± 7,65E-01	-2,00E+01 ± 3,56E+00	-2,02E+01 ± 8,09E-01	-2,16E+01 ± 1,16E-01	-2,15E+01 ± 1,65E-01
$P_{11}$	5,23E+04 ± 5,87E+02	5,22E+04 ± 5,31E+02	5,20E+04 ± 5,71E+02	5,23E+04 ± 5,15E+02	6,60E+04 ± 5,53E+03	7,50E+04 ± 3,34E+03	6,06E+04 ± 8,92E+03	1,50E+06 ± 2,77E+05	5,22E+04 ± 6,02E+02	5,21E+04 ± 5,75E+02
$P_{12}$	1,07E+06 ± 1,70E+03	1,07E+06 ± 1,86E+03	1,07E+06 ± 1,58E+03	1,10E+06 ± 3,89E+04	1,07E+06 ± 2,73E+03	1,09E+06 ± 5,96E+03	1,09E+06 ± 1,55E+04	3,80E+06 ± 2,31E+05	1,07E+06 ± 1,59E+03	1,07E+06 ± 1,18E+03
$P_{13}$	1,54E+04 ± 1,33E+00	1,54E+04 ± 1,07E-05	1,54E+04 ± 6,25E+00	1,54E+04 ± 2,15E-06	1,54E+04 ± 1,22E+00	1,54E+04 ± 9,89E-02	1,55E+04 ± 1,36E+01	1,54E+04 ± 2,42E-04	1,54E+04 ± 7,33E-01	1,54E+04 ± 7,30E-07
$P_{14}$	1,82E+04 ± 3,25E+02	1,82E+04 ± 2,50E+02	1,82E+04 ± 6,61E+01	1,81E+04 ± 2,82E+01	1,82E+04 ± 5,63E+01	1,89E+04 ± 2,79E+02	1,81E+04 ± 5,77E+01	1,88E+04 ± 1,06E+02	1,82E+04 ± 2,42E+02	1,81E+04 ± 3,02E+01



TABELA 26 – RESULTADOS OBTIDOS PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADe-v1	HSADe-v2
$P_{15}$	3,29E+04 ± 1,11E+02	3,30E+04 ± 6,23E+01	3,28E+04 ± 2,21E+01	3,28E+04 ± 1,50E+01	3,28E+04 ± 3,89E+01	3,30E+04 ± 2,70E+01	3,28E+04 ± 3,45E+01	3,30E+04 ± 3,68E+01	3,29E+04 ± 5,89E+01	3,28E+04 ± 9,66E+01
$P_{16}$	1,32E+05 ± 4,68E+03	1,31E+05 ± 3,63E+03	1,29E+05 ± 1,21E+03	1,29E+05 ± 8,75E+02	1,33E+05 ± 2,27E+03	1,37E+05 ± 1,99E+03	1,30E+05 ± 1,50E+03	1,37E+05 ± 1,82E+03	1,31E+05 ± 3,66E+03	1,27E+05 ± 2,56E+03
$P_{17}$	1,90E+06 ± 2,70E+04	1,91E+06 ± 2,99E+04	1,91E+06 ± 1,23E+04	1,91E+06 ± 8,76E+03	2,13E+06 ± 1,26E+05	2,22E+06 ± 2,08E+05	1,99E+06 ± 4,93E+04	2,44E+06 ± 3,86E+05	1,90E+06 ± 2,76E+04	1,88E+06 ± 1,07E+04
$P_{18}$	9,38E+05 ± 3,02E+03	9,39E+05 ± 2,69E+03	9,40E+05 ± 1,79E+03	9,39E+05 ± 1,81E+03	1,01E+06 ± 3,97E+04	1,71E+06 ± 2,05E+05	1,02E+06 ± 3,43E+04	1,83E+07 ± 2,95E+06	9,38E+05 ± 4,01E+03	9,37E+05 ± 2,03E+03
$P_{19}$	9,74E+05 ± 1,10E+05	9,62E+05 ± 8,13E+04	1,04E+06 ± 8,40E+04	1,00E+06 ± 3,93E+04	1,41E+06 ± 1,49E+05	2,54E+06 ± 2,34E+05	1,12E+06 ± 7,44E+04	1,90E+07 ± 3,86E+06	9,56E+05 ± 6,97E+04	9,42E+05 ± 1,58E+03
$P_{20}$	9,38E+05 ± 3,02E+03	9,39E+05 ± 2,69E+03	9,40E+05 ± 1,79E+03	9,39E+05 ± 1,81E+03	1,01E+06 ± 3,97E+04	1,71E+06 ± 2,05E+05	1,10E+06 ± 2,97E+05	1,83E+07 ± 2,95E+06	9,38E+05 ± 4,01E+03	9,37E+05 ± 2,03E+03
$P_{21}$	1,75E+01 ± 1,25E+00	1,72E+01 ± 1,66E+00	1,40E+01 ± 2,30E+00	1,80E+01 ± 1,14E+00	1,90E+01 ± 1,76E+00	1,84E+01 ± 1,77E+00	1,50E+01 ± 2,55E+00	2,85E+01 ± 2,65E+00	1,33E+01 ± 2,39E+00	1,61E+01 ± 1,41E+00
$P_{22}$	1,73E+01 ± 1,86E+00	1,74E+01 ± 2,44E+00	1,21E+01 ± 3,05E+00	1,97E+01 ± 2,15E+00	1,60E+01 ± 2,31E+00	1,82E+01 ± 1,89E+00	1,88E+01 ± 2,88E+00	2,08E+01 ± 4,93E+00	1,36E+01 ± 2,03E+00	1,45E+01 ± 2,18E+00

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do valor da função objetivo calculados sobre 25 experimentos, sendo o valor da função objetivo de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 150.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324.

TABELA 27 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	
$P_1$	+	+	-	≈	+	+	≈	+	
$P_2$	+	+	+	+	+	+	+	+	
$P_3$	+	+	+	+	+	+	+	+	
$P_4$	+	+	+	+	+	+	+	+	
$P_5$	+	+	+	+	+	+	+	+	
$P_6$	+	+	≈	+	+	+	+	+	
$P_7$	+	+	-	+	+	+	+	+	
$P_8$	≈	≈	≈	≈	≈	≈	≈	≈	
$P_9$	≈	≈	≈	-	+	+	+	+	
$P_{10}$	≈	≈	-	≈	≈	+	≈	+	
$P_{11}$	≈	≈	≈	≈	+	+	+	+	
$P_{12}$	+	≈	≈	+	+	+	+	+	
$P_{13}$	≈	≈	≈	≈	≈	+	≈	+	
$P_{14}$	≈	≈	+	≈	+	+	≈	+	
$P_{15}$	≈	+	-	-	-	+	-	+	
$P_{16}$	≈	≈	-	-	≈	+	≈	+	
$P_{17}$	≈	≈	+	+	+	+	+	+	
$P_{18}$	≈	≈	+	≈	+	+	+	+	
$P_{19}$	≈	≈	+	+	+	+	+	+	
$P_{20}$	≈	≈	+	≈	+	+	+	+	
$P_{21}$	+	+	≈	+	+	+	+	+	
$P_{22}$	+	+	-	+	+	+	+	+	
<b>Total</b>	≈	12	12	7	8	4	1	6	1
	-	0	0	6	3	1	0	1	0
	+	10	10	9	11	17	21	15	21

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para o problema de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 28 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2011 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	
$P_1$	+	+	-	≈	+	+	≈	+	
$P_2$	+	+	≈	+	+	+	≈	+	
$P_3$	+	+	+	+	+	+	+	+	
$P_4$	+	+	+	+	+	+	+	+	
$P_5$	≈	≈	≈	+	+	+	+	+	
$P_6$	+	+	-	+	+	+	+	+	
$P_7$	+	+	-	+	+	+	+	+	
$P_8$	≈	≈	≈	≈	≈	≈	≈	≈	
$P_9$	≈	≈	≈	-	≈	+	+	+	
$P_{10}$	≈	≈	-	-	≈	+	≈	+	
$P_{11}$	≈	≈	≈	≈	+	+	+	+	
$P_{12}$	+	≈	+	+	+	+	+	+	
$P_{13}$	+	+	+	+	+	+	+	+	
$P_{14}$	≈	≈	+	≈	+	+	+	+	
$P_{15}$	+	+	≈	+	+	+	≈	+	
$P_{16}$	+	+	+	+	+	+	+	+	
$P_{17}$	+	+	+	+	+	+	+	+	
$P_{18}$	≈	+	+	+	+	+	+	+	
$P_{19}$	≈	≈	+	+	+	+	+	+	
$P_{20}$	≈	+	+	+	+	+	+	+	
$P_{21}$	+	+	-	+	+	+	≈	+	
$P_{22}$	+	+	-	+	+	+	+	+	
<b>Total</b>	≈	10	8	6	4	3	1	6	1
	-	0	0	6	2	0	0	0	0
	+	12	14	10	16	19	21	16	21

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para o problema de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

## 6.5 ESTUDO DE CASO: CONJUNTO DE FUNÇÕES CEC2013

Um conjunto de 28 funções testes propostas no *Special Session & Competition on Real-Parameter Single Objective Optimization* do CEC2013 foi escolhido como o terceiro estudo de caso para a verificação do desempenho das duas versões do algoritmo *HSADE* apresentadas nesta dissertação.

As características deste conjunto de funções testes podem ser brevemente resumidas como se segue:

- $F_1$  a  $F_5$  funções unimodais;
- $F_6$  a  $F_{20}$  funções multimodais básicas;
- $F_{21}$  a  $F_{28}$  funções multimodais compostas.

A formulação matemática de cada uma destas funções, bem como suas respectivas propriedades, são descritas em Liang et al. (2013) e não serão apresentadas nesta dissertação.

Para este estudo de caso, todas as funções foram avaliadas nas dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ .

Para reduzir o impacto da natureza estocástica dos algoritmos, cada algoritmo foi executado 50 vezes para cada função teste. Além disto, cada experimento foi finalizado após um número máximo de  $D \cdot 10000$  (100000 para  $D = 10$ , 300000 para  $D = 30$  e 500000 para  $D = 50$ ) avaliações da função objetivo (LIANG et al., 2013).

### 6.5.1 Definição dos parâmetros

Com relação aos parâmetros utilizados pelos algoritmos propostos nesta dissertação, a mesma configuração de parâmetros foi utilizada na resolução de todos os problemas de otimização deste estudo de caso com o intuito de obter uma comparação justa.

Os valores de parâmetros utilizados neste estudo de caso para cada versão do algoritmo *HSADE* e para os oito algoritmos de comparação são os mesmos definidos na seção 6.3.1

## 6.5.2 Resultados experimentais e discussão

As TABELAS 37, 40 e 43 apresentam a média e o desvio padrão do erro da função objetivo calculados sobre 50 experimentos para cada uma das 28 funções testes do CEC2013 para as dimensões 10, 30 e 50, respectivamente, obtidos para cada um dos algoritmos: JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, HSADE-v1 e HSADE-v2.

As TABELAS 38, 41 e 44 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando HSADE-v1 como algoritmo de controle enquanto as TABELAS 39, 42 e 45 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando HSADE-v2 como algoritmo de controle. Em todas estas tabelas, os sinais "+", "≈" e "-" indicam se o algoritmo de controle apresentou um desempenho melhor, semelhante ou pior, respectivamente, do que o algoritmo de comparação para a função teste correspondente. No final de cada tabela é apresentado o número total de funções testes nas quais o algoritmo de controle apresentou um desempenho melhor (+), similar (≈) ou pior (-) quando comparado com cada um dos oito algoritmos estado da arte baseados na evolução diferencial.

Nas seções subseqüentes, uma análise mais detalhada dos resultados experimentais obtidos na solução das funções testes propostas no CEC2013 será apresentada para cada um dos algoritmos HSADE-v1 e HSADE-v2.

### 6.5.2.1 Análise do desempenho do algoritmo HSADE-v1

Nesta seção, uma análise dos resultados obtidos no teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v1 como algoritmo de controle é realizada.

#### 6.5.2.1.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados nas TABELAS 38, 41 e 44. Uma análise dos resultados obtidos para cada grupo de função teste é detalhada na seqüência.

- Funções unimodais

O algoritmo HSADE-v1 apresentou um desempenho excelente na solução das funções unimodais. Nenhum algoritmo apresentou um desempenho melhor que HSADE-v1 em todas as cinco funções para o caso de dimensão 30 e 50.

Para o caso de dimensão 10, HSADE-v1 foi superado apenas por JADE *com arquivo* e DEGL nas funções  $F_2$  e  $F_4$ . Em todos os outros casos HSADE-v1 foi superior ou apresentou desempenho semelhante.

- Funções multimodais básicas

Em geral, o algoritmo HSADE-v1 teve um desempenho muito bom na solução das quinze funções multimodais básicas.

Das três dimensões consideradas, o caso de dimensão 10 foi aquele no qual HSADE-v1 apresentou o seu pior desempenho. Entretanto, mesmo assim, HSADE-v1 teve um desempenho melhor, ou pelo menos semelhante em no mínimo dez funções quando comparado a cada uma das oito variantes DE.

Para o caso de 30 e 50 dimensões, HSADE-v1 apresentou um desempenho excepcional. No caso de dimensão 30, HSADE-v1 foi superado em apenas duas funções por JADE *com arquivo* ( $F_6$ ,  $F_{14}$ ), EPSDE ( $F_{14}$ ,  $F_{17}$ ), DEGL ( $F_9$ ,  $F_{10}$ ) e superado em apenas uma função por JADE *sem arquivo* ( $F_{14}$ ), CoDE ( $F_8$ ), jDE ( $F_{14}$ ) e DEahcSPX ( $F_{10}$ ). SaDE não superou HSADE-v1 em nenhuma função. Para o caso de dimensão 50, HSADE-v1 foi superado em apenas em três funções por JADE *sem arquivo* ( $F_8$ ,  $F_{14}$ ,  $F_{17}$ ), superado somente em duas funções por JADE *com arquivo* ( $F_{14}$ ,  $F_{17}$ ) e superado em apenas uma função por jDE ( $F_{11}$ ) e EPSDE ( $F_{17}$ ). CoDE, SaDE, DEGL e DEahcSPX não superaram HSADE-v1 em nenhuma função.

- Funções multimodais compostas

Em geral, o algoritmo HSADE-v1 teve um desempenho melhor, ou pelo menos comparável, do que as oito variantes DE para as oito funções testes multimodais compostas.

Como observado no caso das funções multimodais básicas, das três dimensões consideradas, o caso de dimensão 10 foi aquele no qual HSADE-v1 apresentou o seu pior desempenho. HSADE-v1 apresentou desempenho semelhante ou melhor do que JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX em sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), sete ( $F_{22}$  a  $F_{28}$ ), três ( $F_{23}, F_{27}, F_{28}$ ), sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), seis ( $F_{21}, F_{23}$  a  $F_{26}, F_{28}$ ), cinco ( $F_{21}$  a  $F_{23}, F_{26}, F_{28}$ ), cinco ( $F_{22}$  a  $F_{24}, F_{26}, F_{28}$ ) funções, respectivamente. Além disso, em grande parte destas funções, o desempenho apresentado foi semelhante e não superior.

Já para o caso de dimensão 30, HSADE-v1 apresentou um desempenho muito bom. HSADE-v1 foi superado na função  $F_{21}$  apenas por JADE com e sem arquivo, CoDE e jDE, superado na função  $F_{22}$  somente por JADE com e sem arquivo e na função  $F_{24}$  apenas por JADE *sem arquivo*. Em todos os outros casos HSADE-v1 foi superior ou apresentou desempenho semelhante. Em especial nota-se que SaDE, EPSDE, DEGL e DEahcSPX não superaram HSADE-v1 em nenhuma função.

Para o caso de dimensão 50, HSADE-v1 também apresentou um desempenho muito bom. HSADE-v1 apresentou desempenho semelhante ou melhor do que JADE com e sem arquivo, CoDE, SaDE, jDE, DEGL e DEahcSPX em cinco ( $F_{22}, F_{23}, F_{25}, F_{26}, F_{27}$ ), cinco ( $F_{22}, F_{23}, F_{24}, F_{26}, F_{28}$ ), cinco ( $F_{21}$  a  $F_{25}$ ), cinco ( $F_{22}$  a  $F_{25}, F_{27}$ ), seis ( $F_{21}$  a  $F_{24}, F_{26}, F_{28}$ ), sete ( $F_{22}$  a  $F_{28}$ ) funções, respectivamente. Entretanto, na maioria destas funções, o desempenho apresentado foi superior. No caso do algoritmo EPSDE, HSADE-v1 obteve desempenho superior ou semelhante em todas as funções.

#### 6.5.2.1.2 Teste *post hoc* de *Friedman*

A TABELA 29 apresenta a classificação média dos algoritmos considerados e o valor-p obtidos através do teste de *Friedman* para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ .

Os baixos valores-p obtidos demonstram a existência de diferenças significativas no desempenho entre os algoritmos comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada para as três dimensões avaliadas com um nível de significância de  $\alpha = 1E-06$ .

Pode-se observar também através dos dados da TABELA 29, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v1 foi o que apresentou o melhor resultado nas dimensões 30 e 50.

TABELA 29 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2013

	<i>D</i> = 10	<i>D</i> = 30	<i>D</i> = 50
JADE <i>com arquivo</i>	4,25	3,91	3,64
JADE <i>sem arquivo</i>	4,36	3,73	3,64
CoDE	7,64	4,27	3,84
SaDE	3,61	6,34	5,64
jDE	5,04	5,16	4,68
EPSDE	4,63	6,63	6,29
DEGL	5,23	5,95	6,68
DEahcSPX	5,79	6,21	7,75
HSADE-v1	4,46	2,80	2,84

<i>Valor-p</i>	6,94E-07	5,13E-10	5,50E-15
----------------	----------	----------	----------

FONTE: O autor (2014)

Uma vez calculada a classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. As TABELAS 30, 31 e 32 apresentam os valores-p e seus respectivo valores-p ajustados através do procedimento de Holm para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$  respectivamente considerando o algoritmo HSADE-v1 como algoritmo de controle.

Como já havia sido verificado no teste de *Mann-Whitney*, o algoritmo HSADE-v1 não apresentou um bom resultado para a dimensão 10. Como se pode observar na TABELA 30, o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v1 apenas sobre CoDE. Para todos os outros algoritmos, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v1.

Para as dimensões 30 e 50, verifica-se que o teste *post hoc* de *Friedman* mostrou uma melhoria do algoritmo HSADE-v3 sobre EPSDE, DEahcSPX, SaDE, DEGL e jDE considerando-se um nível de significância  $\alpha = 0,05$ . Para os outros três



algoritmos, CoDE e JADE com e sem arquivo, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v1.

TABELA 30 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
CoDE	1,41E-05	1,13E-04
DEahcSPX	0,071	0,497
DEGL	0,294	1
jDE	0,435	1
EPSDE	0,826	1
JADE	1	1
JADE <i>withArch</i>	1	1
SaDE	1	1

FONTE: O autor (2014)

TABELA 31 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
EPSDE	1,78E-07	1,42E-06
SaDE	1,36E-06	9,52E-06
DEahcSPX	3,16E-06	1,90E-05
DEGL	1,76E-05	8,78E-05
jDE	0,001	0,005
CoDE	0,045	0,136
JADE <i>withArch</i>	0,130	0,261
JADE	0,205	0,261

FONTE: O autor (2014)

TABELA 32 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	1,96E-11	1,56E-10
DEGL	1,56E-07	1,09E-06
EPSDE	2,49E-06	1,50E-05
SaDE	1,28E-04	6,40E-04
jDE	0,012	0,048
CoDE	0,172	0,516
JADE <i>withArch</i>	0,272	0,545
JADE	0,272	0,545

FONTE: O autor (2014)

### 6.5.2.2 Análise do desempenho do algoritmo HSADE-v2

Nesta seção, uma análise dos resultados obtidos no teste de *Mann-Whitney* e no teste *post hoc* de *Friedman* considerando HSADE-v2 como algoritmo de controle é realizada.

#### 6.5.2.2.1 Teste de *Mann-Whitney*

Os resultados obtidos no teste de *Mann-Whitney* são apresentados nas TABELAS 39, 42 e 45. Uma análise dos resultados obtidos para cada grupo de função teste é detalhada na sequência.

- *Funções unimodais*

Assim como o observado na primeira versão, o algoritmo HSADE-v2 apresentou um desempenho muito bom na solução das funções unimodais.

Nota-se que nenhum dos seis algoritmos JADE *sem arquivo*, CoDE, SaDE, jDE, EPSDE, DEahcSPX apresentaram um resultado melhor do HSADE-v2 em nenhuma das cinco funções nas três dimensões (10, 30 e 50). JADE *com arquivo* superou HSADE-v2 em apenas uma função ( $F_4$ ) nas três dimensões (10, 30 e 50) e

DEGL superou HSADE-v2 em apenas uma função ( $F_4$ ) na dimensão 10. Em todos os outros casos HSADE-v1 foi superior ou apresentou desempenho semelhante.

- Funções multimodais básicas

No caso das funções multimodais básicas, observa-se também que algoritmo HSADE-v2 apresentou a mesma tendência observada na primeira versão.

O caso de dimensão 10 foi aquele no qual HSADE-v2 apresentou o seu pior desempenho. Entretanto, mesmo assim, HSADE-v2 teve um desempenho melhor, ou pelo menos semelhante em no mínimo onze funções quando comparado a cada uma das oito variantes DE.

Para o caso de dimensão 30 e 50, verifica-se que HSADE-v2 possui desempenho semelhante ou melhor do que todos os oito algoritmos de comparação em nove ( $F_7, F_{10}, F_{11}, F_{12}, F_{13}, F_{15}, F_{16}, F_{18}, F_{20}$ ) e onze ( $F_6, F_8, F_9, F_{10}, F_{12}, F_{13}, F_{15}, F_{16}, F_{18}, F_{19}, F_{20}$ ) funções para as dimensões 30 e 50, respectivamente. Além disto, DEahcSPX não superou HSADE-v2 em nenhuma função no caso de dimensão 30 e 50 enquanto DEGL não superou HSADE-v2 em nenhuma função no caso de dimensão 50.

- Funções multimodais compostas

Assim como foi observado nos outros conjuntos de funções, no caso das funções multimodais compostas, observa-se também que algoritmo HSADE-v2 apresentou a mesma tendência observada na primeira versão.

Em geral, o algoritmo HSADE-v2 apresentou um desempenho bom na solução das funções multimodais compostas. Entretanto, das três dimensões consideradas, o caso de dimensão 10 foi aquele no qual HSADE-v1 apresentou o seu pior desempenho. Para o caso de 10 dimensões, HSADE-v1 apresentou desempenho semelhante ou melhor do que JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX em sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), sete ( $F_{22}$  a  $F_{28}$ ), quatro ( $F_{23}, F_{26}$  a  $F_{28}$ ), sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), seis ( $F_{21}$  a  $F_{23}, F_{25}, F_{26}, F_{28}$ ), sete ( $F_{22}$  a  $F_{28}$ ) funções, respectivamente.

Já para o caso dimensão 30, HSADE-v2 apresentou um desempenho muito bom. HSADE-v2 foi superado na função  $F_{21}$  apenas por jDE, superado na função  $F_{22}$  por JADE com e sem arquivo, CoDE, jDE e EPSDE, superado na função  $F_{24}$  somente por JADE *sem arquivo* e superado na função  $F_{24}$  somente por jDE e DEahcSPX. Em todos os outros casos, HSADE-v2 foi superior ou, em alguns poucos casos, apresentou desempenho semelhante.

Para o caso de dimensão 50, HSADE-v1 também apresentou um bom desempenho. HSADE-v1 apresentou desempenho semelhante ou melhor do que JADE com e sem arquivo, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX em cinco ( $F_{21}, F_{23}, F_{25}, F_{26}, F_{27}$ ), cinco ( $F_{23}, F_{24}, F_{26}$  a  $F_{28}$ ), seis ( $F_{21}, F_{23}$  a  $F_{27}$ ), cinco ( $F_{23}$  a  $F_{27}$ ), sete ( $F_{21}, F_{23}$  a  $F_{28}$ ), sete ( $F_{21}$  a  $F_{24}, F_{26}$  a  $F_{28}$ ), sete ( $F_{22}$  a  $F_{28}$ ) funções, respectivamente. Além disso, na maioria destas funções o desempenho apresentado foi superior.

#### 6.5.2.2.2 Teste *post hoc* de *Friedman*

A TABELA 33 apresenta a classificação média dos algoritmos considerados e o valor-p obtidos através do teste de *Friedman* para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$ .

Os baixos valores-p obtidos demonstram a existência de diferenças significativas no desempenho entre os algoritmos comparados. Mais especificamente, pode-se dizer que a hipótese nula pode ser rejeitada para as três dimensões avaliadas com um nível de significância de  $\alpha = 1E-06$ .

Pode-se observar também através dos dados da TABELA 33, que segundo a classificação média obtida através do teste de *Friedman*, o algoritmo HSADE-v2 foi o que apresentou o melhor resultado nas dimensões 30 e 50. Na dimensão 10, apenas o algoritmo SaDE apresentou uma classificação melhor.

TABELA 33 – CLASSIFICAÇÃO MÉDIA E VALOR-P CORRESPONDENTE OBTIDOS ATRAVÉS DO TESTE DE *FRIEDAN* PARA OS PROBLEMAS DE OTIMIZAÇÃO PROPOSTOS NO CEC2013

	<i>D</i> = 10	<i>D</i> = 30	<i>D</i> = 50
JADE <i>com arquivo</i>	4,34	3,95	3,64
JADE <i>sem arquivo</i>	4,48	3,77	3,54
CoDE	7,64	4,20	3,84
SaDE	3,71	6,27	5,64
jDE	5,00	5,09	4,61
EPSDE	4,68	6,55	6,21
DEGL	5,39	5,98	6,75
DEahcSPX	5,91	6,21	7,75
HSADE-v2	3,84	2,98	3,02

<i>Valor-p</i>	1,86E-07	4,19E-09	9,70E-15
----------------	----------	----------	----------

FONTE: O autor (2014)

Uma vez calculada a classificação média de cada algoritmo, o procedimento *post hoc* descrito na seção 6.2.2 foi aplicado para a determinação dos valores-p. As TABELAS 34, 35 e 36 apresentam os valores-p e seus respectivo valores-p ajustados através do procedimento de Holm para as dimensões  $D = 10$ ,  $D = 30$  e  $D = 50$  respectivamente considerando o algoritmo HSADE-v2 como algoritmo de controle.

Como já havia sido verificado no teste de *Mann-Whitney* e também na primeira versão do algoritmo HSADE, o algoritmo HSADE-v2 não apresentou um bom resultado para a dimensão 10. Como se pode observar na TABELA 34, o teste *post hoc* de *Friedman* mostra uma melhoria significativa do algoritmo HSADE-v2 apenas sobre CoDE e DEahcSPX. Para todos os outros algoritmos, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v2.

Para as dimensões 30 e 50, verifica-se que o teste *post hoc* de *Friedman* mostrou uma melhoria do algoritmo HSADE-v2 sobre EPSDE, DEahcSPX, SaDE, DEGL e jDE considerando-se um nível de significância  $\alpha = 0,12$ . Para os outros três algoritmos, CoDE e JADE com e sem arquivo, o teste *post hoc* de *Friedman* não detectou nenhuma diferença significativa quando comparado com HSADE-v2 devido ao elevado valor-p ajustado obtido.

TABELA 34 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
CoDE	2,03E-07	1,62E-06
DEahcSPX	0,005	0,033
DEGL	0,034	0,203
jDE	0,113	0,564
EPSDE	0,252	1
JADE	0,380	1
JADE <i>withArch</i>	0,495	1
SaDE	1	1

FONTE: O autor (2014)

TABELA 35 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
EPSDE	1,06E-06	8,51E-06
SaDE	7,15E-06	5,01E-05
DEahcSPX	1,01E-05	6,03E-05
DEGL	4,15E-05	2,08E-04
jDE	0,004	0,016
CoDE	0,097	0,291
JADE <i>withArch</i>	0,188	0,375
JADE	0,283	0,375

FONTE: O autor (2014)

TABELA 36 – RESULTADO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

	<b>Valor-p não ajustado</b>	<b>Valor-p ajustado</b>
DEahcSPX	1,01E-10	8,09E-10
DEGL	3,41E-07	2,39E-06
EPSDE	1,26E-05	7,55E-05
SaDE	3,35E-04	0,0017
jDE	0,030	0,117
CoDE	0,262	0,785
JADE <i>withArch</i>	0,393	0,786
JADE	0,479	0,786

FONTE: O autor (2014)

TABELA 37 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,34E-29 ± 4,83E-29	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	1,37E-25 ± 6,64E-26	3,97E+00 ± 2,10E+01	4,22E+03 ± 3,48E+03	4,70E+03 ± 1,21E+04	2,13E+03 ± 1,96E+03	8,55E+01 ± 2,82E+02	1,94E-25 ± 6,56E-26	8,41E-07 ± 4,47E-06	5,70E-25 ± 1,06E-24	1,76E-25 ± 2,10E-25
$F_3$	3,09E+03 ± 2,15E+04	5,36E+05 ± 3,63E+06	7,94E+05 ± 2,83E+06	2,42E+01 ± 1,52E+02	3,44E+03 ± 2,41E+04	2,71E+03 ± 9,00E+03	3,74E-01 ± 1,50E+00	4,44E-01 ± 1,49E+00	1,20E+03 ± 8,08E+03	1,61E+00 ± 2,78E+00
$F_4$	3,32E-14 ± 2,33E-13	6,85E-03 ± 3,13E-02	1,38E+01 ± 4,35E+01	2,72E+01 ± 5,19E+01	2,26E+00 ± 5,82E+00	2,80E+01 ± 2,00E+02	5,56E-27 ± 2,01E-27	4,12E-04 ± 1,98E-03	1,75E-12 ± 1,24E-11	2,91E-07 ± 2,07E-06
$F_5$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,74E-16 ± 1,02E-15	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_6$	8,08E+00 ± 3,78E+00	8,08E+00 ± 3,78E+00	6,83E+00 ± 4,50E+00	6,22E+00 ± 4,71E+00	8,36E+00 ± 3,45E+00	3,85E+00 ± 4,84E+00	5,75E+00 ± 4,78E+00	3,85E-01 ± 1,92E+00	7,40E+00 ± 4,21E+00	7,12E+00 ± 4,42E+00
$F_7$	2,48E+00 ± 5,08E+00	1,77E+00 ± 2,73E+00	1,61E+01 ± 1,60E+01	1,24E-01 ± 5,63E-01	6,05E-01 ± 1,25E+00	9,96E-01 ± 1,72E+00	1,02E-01 ± 6,12E-01	5,23E-03 ± 1,31E-02	2,91E-01 ± 6,67E-01	9,45E-01 ± 5,05E+00
$F_8$	2,04E+01 ± 8,60E-02	2,03E+01 ± 9,43E-02	2,04E+01 ± 1,49E-01	2,04E+01 ± 6,59E-02	2,03E+01 ± 8,15E-02	2,04E+01 ± 6,64E-02	2,03E+01 ± 8,38E-02	2,04E+01 ± 6,77E-02	2,03E+01 ± 1,02E-01	2,03E+01 ± 1,66E-01
$F_9$	4,14E+00 ± 8,56E-01	3,72E+00 ± 1,06E+00	4,29E+00 ± 1,69E+00	1,36E+00 ± 8,09E-01	2,08E+00 ± 1,14E+00	4,88E+00 ± 7,05E-01	9,77E-01 ± 1,03E+00	7,07E-01 ± 8,24E-01	1,99E+00 ± 9,70E-01	1,50E+00 ± 9,37E-01
$F_{10}$	2,16E-02 ± 1,74E-02	3,66E-02 ± 3,60E-02	3,02E-01 ± 5,28E-01	3,60E-02 ± 2,20E-02	8,78E-02 ± 4,28E-02	3,69E-02 ± 1,87E-02	5,14E-02 ± 5,06E-02	9,20E-02 ± 9,51E-02	6,72E-02 ± 5,49E-02	3,10E-02 ± 2,38E-02
$F_{11}$	0,00E+00 ± 0,00E+00	1,95E-02 ± 1,39E-01	1,15E+00 ± 1,05E+00	0,00E+00 ± 0,00E+00	1,95E-02 ± 1,39E-01	0,00E+00 ± 0,00E+00	2,99E+00 ± 3,16E+00	1,35E+00 ± 1,25E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_{12}$	5,66E+00 ± 2,54E+00	5,07E+00 ± 1,87E+00	1,66E+01 ± 6,54E+00	5,56E+00 ± 1,93E+00	7,27E+00 ± 2,49E+00	6,55E+00 ± 2,21E+00	5,54E+00 ± 4,95E+00	7,30E+00 ± 3,77E+00	5,85E+00 ± 2,39E+00	5,72E+00 ± 2,52E+00
$F_{13}$	8,95E+00 ± 5,63E+00	9,97E+00 ± 5,54E+00	3,06E+01 ± 1,24E+01	7,43E+00 ± 4,64E+00	1,28E+01 ± 6,27E+00	7,62E+00 ± 3,50E+00	7,47E+00 ± 6,31E+00	1,29E+01 ± 8,52E+00	7,65E+00 ± 5,04E+00	9,30E+00 ± 5,56E+00



TABELA 37 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013

(continuação)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{14}$	1,25E-01 ± 6,49E-02	4,25E-01 ± 1,71E+00	1,83E+01 ± 3,16E+01	3,06E-02 ± 3,61E-02	2,09E-01 ± 6,77E-01	9,39E-02 ± 4,76E-01	9,58E+02 ± 2,19E+02	6,92E+01 ± 1,09E+02	4,91E-01 ± 1,70E+00	2,57E+00 ± 1,66E+01
$F_{15}$	4,72E+02 ± 1,63E+02	4,43E+02 ± 1,45E+02	8,23E+02 ± 2,89E+02	7,14E+02 ± 2,22E+02	7,90E+02 ± 1,47E+02	7,98E+02 ± 2,20E+02	1,23E+03 ± 2,08E+02	1,22E+03 ± 3,14E+02	4,77E+02 ± 1,86E+02	4,70E+02 ± 1,98E+02
$F_{16}$	7,87E-01 ± 3,38E-01	7,53E-01 ± 3,74E-01	9,98E-01 ± 6,44E-01	1,07E+00 ± 1,77E-01	1,06E+00 ± 1,87E-01	1,11E+00 ± 1,95E-01	1,12E+00 ± 1,63E-01	1,14E+00 ± 2,13E-01	1,12E-01 ± 6,99E-02	6,26E-02 ± 4,16E-02
$F_{17}$	9,93E+00 ± 1,42E+00	1,01E+01 ± 4,67E-02	1,05E+01 ± 2,12E+00	1,01E+01 ± 4,85E-03	1,01E+01 ± 2,34E-02	1,01E+01 ± 2,74E-03	2,81E+01 ± 3,43E+00	1,44E+01 ± 2,56E+00	1,01E+01 ± 2,20E-02	1,01E+01 ± 2,37E-02
$F_{18}$	1,55E+01 ± 1,92E+00	1,53E+01 ± 1,77E+00	2,65E+01 ± 1,04E+01	2,70E+01 ± 3,80E+00	2,61E+01 ± 4,44E+00	2,44E+01 ± 3,56E+00	3,20E+01 ± 3,38E+00	3,30E+01 ± 1,21E+01	1,67E+01 ± 2,84E+00	1,66E+01 ± 2,99E+00
$F_{19}$	2,62E-01 ± 5,06E-02	2,62E-01 ± 5,35E-02	6,55E-01 ± 2,56E-01	5,77E-01 ± 1,07E-01	2,65E-01 ± 7,07E-02	3,03E-01 ± 5,65E-02	1,46E+00 ± 3,49E-01	8,93E-01 ± 4,79E-01	3,67E-01 ± 1,26E-01	4,24E-01 ± 1,17E-01
$F_{20}$	2,63E+00 ± 5,95E-01	2,52E+00 ± 5,64E-01	3,28E+00 ± 5,00E-01	2,38E+00 ± 4,25E-01	2,33E+00 ± 5,11E-01	2,50E+00 ± 4,84E-01	2,41E+00 ± 5,27E-01	2,76E+00 ± 4,81E-01	2,72E+00 ± 5,92E-01	2,62E+00 ± 5,71E-01
$F_{21}$	3,96E+02 ± 2,80E+01	4,00E+02 ± 0,00E+00	3,63E+02 ± 7,74E+01	3,96E+02 ± 2,80E+01	3,88E+02 ± 4,76E+01	4,00E+02 ± 0,00E+00	4,00E+02 ± 1,39E-14	3,65E+02 ± 7,71E+01	4,00E+02 ± 8,04E-15	3,96E+02 ± 2,80E+01
$F_{22}$	2,95E+01 ± 3,46E+01	2,29E+01 ± 3,45E+01	1,33E+02 ± 1,47E+02	2,01E+01 ± 2,58E+01	1,37E+01 ± 1,09E+01	6,46E+00 ± 5,10E+00	6,78E+02 ± 3,25E+02	1,91E+02 ± 2,39E+02	4,01E+01 ± 4,48E+01	4,02E+01 ± 4,00E+01
$F_{23}$	5,54E+02 ± 2,23E+02	5,70E+02 ± 2,11E+02	1,13E+03 ± 3,42E+02	6,50E+02 ± 2,58E+02	8,40E+02 ± 2,75E+02	8,52E+02 ± 1,76E+02	1,05E+03 ± 3,99E+02	1,15E+03 ± 3,65E+02	5,42E+02 ± 2,15E+02	5,25E+02 ± 2,27E+02
$F_{24}$	2,04E+02 ± 1,46E+01	2,01E+02 ± 1,59E+01	2,08E+02 ± 1,32E+01	1,95E+02 ± 2,17E+01	2,01E+02 ± 1,86E+01	2,02E+02 ± 1,62E+01	2,01E+02 ± 1,47E+01	2,03E+02 ± 1,41E+01	2,05E+02 ± 4,21E+00	2,03E+02 ± 1,24E+01
$F_{25}$	2,04E+02 ± 7,59E+00	2,02E+02 ± 7,98E+00	2,09E+02 ± 5,38E+00	2,00E+02 ± 4,64E-01	2,04E+02 ± 3,63E+00	2,05E+02 ± 6,42E+00	2,01E+02 ± 1,42E+01	2,02E+02 ± 2,40E+00	2,03E+02 ± 3,81E+00	1,99E+02 ± 1,53E+01
$F_{26}$	1,51E+02 ± 7,01E+01	1,51E+02 ± 6,69E+01	1,57E+02 ± 5,30E+01	1,10E+02 ± 1,88E+01	1,51E+02 ± 5,37E+01	1,21E+02 ± 2,18E+01	1,64E+02 ± 6,24E+01	1,49E+02 ± 4,55E+01	1,49E+02 ± 5,18E+01	1,39E+02 ± 4,55E+01

TABELA 37 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{27}$	3,28E+02 ± 6,76E+01	3,15E+02 ± 5,09E+01	4,01E+02 ± 8,98E+01	3,02E+02 ± 1,40E+01	3,48E+02 ± 8,31E+01	3,02E+02 ± 1,40E+01	3,38E+02 ± 7,61E+01	3,24E+02 ± 6,13E+01	3,22E+02 ± 4,15E+01	3,23E+02 ± 5,56E+01
$F_{28}$	2,96E+02 ± 2,80E+01	3,05E+02 ± 3,67E+01	3,30E+02 ± 1,43E+02	2,96E+02 ± 2,80E+01	2,92E+02 ± 3,92E+01	2,92E+02 ± 3,92E+01	3,01E+02 ± 4,60E+01	3,00E+02 ± 0,00E+00	3,06E+02 ± 5,74E+01	3,00E+02 ± 0,00E+00

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 100.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324.

TABELA 38 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	+	≈	≈	≈	≈	≈
	$F_2$	-	+	+	+	+	+	-	+
	$F_3$	≈	+	+	+	+	+	+	+
	$F_4$	-	+	+	+	+	+	-	+
	$F_5$	≈	≈	+	≈	≈	≈	≈	≈
<i>Funções Multimodais Básicas</i>	$F_6$	+	+	+	+	+	≈	≈	-
	$F_7$	+	+	+	-	+	+	-	-
	$F_8$	+	≈	+	≈	≈	+	≈	+
	$F_9$	+	+	+	-	≈	+	-	-
	$F_{10}$	-	-	+	-	+	-	≈	≈
	$F_{11}$	≈	≈	+	≈	≈	≈	+	+
	$F_{12}$	≈	≈	+	≈	+	≈	-	+
	$F_{13}$	≈	+	+	≈	+	≈	≈	+
	$F_{14}$	-	-	+	-	-	-	+	+
	$F_{15}$	≈	≈	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	-	≈	+	+	+	+	+	+
	$F_{18}$	-	-	+	+	+	+	+	+
	$F_{19}$	-	-	+	+	-	-	+	+
$F_{20}$	≈	≈	+	-	-	-	-	≈	
<i>Funções Multimodais Compostas</i>	$F_{21}$	≈	≈	-	-	≈	≈	≈	-
	$F_{22}$	-	-	+	-	-	-	+	+
	$F_{23}$	≈	≈	+	+	+	+	+	+
	$F_{24}$	≈	≈	+	-	≈	≈	-	≈
	$F_{25}$	≈	≈	+	-	≈	+	-	-
	$F_{26}$	≈	≈	+	-	+	≈	≈	≈
	$F_{27}$	≈	≈	+	≈	≈	-	-	-
	$F_{28}$	≈	≈	≈	≈	≈	≈	≈	≈
<b>Total</b>	≈	15	15	1	8	10	10	9	7
	-	8	5	1	10	4	6	9	6
	+	5	8	26	10	14	12	10	15

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 39 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	+	≈	≈	≈	≈	≈
	$F_2$	≈	+	+	+	+	+	+	+
	$F_3$	≈	+	+	+	+	+	≈	+
	$F_4$	-	+	+	+	+	+	-	+
	$F_5$	≈	≈	+	≈	≈	≈	≈	≈
<i>Funções Multimodais Básicas</i>	$F_6$	≈	+	+	+	≈	-	≈	-
	$F_7$	+	+	+	≈	+	+	-	-
	$F_8$	+	≈	+	+	≈	+	+	+
	$F_9$	+	+	+	≈	+	+	-	-
	$F_{10}$	-	≈	+	≈	+	+	+	+
	$F_{11}$	≈	≈	+	≈	≈	≈	+	+
	$F_{12}$	≈	≈	+	≈	+	+	≈	+
	$F_{13}$	≈	≈	+	≈	+	≈	-	+
	$F_{14}$	-	-	+	-	-	-	+	+
	$F_{15}$	≈	≈	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	-	-	+	+	≈	≈	+	+
	$F_{18}$	≈	-	+	+	+	+	+	+
	$F_{19}$	-	-	+	+	-	-	+	+
$F_{20}$	≈	≈	+	-	-	≈	-	≈	
<i>Funções Multimodais Compostas</i>	$F_{21}$	≈	≈	-	-	≈	≈	≈	-
	$F_{22}$	-	-	+	-	-	-	+	+
	$F_{23}$	≈	≈	+	+	+	+	+	+
	$F_{24}$	≈	≈	+	-	≈	≈	-	≈
	$F_{25}$	+	≈	+	+	+	+	≈	≈
	$F_{26}$	≈	≈	+	≈	+	≈	+	≈
	$F_{27}$	+	+	+	≈	+	≈	-	≈
	$F_{28}$	≈	≈	≈	≈	≈	≈	≈	≈
<b>Total</b>	≈	16	15	1	11	9	11	8	8
	-	6	5	1	6	4	4	7	4
	+	6	8	26	11	15	13	13	16

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 40 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADe-v1	HSADe-v2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	2,47E-31 ± 1,77E-30	0,00E+00 ± 0,00E+00	1,87E-28 ± 3,44E-28	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	7,47E+03 ± 5,89E+03	1,34E+04 ± 7,93E+03	7,62E+04 ± 4,54E+04	4,23E+05 ± 2,29E+05	1,33E+05 ± 7,74E+04	2,62E+06 ± 5,72E+06	4,19E+04 ± 2,96E+04	2,59E+05 ± 1,23E+05	3,80E+03 ± 3,02E+03	5,82E+03 ± 5,45E+03
$F_3$	8,85E+05 ± 3,67E+06	1,51E+05 ± 7,42E+05	8,14E+05 ± 2,09E+06	1,10E+07 ± 1,50E+07	1,45E+06 ± 2,27E+06	4,19E+07 ± 1,28E+08	5,57E+03 ± 1,55E+04	7,48E+05 ± 2,34E+06	1,11E+05 ± 5,61E+05	6,49E+04 ± 4,62E+05
$F_4$	5,06E+03 ± 1,21E+04	5,12E+03 ± 1,35E+04	5,23E-02 ± 6,18E-02	3,08E+03 ± 1,55E+03	5,53E+00 ± 7,53E+00	2,03E+04 ± 1,67E+04	3,50E-01 ± 2,16E-01	9,66E+02 ± 4,66E+02	7,49E+02 ± 2,83E+03	3,13E-04 ± 4,86E-04
$F_5$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	9,75E-16 ± 3,18E-15	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_6$	1,55E+00 ± 6,28E+00	1,19E+00 ± 5,20E+00	3,04E+00 ± 7,79E+00	2,89E+01 ± 2,65E+01	1,23E+01 ± 3,14E+00	4,42E+00 ± 4,36E+00	3,29E+00 ± 7,15E+00	1,00E+01 ± 1,39E+01	1,55E+00 ± 6,28E+00	3,46E-05 ± 2,26E-04
$F_7$	4,24E+00 ± 4,56E+00	3,17E+00 ± 3,47E+00	8,28E+00 ± 5,50E+00	3,05E+01 ± 1,34E+01	3,30E+00 ± 2,94E+00	2,15E+01 ± 1,55E+01	3,92E+00 ± 3,96E+00	2,92E+00 ± 2,68E+00	2,85E+00 ± 2,48E+00	2,62E+00 ± 2,11E+00
$F_8$	2,09E+01 ± 5,15E-02	2,09E+01 ± 7,40E-02	2,08E+01 ± 1,07E-01	2,10E+01 ± 4,02E-02	2,10E+01 ± 5,18E-02	2,10E+01 ± 5,44E-02	2,09E+01 ± 5,24E-02	2,09E+01 ± 5,79E-02	2,09E+01 ± 5,30E-02	2,09E+01 ± 4,81E-02
$F_9$	2,70E+01 ± 1,47E+00	2,69E+01 ± 1,60E+00	1,41E+01 ± 3,09E+00	1,75E+01 ± 2,89E+00	2,78E+01 ± 2,02E+00	2,92E+01 ± 1,82E+00	9,22E+00 ± 3,36E+00	3,63E+01 ± 5,23E+00	1,25E+01 ± 2,13E+00	1,09E+01 ± 2,44E+00
$F_{10}$	3,84E-02 ± 2,47E-02	6,74E-02 ± 4,00E-02	3,28E-02 ± 1,87E-02	2,90E-01 ± 1,25E-01	3,72E-02 ± 2,10E-02	1,07E-01 ± 5,29E-02	2,71E-02 ± 1,92E-02	2,87E-02 ± 2,35E-02	3,90E-02 ± 1,93E-02	3,86E-02 ± 3,50E-02
$F_{11}$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	3,90E-02 ± 1,95E-01	2,15E-01 ± 6,69E-01	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	6,47E+01 ± 5,50E+01	2,05E+01 ± 6,28E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_{12}$	2,40E+01 ± 3,92E+00	2,53E+01 ± 4,98E+00	3,63E+01 ± 1,22E+01	4,77E+01 ± 1,43E+01	5,98E+01 ± 1,11E+01	4,74E+01 ± 7,30E+00	1,59E+02 ± 2,22E+01	4,15E+01 ± 4,69E+01	1,97E+01 ± 5,57E+00	1,70E+01 ± 3,96E+00
$F_{13}$	4,15E+01 ± 1,08E+01	4,78E+01 ± 1,20E+01	7,87E+01 ± 2,43E+01	1,01E+02 ± 2,34E+01	9,27E+01 ± 1,48E+01	7,59E+01 ± 1,72E+01	1,63E+02 ± 1,14E+01	1,14E+02 ± 5,75E+01	3,82E+01 ± 1,68E+01	3,33E+01 ± 1,11E+01

TABELA 40 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013

(continuação)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADe-v1	HSADe-v2
$F_{14}$	2,78E-02 ± 2,45E-02	2,29E-02 ± 2,05E-02	2,67E+00 ± 2,71E+00	7,78E-01 ± 1,28E+00	1,63E-03 ± 5,65E-03	1,53E-01 ± 2,58E-01	6,66E+03 ± 3,38E+02	9,34E+02 ± 3,42E+02	1,77E-01 ± 1,77E-01	1,75E+01 ± 6,32E+00
$F_{15}$	3,31E+03 ± 2,76E+02	3,31E+03 ± 3,33E+02	3,43E+03 ± 6,47E+02	4,84E+03 ± 9,70E+02	5,08E+03 ± 3,84E+02	5,80E+03 ± 1,03E+03	7,15E+03 ± 3,27E+02	7,02E+03 ± 3,39E+02	2,35E+03 ± 4,00E+02	2,36E+03 ± 4,13E+02
$F_{16}$	1,84E+00 ± 6,60E-01	1,76E+00 ± 5,92E-01	3,74E-01 ± 1,93E-01	2,22E+00 ± 3,23E-01	2,37E+00 ± 3,02E-01	2,44E+00 ± 2,97E-01	2,44E+00 ± 3,14E-01	2,38E+00 ± 2,85E-01	7,38E-02 ± 3,43E-02	6,56E-02 ± 2,51E-02
$F_{17}$	3,04E+01 ± 3,07E-14	3,04E+01 ± 2,11E-14	3,04E+01 ± 2,74E-02	3,04E+01 ± 2,07E-02	3,04E+01 ± 8,46E-04	3,04E+01 ± 1,97E-02	1,83E+02 ± 1,14E+01	6,29E+01 ± 1,07E+01	3,04E+01 ± 9,14E-05	3,04E+01 ± 7,99E-03
$F_{18}$	7,67E+01 ± 6,29E+00	7,62E+01 ± 7,09E+00	6,45E+01 ± 9,46E+00	1,36E+02 ± 3,75E+01	1,59E+02 ± 1,53E+01	1,35E+02 ± 1,45E+01	1,92E+02 ± 9,51E+00	1,88E+02 ± 4,15E+01	4,93E+01 ± 6,77E+00	4,75E+01 ± 4,29E+00
$F_{19}$	1,44E+00 ± 1,24E-01	1,42E+00 ± 1,31E-01	1,56E+00 ± 3,01E-01	4,18E+00 ± 7,79E-01	1,65E+00 ± 1,55E-01	1,83E+00 ± 1,61E-01	1,23E+01 ± 1,52E+00	2,71E+00 ± 7,41E-01	1,19E+00 ± 2,20E-01	1,92E+00 ± 3,19E-01
$F_{20}$	1,04E+01 ± 4,78E-01	1,05E+01 ± 5,29E-01	1,07E+01 ± 5,92E-01	1,09E+01 ± 6,60E-01	1,17E+01 ± 3,63E-01	1,18E+01 ± 7,76E-01	1,16E+01 ± 4,29E-01	1,23E+01 ± 4,12E-01	1,05E+01 ± 5,51E-01	1,03E+01 ± 7,33E-01
$F_{21}$	3,09E+02 ± 6,80E+01	3,10E+02 ± 7,62E+01	3,10E+02 ± 8,51E+01	3,43E+02 ± 8,83E+01	2,74E+02 ± 5,79E+01	3,21E+02 ± 7,12E+01	3,09E+02 ± 7,78E+01	3,04E+02 ± 8,90E+01	3,37E+02 ± 6,74E+01	3,09E+02 ± 8,15E+01
$F_{22}$	8,90E+01 ± 3,39E+01	9,53E+01 ± 2,91E+01	1,18E+02 ± 2,31E+01	1,18E+02 ± 2,43E+01	1,40E+02 ± 2,27E+01	1,21E+02 ± 5,60E+01	6,60E+03 ± 3,69E+02	9,29E+02 ± 2,56E+02	1,11E+02 ± 3,75E+00	1,24E+02 ± 2,53E+01
$F_{23}$	3,49E+03 ± 3,79E+02	3,44E+03 ± 4,01E+02	3,46E+03 ± 7,66E+02	4,82E+03 ± 1,34E+03	5,65E+03 ± 4,94E+02	6,05E+03 ± 7,98E+02	7,05E+03 ± 3,30E+02	7,11E+03 ± 3,66E+02	2,42E+03 ± 4,76E+02	2,39E+03 ± 4,97E+02
$F_{24}$	2,14E+02 ± 1,17E+01	2,07E+02 ± 9,00E+00	2,20E+02 ± 9,42E+00	2,27E+02 ± 6,33E+00	2,15E+02 ± 1,14E+01	2,26E+02 ± 1,90E+01	2,16E+02 ± 1,17E+01	2,13E+02 ± 9,14E+00	2,09E+02 ± 4,98E+00	2,11E+02 ± 4,48E+00
$F_{25}$	2,75E+02 ± 1,26E+01	2,74E+02 ± 1,15E+01	2,55E+02 ± 7,67E+00	2,65E+02 ± 1,38E+01	2,49E+02 ± 7,57E+00	2,92E+02 ± 4,77E+00	2,51E+02 ± 1,00E+01	2,46E+02 ± 5,19E+00	2,47E+02 ± 6,63E+00	2,52E+02 ± 5,21E+00
$F_{26}$	2,18E+02 ± 4,63E+01	2,23E+02 ± 5,14E+01	2,14E+02 ± 3,94E+01	2,00E+02 ± 1,14E-02	2,02E+02 ± 1,43E+01	2,00E+02 ± 4,86E-01	2,27E+02 ± 4,88E+01	2,02E+02 ± 1,72E+01	2,02E+02 ± 1,65E+01	2,02E+02 ± 1,55E+01

TABELA 40 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{27}$	6,87E+02 ± 1,81E+02	6,44E+02 ± 2,16E+02	5,70E+02 ± 1,12E+02	6,08E+02 ± 6,73E+01	6,96E+02 ± 1,96E+02	9,54E+02 ± 1,18E+02	4,79E+02 ± 1,09E+02	5,49E+02 ± 2,44E+02	4,95E+02 ± 1,08E+02	3,96E+02 ± 4,22E+01
$F_{28}$	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00	3,16E+02 ± 1,45E+02	3,00E+02 ± 2,50E-13	3,00E+02 ± 0,00E+00	3,00E+02 ± 0,00E+00

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 300.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324.

TABELA 41 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	≈	≈	+
	$F_2$	+	+	+	+	+	+	+	+
	$F_3$	≈	≈	+	+	+	+	≈	+
	$F_4$	≈	+	+	+	+	+	+	+
	$F_5$	≈	≈	≈	≈	≈	≈	≈	+
<i>Funções Multimodais Básicas</i>	$F_6$	-	+	+	+	+	+	+	+
	$F_7$	≈	≈	+	+	≈	+	≈	≈
	$F_8$	≈	≈	-	≈	≈	≈	≈	≈
	$F_9$	+	+	+	+	+	+	-	+
	$F_{10}$	≈	+	≈	+	≈	+	-	-
	$F_{11}$	≈	≈	≈	+	≈	≈	+	+
	$F_{12}$	+	+	+	+	+	+	+	+
	$F_{13}$	≈	+	+	+	+	+	+	+
	$F_{14}$	-	-	+	≈	-	-	+	+
	$F_{15}$	+	+	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	+	+	+	+	+	-	+	+
	$F_{18}$	+	+	+	+	+	+	+	+
	$F_{19}$	+	+	+	+	+	+	+	+
$F_{20}$	≈	≈	+	+	+	+	+	+	
<i>Funções Compostas</i>	$F_{21}$	-	-	-	≈	-	≈	≈	≈
	$F_{22}$	-	-	+	+	+	≈	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	+	-	+	+	+	+	+	+
	$F_{25}$	+	+	+	+	≈	+	+	≈
	$F_{26}$	+	+	+	+	+	+	+	+
	$F_{27}$	+	+	+	+	+	+	≈	≈
	$F_{28}$	≈	≈	≈	≈	≈	≈	≈	+
<b>Total</b>	≈	11	8	5	6	8	7	8	5
	-	4	4	2	0	2	2	2	1
	+	13	16	21	22	18	19	18	22

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.



TABELA 42 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	≈	≈	+	
	$F_2$	≈	+	+	+	+	+	+	+	
	$F_3$	+	≈	+	+	+	+	≈	+	
	$F_4$	-	+	+	+	+	+	+	+	
	$F_5$	≈	≈	≈	≈	≈	≈	≈	+	
<i>Funções Multimodais Básicas</i>	$F_6$	-	≈	+	+	+	+	+	+	
	$F_7$	≈	≈	+	+	≈	+	≈	≈	
	$F_8$	≈	≈	-	≈	≈	≈	≈	≈	
	$F_9$	+	+	+	+	+	+	-	+	
	$F_{10}$	≈	+	≈	+	≈	+	≈	≈	
	$F_{11}$	≈	≈	≈	+	≈	≈	+	+	
	$F_{12}$	+	+	+	+	+	+	+	+	
	$F_{13}$	+	+	+	+	+	+	+	+	
	$F_{14}$	-	-	-	-	-	-	-	+	+
	$F_{15}$	+	+	+	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+	+
	$F_{17}$	-	-	-	-	-	-	-	+	+
	$F_{18}$	+	+	+	+	+	+	+	+	+
	$F_{19}$	-	-	-	+	-	-	-	+	+
$F_{20}$	≈	+	+	+	+	+	+	+	+	
<i>Funções Multimodais Compostas</i>	$F_{21}$	≈	≈	≈	+	-	≈	≈	+	
	$F_{22}$	-	-	-	-	+	-	+	+	
	$F_{23}$	+	+	+	+	+	+	+	+	
	$F_{24}$	≈	-	+	+	≈	+	≈	≈	
	$F_{25}$	+	+	≈	+	-	+	≈	-	
	$F_{26}$	+	+	+	+	+	+	+	+	
	$F_{27}$	+	+	+	+	+	+	+	+	
	$F_{28}$	≈	≈	≈	≈	≈	≈	≈	+	
<b>Total</b>	≈	11	9	7	4	8	6	10	4	
	-	6	5	5	3	5	4	1	1	
	+	11	14	16	21	15	18	17	23	

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 43 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013

(continua)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_1$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	1,96E-29 ± 5,80E-29	0,00E+00 ± 0,00E+00	3,57E-10 ± 5,34E-10	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_2$	4,43E+04 ± 2,59E+04	6,95E+04 ± 3,62E+04	6,18E+05 ± 2,15E+05	8,95E+05 ± 3,71E+05	6,84E+05 ± 2,58E+05	9,15E+06 ± 1,77E+07	5,97E+05 ± 2,15E+05	4,82E+06 ± 1,79E+06	3,29E+04 ± 1,77E+04	4,54E+04 ± 2,12E+04
$F_3$	2,11E+05 ± 5,64E+05	1,62E+05 ± 5,96E+05	1,55E+06 ± 2,18E+06	6,74E+06 ± 7,90E+06	3,25E+06 ± 3,10E+06	4,69E+08 ± 1,22E+09	8,44E+05 ± 1,48E+06	6,65E+07 ± 6,61E+07	4,16E+05 ± 1,12E+06	6,75E+04 ± 1,29E+05
$F_4$	8,73E+03 ± 2,23E+04	8,20E+03 ± 2,29E+04	4,71E-01 ± 4,49E-01	1,86E+03 ± 6,55E+02	1,81E+02 ± 1,04E+03	1,92E+04 ± 2,22E+04	2,03E+03 ± 4,45E+02	1,62E+04 ± 3,57E+03	2,99E+03 ± 9,17E+03	3,25E-03 ± 4,52E-03
$F_5$	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00	2,15E-30 ± 4,72E-30	1,16E-20 ± 2,73E-20	1,00E-27 ± 1,44E-27	2,40E-23 ± 1,17E-22	9,67E-17 ± 6,91E-16	6,92E-06 ± 7,68E-06	0,00E+00 ± 0,00E+00	0,00E+00 ± 0,00E+00
$F_6$	4,34E+01 ± 3,67E-14	4,36E+01 ± 7,95E-01	4,34E+01 ± 5,47E-13	4,90E+01 ± 1,23E+01	4,52E+01 ± 8,16E-01	4,36E+01 ± 7,98E+00	4,47E+01 ± 5,69E+00	4,36E+01 ± 3,83E-01	4,36E+01 ± 7,95E-01	4,34E+01 ± 2,90E-14
$F_7$	5,08E+00 ± 3,18E+00	6,52E+00 ± 3,08E+00	1,42E+01 ± 7,26E+00	1,84E+01 ± 9,00E+00	9,45E+00 ± 5,05E+00	3,55E+01 ± 1,57E+01	1,43E+01 ± 1,06E+01	4,37E+01 ± 1,41E+01	5,54E+00 ± 3,57E+00	7,36E+00 ± 3,50E+00
$F_8$	2,11E+01 ± 4,34E-02	2,11E+01 ± 3,30E-02	2,11E+01 ± 3,88E-02	2,11E+01 ± 3,24E-02	2,11E+01 ± 4,37E-02	2,11E+01 ± 3,95E-02	2,11E+01 ± 3,80E-02	2,11E+01 ± 4,16E-02	2,11E+01 ± 3,17E-02	2,11E+01 ± 3,50E-02
$F_9$	5,38E+01 ± 2,46E+00	5,36E+01 ± 2,05E+00	2,38E+01 ± 4,72E+00	4,77E+01 ± 1,25E+01	5,74E+01 ± 2,29E+00	5,84E+01 ± 4,46E+00	3,32E+01 ± 1,84E+01	6,98E+01 ± 4,71E+00	2,55E+01 ± 3,46E+00	2,49E+01 ± 5,91E+00
$F_{10}$	3,11E-02 ± 1,88E-02	9,93E-02 ± 6,64E-02	2,73E-02 ± 1,78E-02	1,88E-01 ± 1,22E-01	3,82E-02 ± 2,66E-02	1,26E-01 ± 6,57E-02	6,54E-02 ± 4,22E-02	2,04E-01 ± 2,12E-01	3,15E-02 ± 1,96E-02	2,87E-02 ± 2,27E-02
$F_{11}$	2,36E-11 ± 1,20E-11	1,50E-11 ± 9,40E-12	1,47E+01 ± 5,38E+00	3,45E-14 ± 9,84E-14	0,00E+00 ± 0,00E+00	7,80E-02 ± 2,70E-01	2,17E+02 ± 1,17E+02	4,05E+01 ± 9,82E+00	2,93E-15 ± 1,05E-14	2,73E+01 ± 2,82E+00
$F_{12}$	7,77E+01 ± 8,55E+00	8,20E+01 ± 8,04E+00	6,72E+01 ± 1,59E+01	1,19E+02 ± 6,51E+01	1,62E+02 ± 1,84E+01	1,61E+02 ± 3,14E+01	3,47E+02 ± 1,58E+01	3,18E+02 ± 3,16E+01	3,70E+01 ± 8,40E+00	3,41E+01 ± 5,50E+00
$F_{13}$	1,27E+02 ± 2,07E+01	1,36E+02 ± 1,95E+01	1,56E+02 ± 3,91E+01	2,80E+02 ± 3,48E+01	2,29E+02 ± 2,61E+01	2,08E+02 ± 2,90E+01	3,41E+02 ± 1,90E+01	3,36E+02 ± 2,93E+01	8,29E+01 ± 2,19E+01	8,84E+01 ± 2,77E+01

TABELA 43 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013

(continuação)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSADe-v1	HSADe-v2
$F_{14}$	2,81E+01 ± 3,57E+00	3,11E+01 ± 3,43E+00	1,29E+03 ± 2,30E+02	9,97E+02 ± 1,09E+02	8,02E+01 ± 1,67E+01	1,24E+02 ± 1,31E+02	1,30E+04 ± 4,24E+02	6,18E+03 ± 2,55E+03	4,05E+01 ± 7,98E+00	2,06E+03 ± 2,25E+02
$F_{15}$	8,86E+03 ± 3,98E+02	8,84E+03 ± 5,03E+02	6,44E+03 ± 8,92E+02	1,25E+04 ± 3,86E+02	1,08E+04 ± 4,69E+02	1,30E+04 ± 1,09E+03	1,39E+04 ± 3,99E+02	1,39E+04 ± 3,63E+02	4,93E+03 ± 5,13E+02	4,95E+03 ± 6,58E+02
$F_{16}$	2,79E+00 ± 7,30E-01	2,37E+00 ± 8,86E-01	3,03E+00 ± 5,32E-01	3,27E+00 ± 2,84E-01	3,20E+00 ± 2,41E-01	3,36E+00 ± 2,68E-01	3,35E+00 ± 3,52E-01	3,29E+00 ± 2,80E-01	1,71E-01 ± 1,86E-01	9,29E-01 ± 6,17E-01
$F_{17}$	5,09E+01 ± 2,12E-02	5,09E+01 ± 2,14E-02	8,92E+01 ± 3,46E+00	8,17E+01 ± 1,95E+00	5,55E+01 ± 8,07E-01	5,08E+01 ± 2,94E-01	3,73E+02 ± 1,31E+01	1,59E+02 ± 2,05E+01	5,10E+01 ± 5,33E-02	8,52E+01 ± 3,42E+00
$F_{18}$	2,03E+02 ± 1,11E+01	2,03E+02 ± 1,26E+01	1,24E+02 ± 5,58E+01	3,66E+02 ± 1,41E+01	3,39E+02 ± 1,92E+01	3,40E+02 ± 1,99E+01	3,88E+02 ± 1,66E+01	4,05E+02 ± 1,98E+01	7,91E+01 ± 7,39E+00	7,48E+01 ± 6,30E+00
$F_{19}$	4,05E+00 ± 1,84E-01	3,97E+00 ± 2,69E-01	1,34E+01 ± 2,24E+00	1,01E+01 ± 1,10E+00	4,16E+00 ± 2,96E-01	5,83E+00 ± 8,05E-01	2,58E+01 ± 2,47E+00	1,45E+01 ± 6,30E+00	2,24E+00 ± 3,38E-01	4,00E+00 ± 4,87E-01
$F_{20}$	1,99E+01 ± 3,78E-01	1,98E+01 ± 4,14E-01	2,10E+01 ± 8,50E-01	2,10E+01 ± 3,27E-01	2,18E+01 ± 3,53E-01	2,15E+01 ± 7,90E-01	2,14E+01 ± 4,13E-01	2,24E+01 ± 3,08E-01	1,90E+01 ± 5,92E-01	1,87E+01 ± 5,68E-01
$F_{21}$	8,17E+02 ± 4,12E+02	7,95E+02 ± 4,03E+02	4,35E+02 ± 4,06E+02	9,28E+02 ± 3,17E+02	3,99E+02 ± 3,83E+02	7,60E+02 ± 3,99E+02	7,91E+02 ± 3,86E+02	2,90E+02 ± 2,77E+02	9,99E+02 ± 1,43E+02	8,55E+02 ± 3,82E+02
$F_{22}$	7,46E+01 ± 4,00E+01	9,95E+01 ± 8,88E+01	1,83E+03 ± 5,61E+02	2,09E+03 ± 4,97E+02	4,49E+02 ± 1,00E+02	9,00E+02 ± 5,62E+02	1,28E+04 ± 5,79E+02	6,14E+03 ± 2,71E+03	5,03E+01 ± 1,05E+01	2,29E+03 ± 2,73E+02
$F_{23}$	9,01E+03 ± 5,86E+02	8,97E+03 ± 6,66E+02	6,81E+03 ± 8,82E+02	1,30E+04 ± 4,35E+02	1,19E+04 ± 5,87E+02	1,32E+04 ± 8,73E+02	1,38E+04 ± 3,73E+02	1,38E+04 ± 6,01E+02	4,83E+03 ± 5,97E+02	4,93E+03 ± 6,43E+02
$F_{24}$	2,23E+02 ± 8,51E+00	2,22E+02 ± 6,17E+00	2,39E+02 ± 1,01E+01	2,39E+02 ± 7,21E+00	2,46E+02 ± 1,44E+01	2,77E+02 ± 4,14E+01	2,48E+02 ± 1,53E+01	2,53E+02 ± 2,92E+01	2,30E+02 ± 1,09E+01	2,28E+02 ± 6,70E+00
$F_{25}$	3,58E+02 ± 1,40E+01	3,61E+02 ± 1,79E+01	2,98E+02 ± 8,54E+00	3,24E+02 ± 2,85E+01	3,19E+02 ± 2,91E+01	3,79E+02 ± 8,63E+00	2,98E+02 ± 1,08E+01	3,92E+02 ± 3,27E+01	3,04E+02 ± 1,06E+01	3,09E+02 ± 1,65E+01
$F_{26}$	3,02E+02 ± 1,03E+02	2,79E+02 ± 9,96E+01	2,77E+02 ± 7,72E+01	2,06E+02 ± 2,90E+01	2,00E+02 ± 5,03E-02	3,01E+02 ± 1,13E+02	3,26E+02 ± 4,76E+01	2,99E+02 ± 1,28E+02	2,44E+02 ± 7,16E+01	2,20E+02 ± 4,58E+01

TABELA 43 – RESULTADOS OBTIDOS PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013

(conclusão)

	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
$F_{27}$	1,21E+03 ± 4,09E+02	1,05E+03 ± 4,57E+02	9,10E+02 ± 1,00E+02	8,31E+02 ± 8,77E+01	1,57E+03 ± 1,66E+02	1,65E+03 ± 1,84E+02	7,97E+02 ± 1,15E+02	1,81E+03 ± 2,93E+02	9,64E+02 ± 1,17E+02	7,16E+02 ± 1,14E+02
$F_{28}$	4,57E+02 ± 4,08E+02	4,58E+02 ± 4,17E+02	4,00E+02 ± 1,39E-14	4,00E+02 ± 0,00E+00	4,00E+02 ± 5,39E-14	4,00E+02 ± 9,55E-14	7,53E+02 ± 9,75E+02	4,00E+02 ± 1,01E-04	4,00E+02 ± 1,97E-14	4,00E+02 ± 1,61E-14

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do erro da função objetivo calculados sobre 50 experimentos, sendo o resultado de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 500.000.

A precisão dos dados apresentados nesta tabela é de 4,9407E-324.

TABELA 44 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	+	≈	+
	$F_2$	+	+	+	+	+	+	+	+
	$F_3$	≈	≈	+	+	+	+	+	+
	$F_4$	≈	+	+	+	+	+	+	+
	$F_5$	≈	≈	+	+	+	+	+	+
<i>Funções Multimodais Básicas</i>	$F_6$	≈	+	+	+	+	+	+	+
	$F_7$	≈	≈	+	+	+	+	+	+
	$F_8$	≈	-	≈	≈	≈	≈	≈	≈
	$F_9$	+	+	≈	+	+	+	≈	+
	$F_{10}$	≈	+	≈	+	≈	+	+	+
	$F_{11}$	+	+	+	+	-	+	+	+
	$F_{12}$	+	+	+	+	+	+	+	+
	$F_{13}$	+	+	+	+	+	+	+	+
	$F_{14}$	-	-	+	+	+	≈	+	+
	$F_{15}$	+	+	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	-	-	+	+	+	-	+	+
	$F_{18}$	+	+	+	+	+	+	+	+
	$F_{19}$	+	+	+	+	+	+	+	+
$F_{20}$	+	+	+	+	+	+	+	+	
<i>Funções Multimodais Compostas</i>	$F_{21}$	-	-	-	+	-	≈	≈	-
	$F_{22}$	+	+	+	+	+	+	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	-	-	+	+	+	+	+	+
	$F_{25}$	+	+	+	+	≈	+	-	+
	$F_{26}$	+	+	≈	-	-	+	+	+
	$F_{27}$	+	≈	-	-	+	+	-	+
	$F_{28}$	-	-	≈	-	-	+	+	+
<b>Total</b>	≈	8	5	6	2	4	3	4	1
	-	5	6	3	3	4	1	2	1
	+	15	17	19	23	20	24	22	26

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

TABELA 45 – RESULTADOS DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013 CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Funções Unimodais</i>	$F_1$	≈	≈	≈	≈	≈	+	≈	+
	$F_2$	≈	+	+	+	+	+	+	+
	$F_3$	≈	≈	+	+	+	+	+	+
	$F_4$	-	+	+	+	+	+	+	+
	$F_5$	≈	≈	+	+	+	+	+	+
<i>Funções Multimodais Básicas</i>	$F_6$	≈	+	+	+	+	+	+	+
	$F_7$	-	≈	+	+	≈	+	+	+
	$F_8$	≈	≈	≈	≈	≈	≈	≈	≈
	$F_9$	+	+	≈	+	+	+	≈	+
	$F_{10}$	≈	+	≈	+	+	+	+	+
	$F_{11}$	-	-	-	-	-	-	+	+
	$F_{12}$	+	+	+	+	+	+	+	+
	$F_{13}$	+	+	+	+	+	+	+	+
	$F_{14}$	-	-	-	-	-	-	+	+
	$F_{15}$	+	+	+	+	+	+	+	+
	$F_{16}$	+	+	+	+	+	+	+	+
	$F_{17}$	-	-	+	-	-	-	+	+
	$F_{18}$	+	+	+	+	+	+	+	+
	$F_{19}$	≈	≈	+	+	+	+	+	+
$F_{20}$	+	+	+	+	+	+	+	+	
<i>Funções Multimodais Compostas</i>	$F_{21}$	≈	≈	-	+	-	≈	≈	-
	$F_{22}$	-	-	-	-	-	-	+	+
	$F_{23}$	+	+	+	+	+	+	+	+
	$F_{24}$	-	-	+	+	+	+	+	+
	$F_{25}$	+	+	-	+	≈	+	-	+
	$F_{26}$	+	+	+	+	+	+	+	+
	$F_{27}$	+	+	+	+	+	+	+	+
	$F_{28}$	-	-	≈	-	-	+	+	+
<b>Total</b>	≈	9	7	5	2	4	2	4	1
	-	8	6	5	5	6	4	1	1
	+	11	15	18	21	18	22	23	26

FONTE: O autor (2014)

NOTA: Os sinais de "+", "≈" e "-" indicam se o algoritmo de controle tem melhor, similar ou pior desempenho respectivamente do que o algoritmo de comparação para a função de referência correspondente de acordo com o teste de *Mann-Whitney* considerando um nível de significância de 0,05.

## 6.6 ANÁLISE GERAL DOS RESULTADOS EXPERIMENTAIS

A fim de realizar uma análise comparativa do desempenho das duas versões de algoritmo *HSADE* na solução dos três estudos de caso, as TABELAS 46 a 59 foram construídas de forma a sumarizar os resultados obtidos na seção 6.3.2, 6.4.2 e 6.5.2 para cada um dos dois algoritmos *HSADE-v1* e *HSADE-v2*.

As TABELAS 46 a 52 sumarizam os resultados obtidos pelo teste de *Mann-Whitney* em cada um dos três estudos de caso. Mais especificamente, as TABELAS 46, 47 e 48 apresentam o comparativo do número total de funções nas dimensões 10, 30 e 50, respectivamente, para os quais cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* apresentaram um desempenho melhor (+), similar ( $\approx$ ) ou pior (-) quando comparado a cada um dos oito algoritmos de comparação na solução do estudo de caso formado pelo conjunto de funções do CEC2005, enquanto a TABELA 49 apresenta o comparativo do número total de problemas para os quais cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* apresentou um desempenho melhor (+), similar ( $\approx$ ) ou pior (-) quando comparado com cada um dos oito algoritmos de comparação na solução do estudo de caso formado pelo conjunto de problemas de otimização do CEC2011. Já as TABELAS 50, 51 e 52 apresentam o comparativo do número total de funções nas dimensões 10, 30 e 50, respectivamente, para os quais cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* apresentaram um desempenho melhor (+), similar ( $\approx$ ) ou pior (-) quando comparado a cada um dos oito algoritmos de comparação na solução do estudo de caso formado pelo conjunto de funções do CEC2013.

As TABELAS 53 a 59 sumarizam os resultados obtidos pelo teste *post hoc* de *Friedman*. Mais especificamente, as TABELAS 53, 54 e 55 apresentam o comparativo dos valores-p ajustados para os casos de dimensão 10, 30 e 50, respectivamente, obtidos para cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* na solução do estudo de caso formado pelo conjunto de funções do CEC2005. A TABELA 56 apresenta o comparativo dos valores-p ajustados obtidos para cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* na solução do estudo de caso formado pelo conjunto de problemas de otimização do CEC2011 e As TABELAS 57, 58 e 59 apresentam o comparativo dos valores-p ajustados para os casos de dimensão 10, 30 e 50, respectivamente, obtidos para cada um dos algoritmos de controle *HSADE-v1* e *HSADE-v2* na solução do estudo de caso

formado pelo conjunto de funções do CEC2013. Além disso, nas TABELAS 53 a 59, os valores-p ajustados menores que um nível de significância  $\alpha = 0,12$  foram destacados.

A partir destes resultados, algumas considerações podem ser realizadas.

Primeiramente, com relação ao teste de *Mann-Whitney*, pode-se observar que nenhuma das duas versões é capaz de ter um desempenho superior do que todos os oito algoritmos de comparação em todas as funções testes em qualquer um dos três estudos de caso, o que é normal e esperado. Entretanto, analisando-se de uma maneira global os resultados obtidos nos três estudos de caso, os dois algoritmos HSADE-v1 e HSADE-v2 apresentaram um bom desempenho. Em especial, nota-se o excelente desempenho apresentado na solução dos problemas de otimização do CEC2011, no qual a quantidade de problemas nos quais HSADE-v1 e HSADE-v2 apresentaram desempenho superior aos algoritmos de comparação foi bem elevada.

Outro ponto importante que é possível verificar em relação aos resultados obtidos no teste de *Mann-Whitney*, é que, no geral, as duas versões apresentaram um desempenho bem similar entre elas, visto que o número total de funções nas quais cada um dos algoritmos HSADE-v1 e HSADE-v2 foram melhores (+), similares ( $\approx$ ) ou piores (-) em relação aos algoritmos de comparação são parecidos.

Com relação ao teste *post hoc* de *Friedman*, nota-se a partir dos dados apresentados nas TABELAS 53 a 59, que embora HSADE-v1 e HSADE-v2 não tenham sido superior em todos os casos, fazendo-se uma análise global, os resultados obtidos através do teste *post hoc* de *Friedman* demonstraram que os algoritmos propostos HSADE-v1 e HSADE-v2 tiveram um bom desempenho.

Além disso, analisando-se especificamente os resultados do teste *post hoc* de *Friedman* obtidos no estudo de caso formado pelo conjunto de funções do CEC2005, verifica-se que o desempenho do algoritmo HSADE-v1 foi ligeiramente superior ao algoritmo HSADE-v2 neste estudo de caso. Primeiramente, analisando-se os valores-p ajustados menores que um nível de significância  $\alpha = 0,12$  (valores destacados nas tabelas). nota-se que para o caso de 10 dimensões o algoritmo HSADE-v1 obteve um melhor resultado já que o teste *post hoc* de *Friedman* mostra uma melhoria do algoritmo HSADE-v1 sobre todos os algoritmos de comparação enquanto HSADE-v2 apresentou uma melhoria apenas sobre CoDE, jDE e JADE *sem arquivo*. Já para caso de 30 e 50 dimensões, o teste *post hoc* de *Friedman*



mostrou uma melhoria de ambos os algoritmos, HSADE-v1 e HSADE-v2, sobre DEGL, EPSDE, DEahcSPX, SaDE, jDE e JADE *sem arquivo* para o caso de 30 dimensões e sobre DEahcSPX, EPSDE, SaDE, DEGL e jDE para o caso de 50 dimensões. Entretanto, fazendo-se uma análise dos valores absolutos dos valores-p ajustados, observa-se que HSADE-v1 obteve menores valores-p ajustados, ou seja, HSADE-v1 apresenta uma maior evidência para rejeitar a hipótese nula.

Já com relação aos resultados do teste *post hoc* de *Friedman* obtidos no estudo de caso formado pelo conjunto de problemas de otimização do CEC2011, verifica-se, diferentemente do que havia sido observado no primeiro estudo no caso, que o desempenho do algoritmo HSADE-v2 foi ligeiramente superior ao algoritmo HSADE-v1 neste estudo de caso. Primeiramente, analisando-se os valores-p ajustados menores que um nível de significância  $\alpha = 0,12$  (valores destacados nas tabelas), nota-se que o algoritmo HSADE-v2 obteve um melhor resultado já que o teste *post hoc* de *Friedman* mostra uma melhoria do algoritmo HSADE-v2 sobre todos os algoritmos de comparação enquanto HSADE-v1 apresentou uma melhoria apenas sobre CoDE, jDE, JADE *sem arquivo*, DEGL, DEahcSPX e EPSDE. Além disso, fazendo-se uma análise dos valores absolutos dos valores-p ajustados, observa-se que HSADE-v2 obteve menores valores-p ajustados, ou seja, HSADE-v2 apresenta uma maior evidência para rejeitar a hipótese nula.

Com relação aos resultados do teste *post hoc* de *Friedman* obtidos no estudo de caso formado pelo conjunto de funções do CEC2013, pode-se dizer que ambos os algoritmos HSADE-v1 e HSADE-v2 tiveram desempenho semelhante, isto porque no caso de dimensão 10 observa-se HSADE-v2 foi ligeiramente superior ao algoritmo HSADE-v1 enquanto que no caso de dimensão 30 e 50 foi o contrário. Primeiramente, analisando-se os valores-p ajustados menores que um nível de significância  $\alpha = 0,12$  (valores destacados nas tabelas), nota-se que para o caso de 10 dimensões o algoritmo HSADE-v2 obteve um melhor resultado já que o teste *post hoc* de *Friedman* mostra uma melhoria do algoritmo HSADE-v2 sobre CoDE e jDE enquanto HSADE-v1 apresentou uma melhoria apenas sobre CoDE. Já para caso de 30 e 50 dimensões, o teste *post hoc* de *Friedman* mostrou uma melhoria de ambos os algoritmos, HSADE-v1 e HSADE-v2, sobre DEGL, EPSDE, DEahcSPX, SaDE, jDE. Entretanto, fazendo-se uma análise dos valores absolutos dos valores-p ajustados, observa-se que HSADE-v1 obteve menores valores-p ajustados, ou seja, HSADE-v1 apresenta uma maior evidência para rejeitar a hipótese nula.

Em suma, de acordo com os resultados obtidos através do teste de *Mann-Whitney* e do teste *post hoc* de *Friedman* nota-se que os dois algoritmos HSADE-v1 e HSADE-v2 apresentaram um desempenho muito bom e consistente na solução dos problemas de otimização propostos no CEC2005, CEC2011 e CEC2013 quando comparado a JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, demonstrando a eficácia e a eficiência dos algoritmos propostos. Em especial nota-se que para determinados casos o algoritmo HSADE-v1 apresentou melhor desempenho e que para outros casos o algoritmo HSADE-v2 apresentou melhor desempenho, mas, fazendo-se uma análise de maneira global, pode-se dizer que o desempenho dos dois algoritmos HSADE-v1 e HSADE-v2 foram parecidos. Não se observou uma clara vantagem, no geral, em termos de desempenho quando se compara os resultados obtidos por ambos os algoritmos.

Este bom resultado apresentado pelas duas versões do algoritmo *HSADE* se deve ao fato destes algoritmos apresentarem um bom compromisso entre *exploration* e *exploitation* o que faz com que eles sejam capazes de explorar o máximo possível do espaço de busca ao mesmo tempo em que são capazes de ter uma boa velocidade de convergência. Soma-se a isto ainda o fato das duas versões do algoritmo *HSADE* serem capazes de adaptar seus parâmetros de controle ao longo do processo de otimização fazendo com os parâmetros sejam adaptados adequadamente para cada problema o que contribui ainda mais na velocidade de convergência ao mesmo tempo em que diminui a probabilidade de o algoritmo ficar estagnado em um mínimo local.

Adicionalmente aos resultados apresentados nesta seção, nos APÊNDICES A, B e C são apresentados os gráficos de convergência para todos os algoritmos (JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1, HSADE-v2).

Mais especificamente, as FIGURAS 67 à 78 (APÊNDICE A) ilustram o gráfico de convergência para todos os algoritmos em cada uma das vinte e cinco funções teste do CEC2005 nas três dimensões analisadas, enquanto as FIGURAS de 79 à 82 (APÊNDICE B) ilustram o gráfico de convergência para todos os algoritmos em cada um dos vinte e dois problemas de otimização do CEC2011. Já as FIGURAS de 83 a 97 (APÊNDICE C) ilustram o gráfico de convergência para todos os algoritmos em cada uma das vinte e oito funções teste do CEC2013 nas

três dimensões analisadas. Nestes gráficos pode-se observar a evolução do valor da função objetivo em função do número de avaliações da função objetivo.

TABELA 46 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	12	12	4	14	9	9	16	7
	HSADE-v2	16	10	4	10	11	6	15	3
-	HSADE-v1	3	1	0	1	1	3	2	4
	HSADE-v2	3	1	0	2	1	4	2	6
+	HSADE-v1	10	12	21	10	15	13	7	14
	HSADE-v2	6	14	21	13	13	15	8	16

FONTE: O autor (2014)

TABELA 47 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	8	6	7	5	6	5	7	4
	HSADE-v2	11	8	6	4	10	6	6	3
-	HSADE-v1	3	1	2	0	1	0	2	1
	HSADE-v2	5	2	4	0	1	0	3	3
+	HSADE-v1	14	18	16	20	18	25	16	20
	HSADE-v2	9	15	15	21	14	19	16	19

FONTE: O autor (2014)

TABELA 48 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	10	6	5	3	4	2	3	1
	HSADE-v2	5	7	3	3	3	2	4	1
-	HSADE-v1	6	1	5	0	4	2	0	3
	HSADE-v2	12	8	5	1	5	3	2	3
+	HSADE-v1	9	18	15	22	17	21	22	21
	HSADE-v2	8	10	17	21	17	20	19	21

FONTE: O autor (2014)

TABELA 49 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA OS PROBLEMAS DO CEC2011

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	12	12	7	8	4	1	6	1
	HSADE-v2	10	8	6	4	3	1	6	1
-	HSADE-v1	0	0	6	3	1	0	1	0
	HSADE-v2	0	0	6	2	0	0	0	0
+	HSADE-v1	10	10	9	11	17	21	15	21
	HSADE-v2	12	14	10	16	19	21	16	21

FONTE: O autor (2014)

TABELA 50 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	15	15	1	8	10	10	9	7
	HSADE-v2	16	15	1	11	9	11	8	8
-	HSADE-v1	8	5	1	10	4	6	9	6
	HSADE-v2	6	5	1	6	4	4	7	4
+	HSADE-v1	5	8	26	10	14	12	10	15
	HSADE-v2	6	8	26	11	15	13	13	16

FONTE: O autor (2014)

TABELA 51 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	11	8	5	6	8	7	8	5
	HSADE-v2	11	9	7	4	8	6	10	4
-	HSADE-v1	4	4	2	0	2	2	2	1
	HSADE-v2	6	5	5	3	5	4	1	1
+	HSADE-v1	13	16	21	22	18	19	18	22
	HSADE-v2	11	14	16	21	15	18	17	23

FONTE: O autor (2014)

TABELA 52 – RESULTADO COMPARATIVO DO TESTE DE *MANN-WHITNEY* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013

		JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
≈	HSADE-v1	8	5	6	2	4	3	4	1
	HSADE-v2	9	7	5	2	4	2	4	1
-	HSADE-v1	5	6	3	3	4	1	2	1
	HSADE-v2	8	6	5	5	6	4	1	1
+	HSADE-v1	15	17	19	23	20	24	22	26
	HSADE-v2	11	15	18	21	18	22	23	26

FONTE: O autor (2014)

TABELA 53 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2005

	HSADE-V1	HSADE-V2
CoDE	6,39E-09	1,96E-08
jDE	0,043	0,099
JADE <i>sem arquivo</i>	0,074	0,120
DEGL	0,088	0,161
DEahcSPX	0,113	0,165
EPSDE	0,113	0,165
SaDE	0,113	0,177
JADE <i>com arquivo</i>	0,113	0,177

FONTE: O autor (2014)

TABELA 54 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2005

	HSADE-V1	HSADE-V2
DEGL	1,02E-04	4,03E-04
EPSDE	5,46E-04	0,001
DEahcSPX	0,001	0,002
SaDE	0,001	0,003
jDE	0,003	0,009
JADE <i>sem arquivo</i>	0,015	0,032
CoDE	0,158	0,243
JADE <i>com arquivo</i>	0,158	0,302

FONTE: O autor (2014)

TABELA 55 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2005

	HSADE-V1	HSADE-V2
DEahcSPX	1,25E-07	4,38E-06
EPSDE	1,48E-07	5,00E-06
SaDE	4,22E-05	0,001
DEGL	3,90E-04	0,008
jDE	0,015	0,113
CoDE	0,328	0,98
JADE <i>sem arquivo</i>	0,76	1
JADE <i>com arquivo</i>	1	1

FONTE: O autor (2014)

TABELA 56 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA OS PROBLEMAS DO CEC2011

	HSADE-V1	HSADE-V2
CoDE	2,95E-12	1,58E-13
jDE	2,70E-08	2,44E-09
JADE <i>sem arquivo</i>	1,05E-04	1,73E-05
DEGL	1,12E-04	2,45E-05
DEahcSPX	0,072	0,014
EPSDE	0,109	0,046
SaDE	0,139	0,064
JADE <i>com arquivo</i>	0,139	0,064

FONTE: O autor (2014)

TABELA 57 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 10 DIMENSÕES PROPOSTAS NO CEC2013

	HSADE-V1	HSADE-V2
CoDE	1,13E-04	1,62E-06
jDE	0,497	0,033
JADE <i>sem arquivo</i>	1	0,203
DEGL	1	0,564
DEahcSPX	1	1
EPSDE	1	1
SaDE	1	1
JADE <i>com arquivo</i>	1	1

FONTE: O autor (2014)

TABELA 58 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 30 DIMENSÕES PROPOSTAS NO CEC2013

	HSADE-V1	HSADE-V2
DEGL	1,42E-06	8,51E-06
EPSDE	9,52E-06	5,01E-05
DEahcSPX	1,90E-05	6,03E-05
SaDE	8,78E-05	2,08E-04
jDE	0,005	0,016
JADE <i>sem arquivo</i>	0,136	0,291
CoDE	0,261	0,375
JADE <i>com arquivo</i>	0,261	0,375

FONTE: O autor (2014)

TABELA 59 – RESULTADO COMPARATIVO DO TESTE *POST HOC* DE *FRIEDMAN* PARA AS FUNÇÕES TESTES DE 50 DIMENSÕES PROPOSTAS NO CEC2013

	HSADE-V1	HSADE-V2
DEahcSPX	1,56E-10	8,09E-10
EPSDE	1,09E-06	2,39E-06
SaDE	1,50E-05	7,55E-05
DEGL	6,40E-04	0,0017
jDE	0,048	0,117
CoDE	0,516	0,785
JADE <i>sem arquivo</i>	0,545	0,786
JADE <i>com arquivo</i>	0,545	0,786

FONTE: O autor (2014)

## 6.7 ANÁLISE SOBRE A ADAPTAÇÃO DOS PARÂMETROS DOS ALGORITMOS PROPOSTOS

Nesta seção será apresentada uma análise mais detalhada do comportamento da adaptação de alguns parâmetros utilizados pelos algoritmos HSADE-v1 e HSADE-v2

### 6.7.1 Algoritmo HSADE-v1

Conforme detalhado na seção 5.1.2.1, o algoritmo HSADE-v1 aplica um processo de auto-adaptação tanto para a escolha da estratégia de geração dos

vetores experimentais quanto para a determinação dos parâmetros de controle do fator de mutação e taxa de cruzamento.

A FIGURA 14 ilustra a evolução da probabilidade média ao longo do processo de otimização de cada uma das estratégias de geração dos vetores experimentais para o caso de quatro funções do CEC2005 ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) para  $D = 30$ .

Observa-se que para  $F_4$ , a estratégia *DE/target-to-pbest/1/bin* é a principal estratégia de geração de vetor experimental. Desde o início,  $p_1$  converge para um valor próximo de 0,9 e assim permanece até o final do processo de otimização. Este comportamento é explicado pelo fato de  $F_4$  ser uma função unimodal e portanto a estratégia *DE/target-to-pbest/1/bin* é bem mais eficiente que a estratégia *DE/rand/2/bin* por convergir mais rapidamente para o ótimo global.

Para  $F_6$ , que é uma função multimodal básica, nota-se um comportamento similar. Entretanto, embora *DE/target-to-pbest/1/bin* seja a principal estratégia, *DE/rand/2/bin* possui uma probabilidade maior (um valor de aproximadamente 0,2) quando comparado com o caso unimodal. Este comportamento é coerente, pois  $F_6$  é uma função multimodal básica e devido a isto o algoritmo utiliza-se a estratégia *DE/target-to-pbest/1/bin* para obter uma convergência mais rápida, mas ao mesmo tempo utiliza-se também da estratégia *DE/rand/2/bin* que produz uma maior diversidade de vetores experimentais com maiores chances de escapar dos mínimos locais.

Por outro lado, nota-se que para outras duas funções,  $F_{14}$  e  $F_{16}$ , que são funções multimodais mais complexas, no início a estratégia *DE/target-to-pbest/1/bin* é a que tem o papel principal. Entretanto, à medida que o processo de otimização evolui, a probabilidade da estratégia *DE/rand/2/bin* aumenta, ou seja, mais vetores experimentais são produzidos pela estratégia *DE/rand/2/bin*. Este comportamento é coerente pois no início do processo de otimização, a diversidade da população é grande e a estratégia *DE/target-to-pbest/1/bin* é a mais eficiente para convergir a população rapidamente. Entretanto, à medida que a população converge, a diversidade da população diminui e portanto é mais difícil de escapar de um mínimo local. Neste sentido, a estratégia *DE/rand/2/bin* tem um papel importante pois ela gera vetores experimentais capazes de aumentar a diversidade da população aumentando, portanto, as chances de alcançar o mínimo global.



Isto mostra a capacidade de adaptação da melhor estratégia de geração dos vetores experimentais em função do problema a ser resolvido, comprovando que o algoritmo HSADE-v1 é capaz de obter um melhor balanço entre *exploration* e *exploitation*.

O algoritmo HSADE-v1 também emprega um processo de adaptação dos parâmetros de controle  $F$  e  $CR$ , conforme descrito na seção 5.1.2.2. A FIGURA 15 ilustra a evolução da média dos parâmetros  $\mu_{F,1}$ ,  $\mu_{F,2}$  e  $\mu_{CR,1}$ ,  $\mu_{CR,2}$ , utilizados na geração dos parâmetros de controle, ao longo do processo de otimização para o caso de quatro funções do CEC2005 ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) para  $D = 30$ .

Primeiramente, pode-se notar que  $\mu_{F,1} \neq \mu_{F,2}$  e  $\mu_{CR,1} \neq \mu_{CR,2}$ , ou seja, os parâmetros de fator de mutação e taxa de cruzamento são adaptados diferentemente para cada estratégia de geração dos vetores experimentais. Além disto, observa-se que para o caso unimodal (função  $F_4$ ) os parâmetros convergem rapidamente para um determinado valor e mantêm-se estáveis durante todo o processo de otimização. Já no caso das funções multimodais ( $F_6$ ,  $F_{14}$  e  $F_{16}$ ) os parâmetros evoluem para diferentes valores ao longo do processo de otimização.

Isto mostra que o algoritmo HSADE-v1 é capaz de obter uma melhor adaptação dos parâmetros em função do problema a ser resolvido e também durante diversas fases do processo de otimização.

Com relação à busca voltada à *exploitation*, na FIGURA 16 é apresentado a evolução da média do fator de ajuste  $\omega$  utilizado para ajustar os movimentos dos vetores experimentais gerados pela busca voltada à *exploitation*. Observa-se que à medida que a otimização evolui, este parâmetro é reduzido fazendo com que a busca voltada à *exploitation* explore o interior de uma vizinhança cada vez menor.

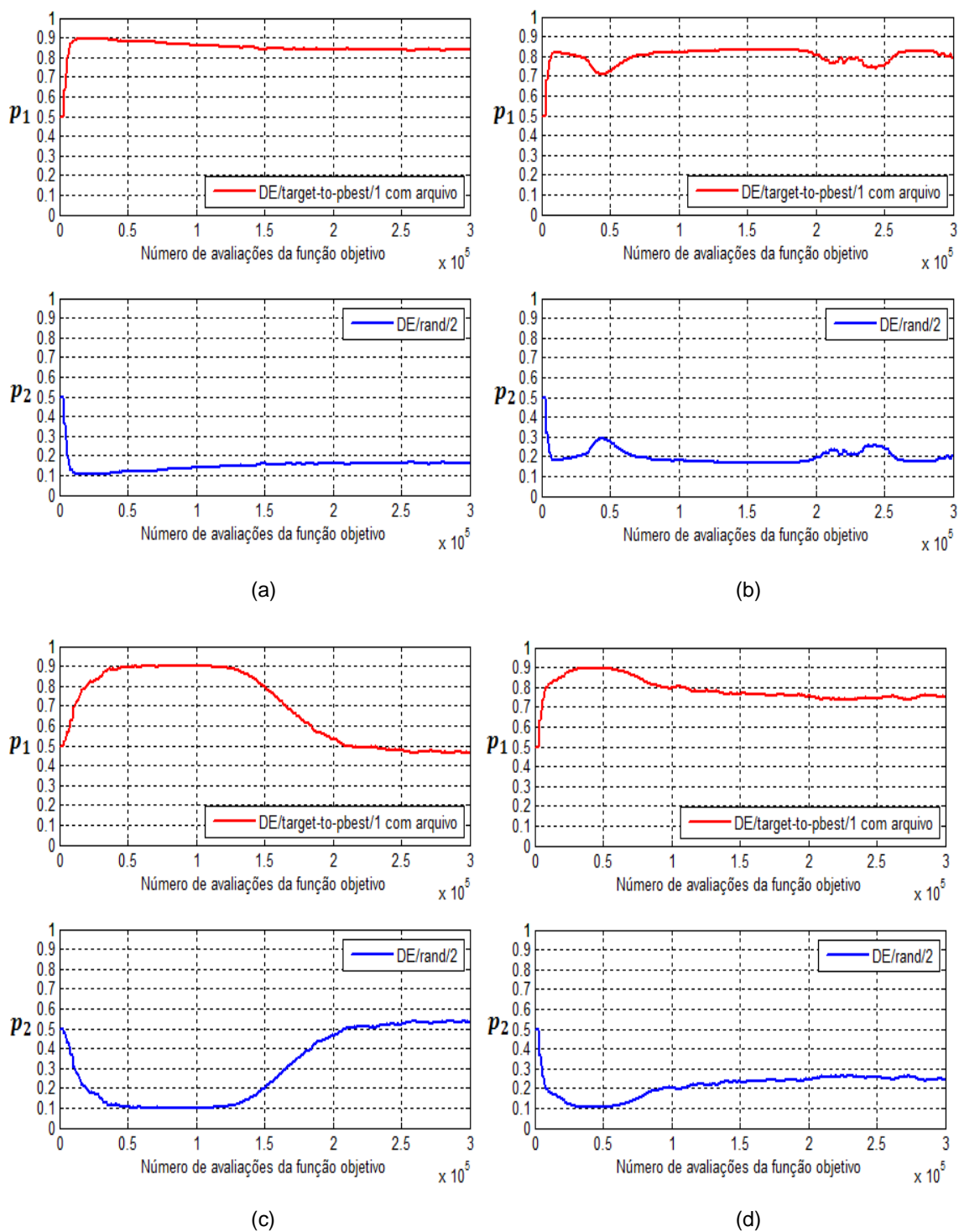


FIGURA 14: Evolução das probabilidades de cada uma das estratégias de geração de vetores experimentais para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)  $F_{16}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

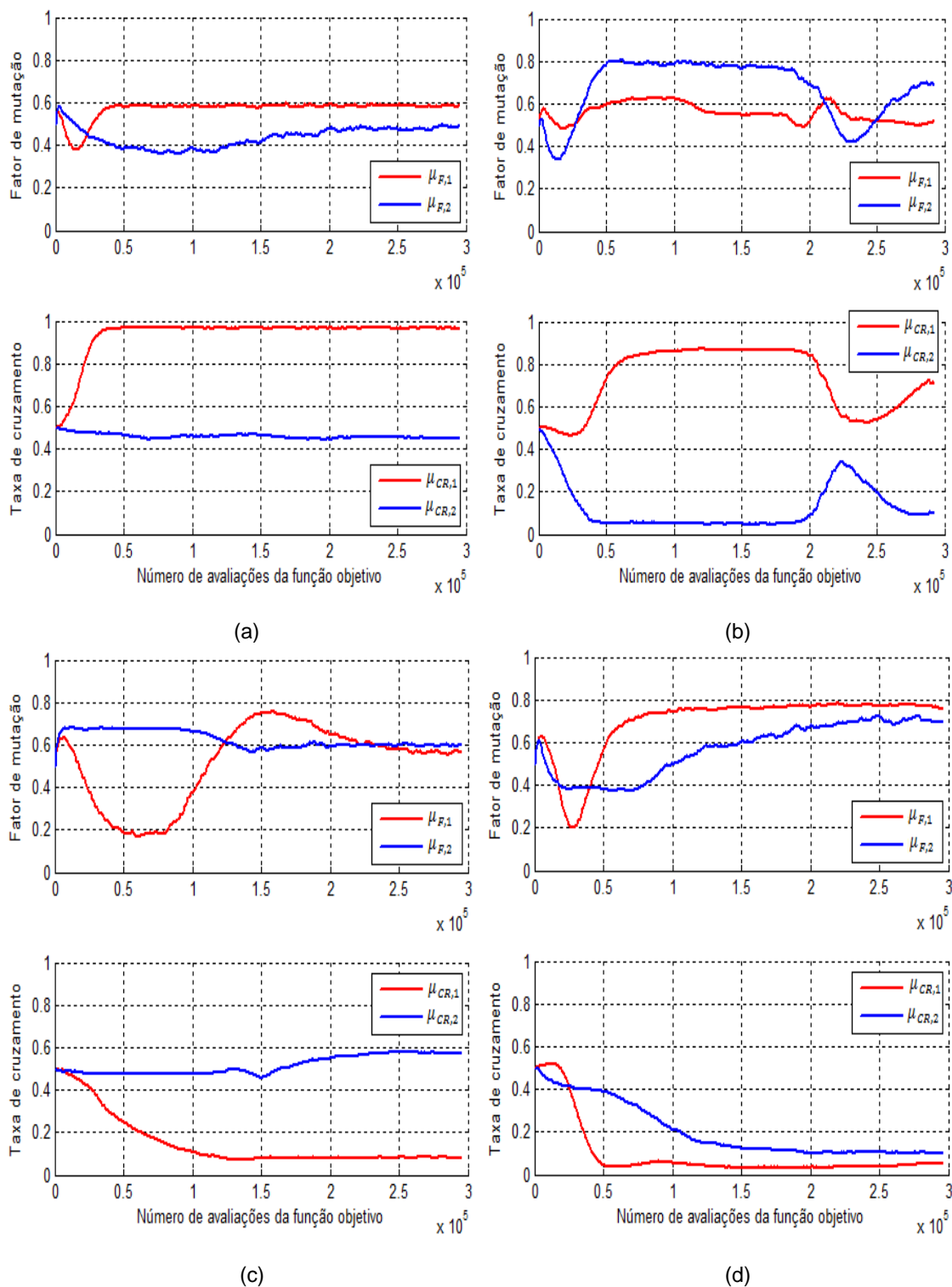


FIGURA 15: Evolução dos parâmetros  $\mu_{F,m}$  e  $\mu_{CR,m}$ ,  $m = 1, 2$ , para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)  $F_{16}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

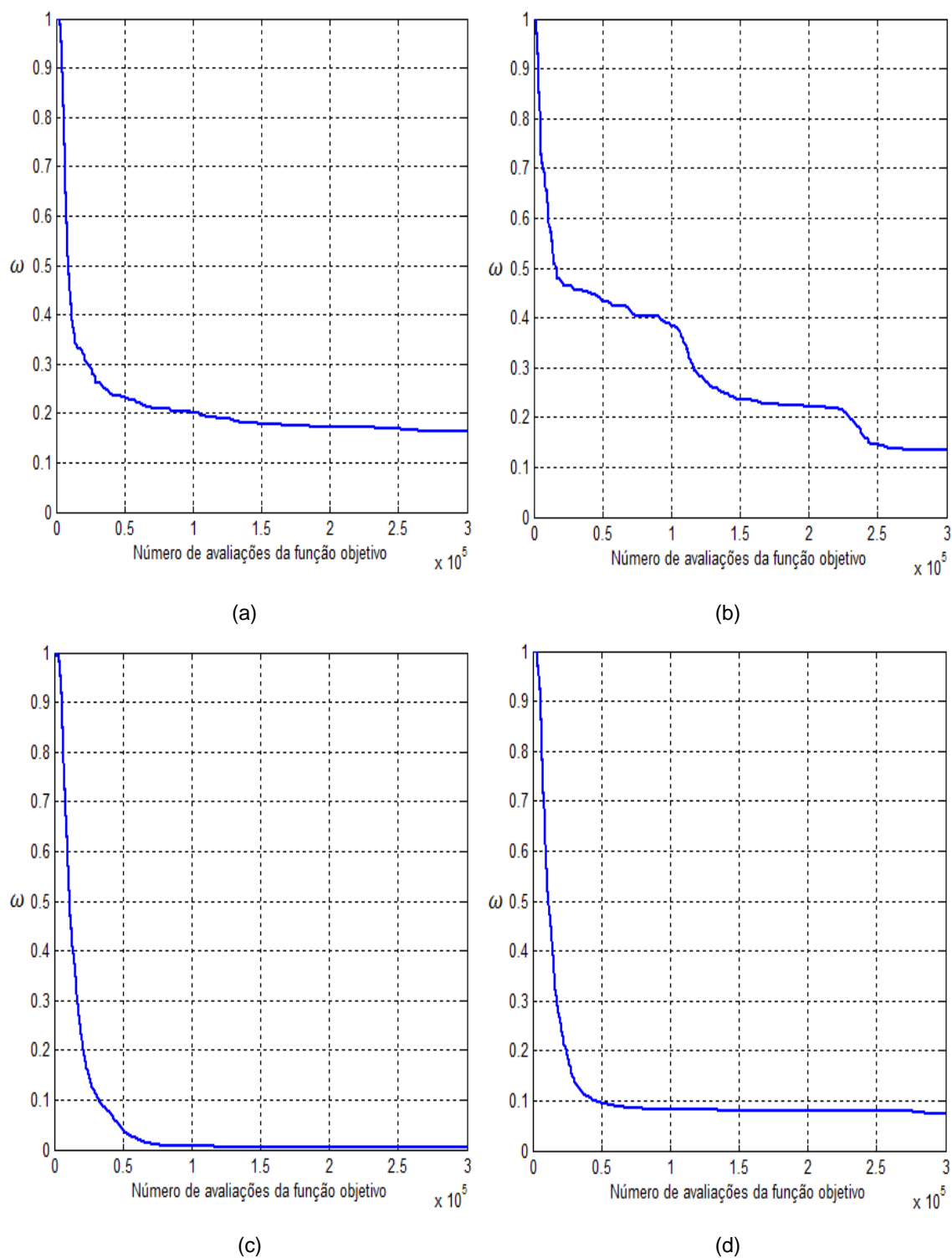


FIGURA 16: Evolução do fator de ajuste  $\omega$  para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)  $F_{16}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

### 6.7.2 Algoritmo HSADE-v2

Assim como na primeira versão, o algoritmo HSADE-v2 também aplica um processo de auto-adaptação dos parâmetros de controle  $F$  e  $CR$ , conforme descrito na seção 5.2.1.2. A FIGURA 17 ilustra a evolução da média os 50 experimentos dos parâmetros  $\mu_F$  e  $\mu_{CR}$  ao longo do processo de otimização para o caso de quatro funções do CEC2005 ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) para  $D = 30$ .

Primeiramente, pode-se notar a mesma tendência do processo de adaptação observada na primeira versão do algoritmo HSADE. Conforme ilustrado na FIGURA 17, verifica-se que para o caso unimodal (função  $F_4$ ) os parâmetros convergem rapidamente para um determinado valor e mantém-se estável durante todo o processo de otimização. Já no caso das funções multimodais ( $F_6$ ,  $F_{14}$  e  $F_{16}$ ) os parâmetros evoluem para diferentes valores ao longo do processo de otimização.

Outro ponto importante a ser observado está relacionado ao parâmetro  $\mu_{CR}$ . Conforme detalhado seção 5.2.1.2, a média generalizada com parâmetro  $\alpha = 2$  foi utilizada no processo de adaptação do parâmetro  $\mu_{CR}$  para reduzir uma tendência implícita para valores pequenos durante o processo de auto-adaptação que existe no algoritmo HSADE-v1. Analisando-se as FIGURAS 15 e 17, observa-se que  $\mu_{CR}$  atinge maiores valores no algoritmo HSADE-2 quando comparado ao algoritmo HSADE-v1, principalmente quando se compara a evolução de  $\mu_{CR}$  para as funções  $F_{14}$  e  $F_{16}$ . Isto é um comportamento vantajoso, visto que as funções exemplificadas nesta seção ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) são funções não separáveis para as quais maiores valores de taxas de cruzamento são os mais adequados.

Isto tudo demonstra que o algoritmo HSADE-v2 é capaz de obter uma adaptação adequada dos parâmetros de controle em função do problema a ser resolvido e também durante diversas fases do processo de otimização.

Com relação à busca voltada à *exploitation*, o algoritmo HSADE-v2 emprega um processo de adaptação dos parâmetros  $\beta$  e  $\omega$  conforme detalhado na seção 5.2.2.

Com relação ao parâmetro  $\beta$ , diferentemente do que é empregado na primeira versão do algoritmo HSADE, no algoritmo HSADE-v2 o parâmetro  $\beta$  é adaptado durante o processo de otimização, e este processo de adaptação pode ser verificado através da FIGURA 18, na qual é apresentada a evolução da média sobre

os 50 experimentos da probabilidade da busca voltada à *exploitation*  $\beta$  ao longo do processo de otimização para o caso de quatro funções do CEC2005 ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) para  $D = 30$ .

Nota-se que, para as funções  $F_4$  e  $F_6$ , a busca voltada à *exploitation* não é a principal estratégia na geração dos vetores experimentais ( $\beta < 0,3$ ), ocorrendo apenas um ligeiro aumento do parâmetro  $\beta$  a medida que o processo de otimização evolui. No caso destas funções, a busca voltada à *exploitation* não foi tão eficiente o que faz com que a probabilidade da busca voltada à *exploitation* seja baixa, aumentando, desta forma, a quantidade de vetores experimentais produzidos pela abordagem ED para não prejudicar a convergência do algoritmo

Já para as funções  $F_{14}$  e  $F_{16}$ , nota-se que no início, a busca voltada à *exploitation* não é a principal estratégia na geração dos vetores experimentais, mas a medida que o processo de otimização evolui, a busca voltada à *exploitation* passa a ter um peso maior na geração dos vetores experimentais. Apenas no final do processo de otimização ocorre uma diminuição parâmetro  $\beta$ . Isto demonstra que, nos momentos em que a busca voltada à *exploitation* é eficiente, o parâmetro  $\beta$  é aumentado, fazendo com que ocorra um aumento na quantidade de vetores experimentais produzidos pela busca voltada à *exploitation* com o intuito de aumentar a velocidade de convergência do algoritmo. Entretanto, a medida que a busca voltada à *exploitation* não é mais tão eficiente, diminui-se o parâmetro  $\beta$ , fazendo com que ocorra uma diminuição da quantidade de vetores experimentais produzidos pela busca voltada à *exploitation* para não prejudicar a convergência do algoritmo.

Isto tudo demonstra a capacidade de adaptação da melhor estratégia de geração dos vetores experimentais pelo algoritmo HSADE-v2 durante diversas fases do processo de otimização.

A FIGURA 19 ilustra evolução da média sobre os 50 experimentos do fator de ajuste  $\omega$  utilizado para ajustar os movimentos dos vetores experimentais gerados pela busca voltada à *exploitation* para as mesmas quatro funções do CEC2005 ( $F_4$ ,  $F_6$ ,  $F_{14}$  e  $F_{16}$ ) para  $D = 30$ .

Nota-se, primeiramente, que o parâmetro  $\omega$  é reduzido rapidamente no início do processo de otimização fazendo com que a busca voltada à *exploitation* explore o interior de uma vizinhança menor, ajustando, desta forma, o espaço de busca de tal

forma que a busca voltada à *exploitation* seja bem sucedida. Entretanto, enquanto na primeira versão do algoritmo *HSADE* o fator de ajuste é apenas reduzido, no algoritmo *HSADE-v2*, o fator de ajuste pode também ser aumentado. Isto fica claro nas funções  $F_{14}$  e  $F_{16}$  nas quais se verifica que ocorre um aumento do fator de ajuste durante o processo de otimização o que faz com que a busca voltada à *exploitation* explore o interior de uma maior vizinhança contribuindo, assim, para aumentar a velocidade de convergência.

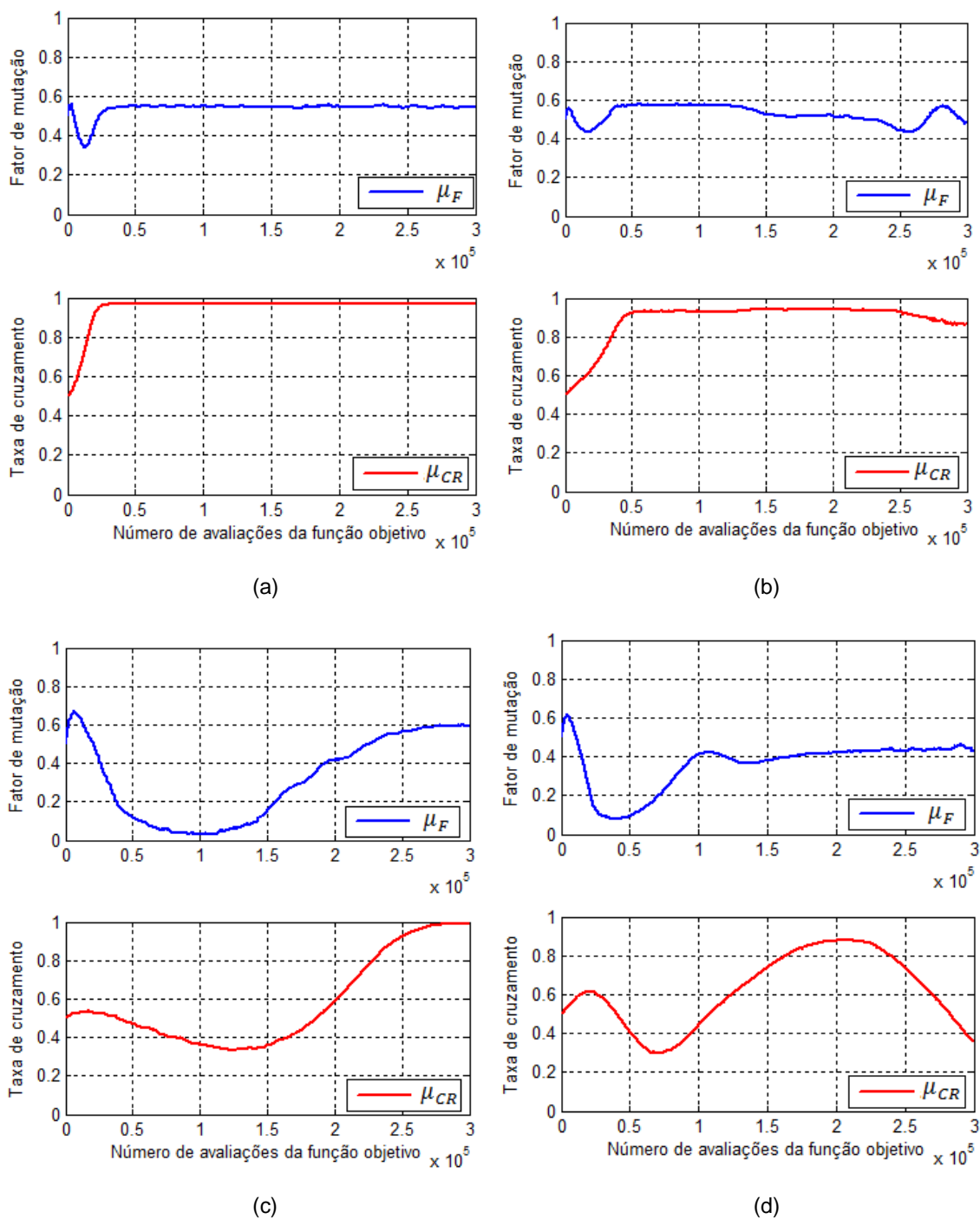


FIGURA 17: Evolução dos parâmetros  $\mu_F$  e  $\mu_{CR}$  para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)  $F_{16}$  para o caso  $D = 30$ .

FONTE: O autor (2014)



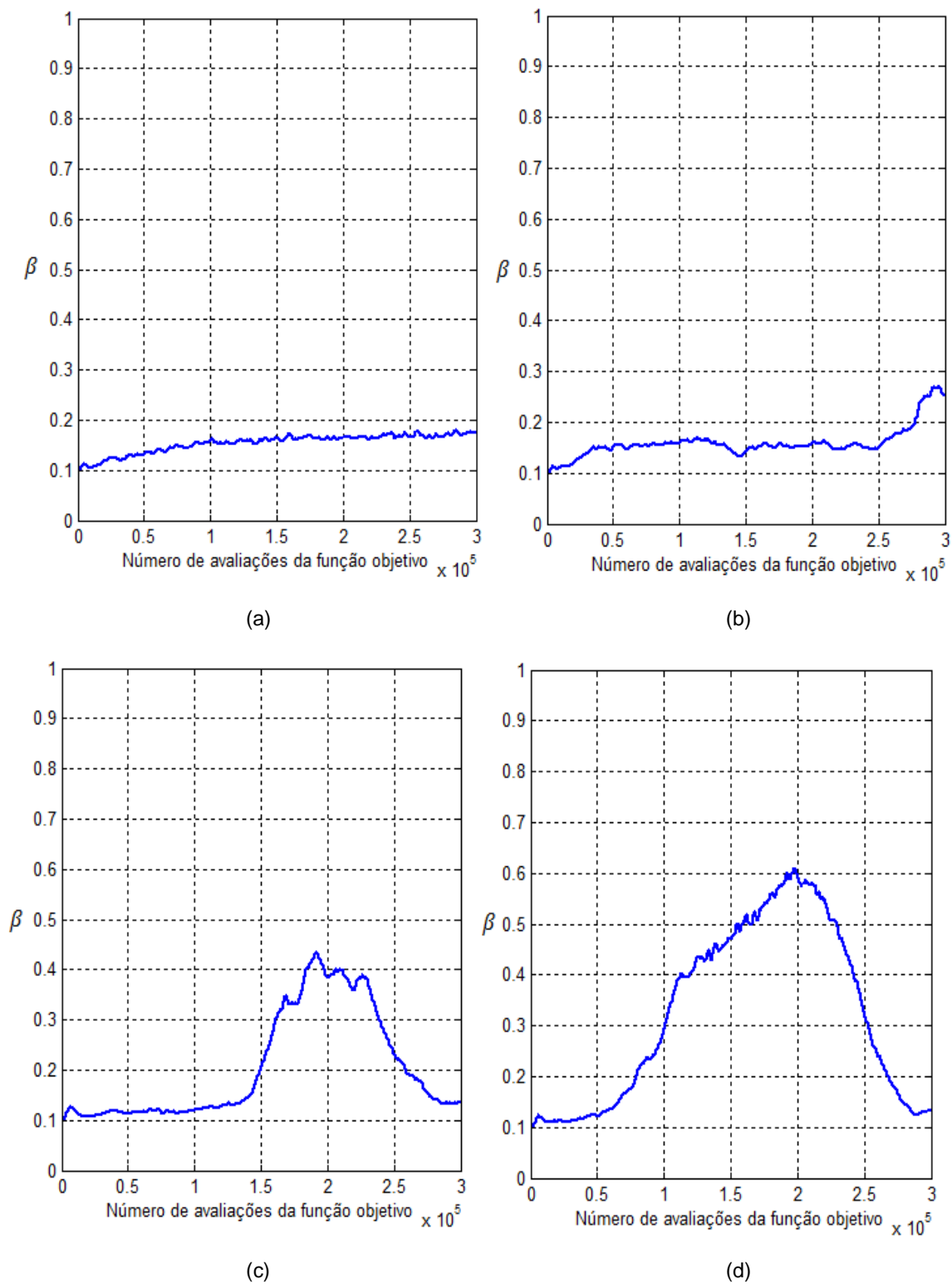


FIGURA 18: Evolução da probabilidade  $\beta$  para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)  $F_{16}$  para o caso  $D = 30$ .  
 FONTE: O autor (2014)

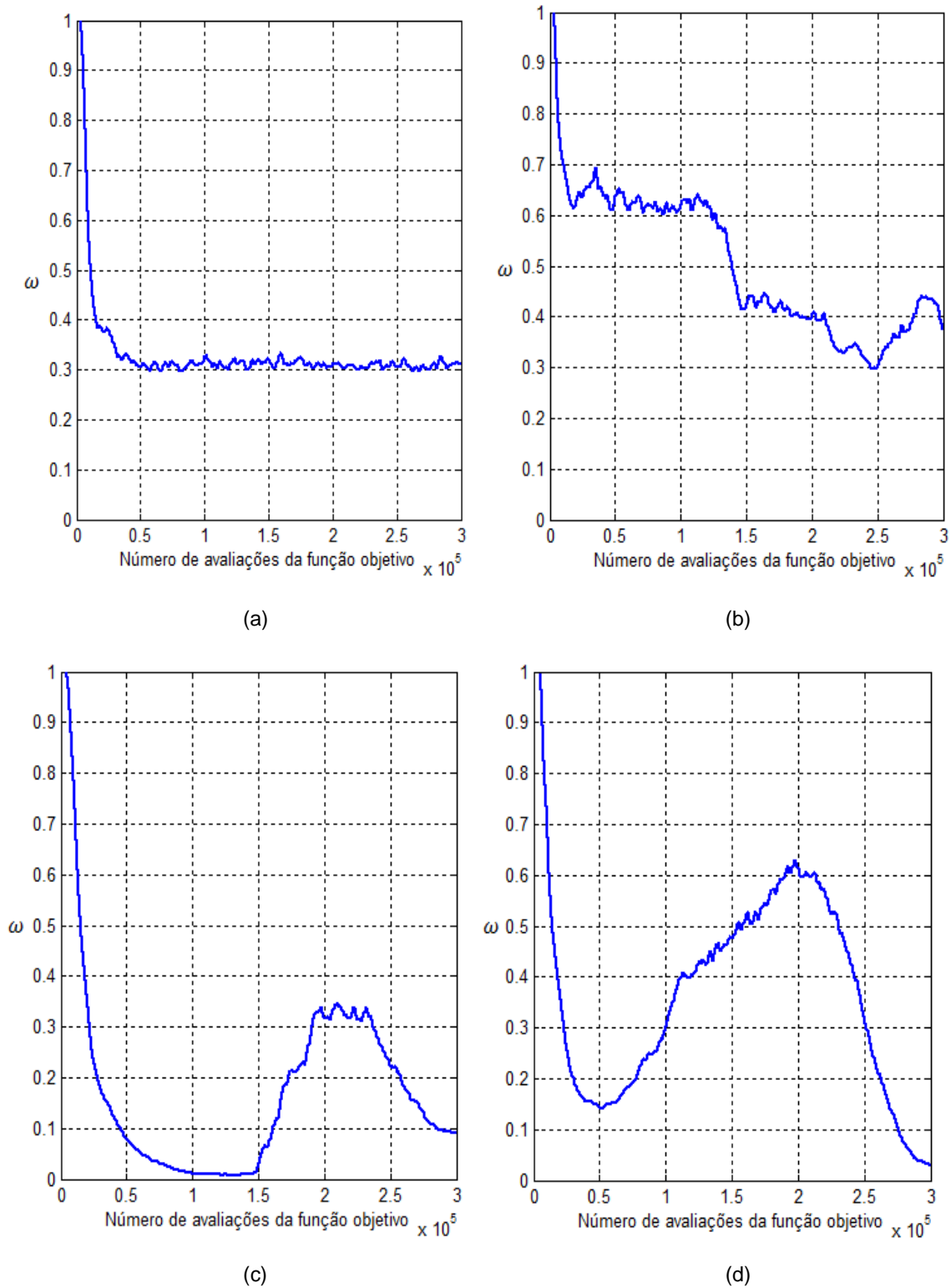


FIGURA 19: Evolução do fator de ajuste  $\omega$  para quatro funções do CEC2005 (a)  $F_4$ , (b)  $F_6$ , (c)  $F_{14}$ , (d)

$F_{16}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

## 7 CALIBRAÇÃO DE MOTORES AUTOMOTIVOS

Todo motor automotivo na atualidade, seja ele diesel ou gasolina, possui um sistema de controle eletrônico cuja responsabilidade é gerenciar o funcionamento do motor de modo que ele opere de acordo com requisitos pré estabelecidos de desempenho, emissões, consumo, dirigibilidade e durabilidade.

Este sistema de controle do motor é formado por sensores, atuadores e por uma *Engine Control Unit* (ECU). A ECU nada mais é do que um conjunto formado por um *hardware* e *software* cuja finalidade é controlar os diversos atuadores do motor a fim de garantir o seu correto funcionamento em diferentes pontos de operação. Para que isto seja possível, a ECU utiliza informações provenientes de sensores instalados no motor que indicam a condição de funcionamento do mesmo em um dado momento. De uma forma simplificada, pode-se dizer que os sinais dos sensores são lidos pela ECU e interpretados através de funções implementadas em seu *software* que por sua vez determinam como devem ser acionados os atuadores.

A FIGURA 20 ilustra os diversos atuadores e sensores que estão presentes em um sistema de controle de motor a gasolina aspirado com injeção indireta.

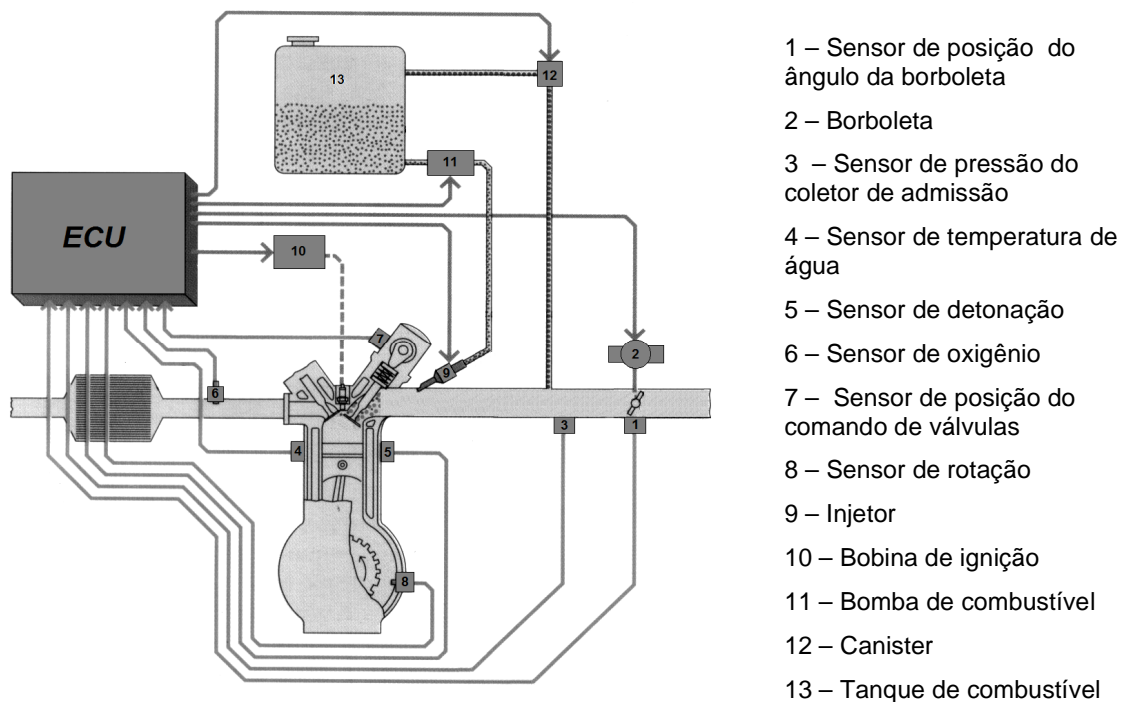


FIGURA 20: Exemplo de um sistema de controle de motor a gasolina aspirado com injeção indireta

FONTE: O autor (2014)

O *software* contido na ECU é formado por um elevado número de funções. Estas funções possuem parâmetros que necessitam ser ajustados de tal forma que se obtenha a resposta desejada do motor, ou seja, o ajuste dos parâmetros do *software* da ECU é realizado para que o motor opere de acordo com requisitos de desempenho, emissões, consumo, dirigibilidade em qualquer ponto de operação. A este processo de ajuste dos parâmetros é dado o nome de calibração.

Em métodos convencionais, a calibração de cada um dos parâmetros das diversas funções do *software* da ECU é realizada manualmente através de testes realizados em dinamômetro de motores, dinamômetro de rolos e em veículo, o que normalmente exige um elevado grau de esforço e tempo.

Entretanto, conforme já mencionado no capítulo 1, houve ao longo dos últimos anos um aumento na complexidade dos *softwares* do sistema de controle do motor, o que gerou, conseqüentemente, um aumento significativo no número de parâmetros a serem ajustados.

Aliado a isto, ainda existe o fato de haver uma crescente demanda por redução no tempo de desenvolvimento dos projetos por parte do gerenciamento de projetos das empresas o que faz com que haja uma redução no período disponível para a realização do processo de calibração.

Neste cenário no qual existe um elevado número de parâmetros a serem calibrados ao mesmo tempo em que o período disponível para a realização da calibração é reduzido, cada vez mais se torna necessário a substituição do ajuste manual dos parâmetros das funções do *software* da ECU por ferramentas computacionais que possam realizar a calibração automática destes parâmetros para que ao final possa-se obter uma calibração na qual todos os requisitos do projetos são atendidos.

Nesse sentido, o objetivo desta seção é de demonstrar a factibilidade e a eficiência da utilização do algoritmo desenvolvido neste trabalho *HSADE* na realização de calibração automática de determinadas funções de *software* da ECU.

## 7.1 APLICAÇÃO DO ALGORITMO HSADE NA REALIZAÇÃO DA CALIBRAÇÃO AUTOMÁTICA DE MOTORES AUTOMOTIVOS

Para verificar a factibilidade e eficácia do uso do algoritmo *HSADE* na realização de calibração automática de funções de *software* de sistema de controle de motor, dois estudos de caso foram selecionados. O primeiro estudo de caso refere-se a calibração automática de uma função de *software* utilizada no gerenciamento de funcionamento de motores diesel e o segundo estudo de caso refere-se a calibração automática de uma função de *software* utilizada no gerenciamento de funcionamento de motores a gasolina. Ambos os estudos de caso estão relacionados a tarefas comuns de calibração realizadas pelos engenheiros durante o processo de calibração do motor.

Os estudos de caso escolhidos referem-se a calibração de duas funções que representam modelos de cálculo de variáveis físicas. A motivação para a escolha destes estudos de caso está relacionada ao fato de que *softwares* de sistema de controle de motor possuem um elevado número de modelos de cálculo de variáveis físicas implementados e, conseqüentemente, esta área tem um grande potencial para a aplicação de calibração automática.

Modelos de cálculo de variáveis físicas são comumente utilizados para modelar grandezas físicas necessárias para o correto gerenciamento do funcionamento no motor. Estes modelos são utilizados, por exemplo, para substituir um sinal de um sensor que não está mais disponível (devido a uma falha elétrica, por exemplo) ou simplesmente para fornecer o sinal de uma grandeza modelada devido ao fato de não haver um sensor instalado no motor para fornecer o sinal diretamente.

Sendo assim, o primeiro estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão de combustível no *rail* em motores diesel com sistema *common rail*. O segundo estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão do ar no coletor de admissão em motores a gasolina.

Cada um dos estudos de caso, bem como os resultados obtidos, serão descritos em detalhes nas seções subsequentes.

### 7.1.1 Metodologia

A metodologia utilizada para a realização da calibração automática dos dois estudos de casos é apresentada na FIGURA 21. Como pode ser observado, o processo de calibração automática foi dividido em três fases: aquisição de dados, calibração e validação.

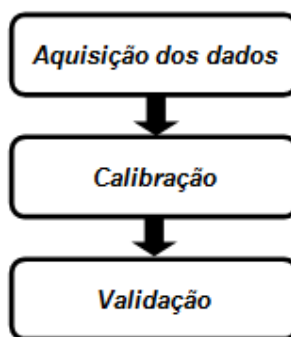


FIGURA 21: Metodologia para a realização da calibração automática

FONTE: O autor (2014)

A primeira fase consiste na aquisição de dois conjuntos de dados para cada estudo de caso: um conjunto que será utilizado para a realização da calibração automática e um segundo conjunto que será utilizado para a validação. Esta fase é extremamente importante, pois a precisão do processo de calibração depende diretamente dos dados utilizados durante o processo de otimização.

Cada conjunto de dados será formado pelos dados de entrada necessários para o modelo e o valor verdadeiro da variável estimada. Todas as variáveis de um mesmo conjunto de dados terão o mesmo número de amostra, cujo valor dependerá da taxa de amostragem e da duração da aquisição dos dados.

A segunda fase consiste na realização efetiva da calibração automática das funções escolhidas como estudo de caso, utilizando-se para isto os algoritmos apresentados nesta dissertação.

O objetivo da calibração automática é encontrar de forma autônoma a melhor configuração dos parâmetros do modelo, a fim de obter uma estimativa confiável, isto é, o erro do modelo é o mais baixo possível. Sendo assim, o processo de calibração automática será equivalente a um processo de minimização do erro de

estimação através de uma seleção adequada dos parâmetros do modelo. Além disto, visto que estes modelos estão relacionados à estimação de grandezas físicas, os parâmetros a serem ajustados estão sujeitos a limites inferiores e superiores.

Conseqüentemente, a calibração automática a ser aplicada nos estudos de caso pode ser tratada como um problema de otimização mono-objetivo sujeito apenas a restrição lateral na qual a função objetivo  $f(\cdot)$  a ser minimizada corresponde ao erro de estimação do modelo. A dimensão  $D$  do problema de otimização corresponderá ao número de parâmetros do modelo a serem ajustados.

A FIGURA 22 apresenta de forma esquemática o processo de calibração automática dos parâmetros do modelo. Nesta figura,  $X_1, X_2, \dots, X_k$  representam as entradas do modelo,  $\hat{Y}$  corresponde à variável estimada pelo modelo e  $Y$  o valor real da variável estimada.

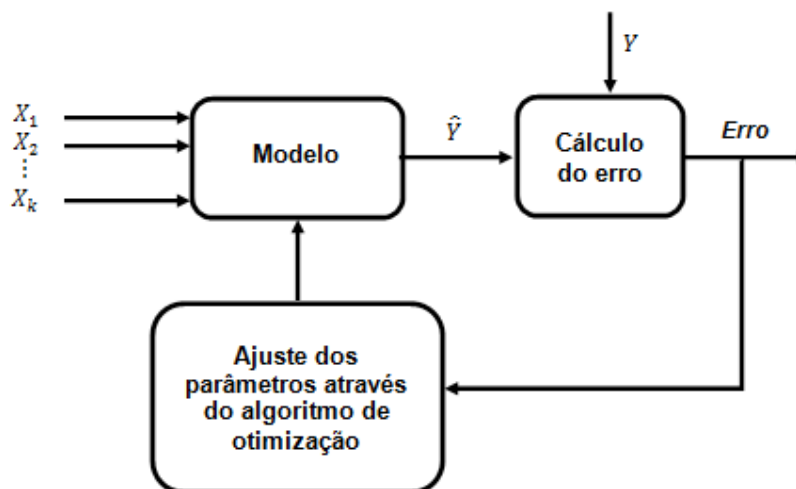


FIGURA 22: Quadro esquemático do processo de calibração automática

FONTE: O autor (2014)

Neste processo, os parâmetros do modelo serão ajustados de forma iterativa pelo algoritmo de otimização de tal forma que o erro de estimação (diferença entre o valor real e valor estimado) seja minimizado. O processo de otimização será executado até que um critério de parada seja atingido. Para este trabalho, será utilizado como critério de parada quando se atinge um número máximo de avaliações da função objetivo.

O erro de estimação do modelo pode ser calculado de diversas formas. Entretanto, neste trabalho, foi escolhido o Erro Médio Quadrático (EMQ) para avaliar a diferença entre o valor estimado e o valor verdadeiro.

$$EMQ = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (71)$$

onde  $n$  é número de amostras.

Sendo assim, a função objetivo a ser minimizada durante o processo de otimização será o EMQ.

Ao final do processo de otimização, uma calibração para o modelo será gerada. Esta calibração será dada por  $\vec{X}_{Melhor}$ , onde  $\vec{X}_{Melhor}$  representa a melhor solução encontrada pelo algoritmo de otimização ao final do processo de otimização, ou seja, o indivíduo da população que fornece o menor valor do EMQ ao final do processo de otimização.

A última fase consiste em utilizar um conjunto de dados que não foi utilizado durante o processo de calibração com o objetivo de qualificar o desempenho do resultado do processo de calibração.

### 7.1.2 Estudo de caso: calibração do modelo de cálculo da pressão de combustível no *rail*

O primeiro estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão de combustível no *rail* em motores diesel com sistema *common rail*.

O modelo de cálculo da pressão de combustível no *rail* utilizado neste estudo de caso é baseado na chamada tabela *lookup*.

Uma tabela *lookup* possui  $n \times m$  pontos e é representada por  $l = n \times m$  triplas  $(x^i, y^j, z^{(i,j)})$ , onde  $z^{(i,j)}$  é o valor da tabela *lookup* na posição  $(x^i, y^j)$ , para  $i = 1, 2, \dots, n$  e  $j = 1, 2, \dots, m$ . Conseqüentemente, o valor desejado  $\tilde{z}$  correspondente ao ponto de operação  $(\tilde{x}, \tilde{y})$  é calculado de acordo com uma interpolação linear da seguinte forma:

$$\begin{aligned} \tilde{x} = & (\tilde{y} - y^j) \cdot \left( (\tilde{x} - x^i) \cdot z^{(i,j)} + (x^{i+1} - \tilde{x}) \cdot z^{(i+1,j)} \right) \\ & + (y^{j+1} - \tilde{y}) \cdot \left( (\tilde{x} - x^i) \cdot z^{(i,j+1)} + (x^{i+1} - \tilde{x}) \cdot z^{(i+1,j+1)} \right) \end{aligned} \quad (72)$$



onde  $i$  e  $j$  são definidos tal que as seguintes desigualdes são satisfeitas:  $x^i \leq \tilde{x} < x^{i+1}$  e  $y^j \leq \tilde{y} < y^{j+1}$ .

A pressão de combustível no *rail* é calculada a partir de uma tabela *lookup* função da rotação do motor e da quantidade de combustível injetado. Sendo assim, a otimização do modelo de pressão de combustível no *rail* corresponde a determinação de valores adequados para cada um dos pontos da tabela *lookup*.

A tabela *lookup* do modelo de pressão de combustível no *rail* possui 300 pontos. Conseqüentemente, a dimensão  $D$  deste estudo de caso será  $D = 300$  (300 parâmetros a serem ajustados).

É importante salientar que uma tabela *lookup* é uma das formas mais simples de se modelar uma variável física em um sistema de controle motor. Conseqüentemente, o modelo de pressão de combustível no *rail* utilizado neste estudo de caso é considerado simples, embora possua um número elevado de parâmetros a serem ajustados.

As restrições laterais impostas aos parâmetros do modelo a serem ajustados foram definidas a fim de ter o espaço de busca alocado em uma área física factível.

As duas versões do algoritmo HSADE serão utilizados para a realização da calibração automática. O detalhamento dos resultados obtidos por cada versão do algoritmo será apresentado nas próximas seções.

Os valores de parâmetros de cada versão do algoritmo HSADE utilizados para este estudo de caso, são os mesmos definidos na seção 6.3.1, exceto pelo tamanho da população. Para as duas versões HSADE-v1 e HSADE-v2, o tamanho da população foi escolhido como  $NP = 120$ .

O processo de calibração automática será finalizado após um número máximo de 120000 avaliações da função objetivo.

Como já mencionado anteriormente, a função objetivo a ser minimizada será o EMQ. O cálculo do EMQ será realizado baseando-se na diferença entre o valor estimado pelo modelo da pressão de combustível no *rail* e o valor de pressão real medido por um sensor localizado no *rail*.

### 7.1.2.1 Aquisição dos dados

Conforme mencionado na seção 7.1.1, dois conjuntos de dados experimentais foram gerados: um conjunto que será utilizado para a realização da calibração automática e um segundo conjunto que será utilizado para a validação.

Os dados experimentais foram obtidos a partir de medições realizadas diretamente em um veículo utilitário equipado com um motor diesel *common rail*.

Vale salientar que a robustez da calibração do modelo de cálculo da pressão do *rail* depende diretamente dos dados utilizados no processo de otimização. Para que a calibração gerada pelo processo de otimização seja robusta e confiável, é necessário que os dados de otimização sejam gerados em diversas condições de funcionamento do motor de tal forma que toda a faixa de funcionamento do motor seja coberta.

A fim de que os dados obtidos fossem gerados em diversas condições de funcionamento do motor, as medições foram realizadas em condições de condução variadas em trechos urbanos e rodoviários. Além disso, de forma a obter uma validação mais robusta, cada conjunto de dados (otimização e validação) foi gerado em diferentes condições de condução.

Os dados para otimização correspondem a uma aquisição realizada durante uma rodagem com o veículo de aproximadamente 44 minutos e os dados de validação correspondem a uma aquisição realizada durante uma rodagem com o veículo de aproximadamente 25 minutos. Além disso, os dados foram adquiridos com uma taxa de 10 amostras por segundo.

Para cada conjunto de dados, os seguintes parâmetros foram gravados:

- rotação do motor;
- quantidade combustível injetado;
- valor do sensor de pressão de *rail*.

### 7.1.2.2 Resultados experimentais e discussão

Os dois algoritmos, HSADE-v1 e HSADE-v2, foram utilizados para gerar uma calibração do modelo de cálculo da pressão do *rail*. Nas seções subsequentes, uma análise mais detalhada dos resultados experimentais obtidos será apresentada para cada uma das duas versões do algoritmo HSADE.

#### 7.1.2.2.1 Análise do algoritmo HSADE-v1

A calibração dos parâmetros do modelo de cálculo da pressão do combustível no *rail* foi gerada através da aplicação do algoritmo HSADE-v1 sobre o conjunto de dados de otimização.

O resultado final do processo de otimização é ilustrado na FIGURA 23. Nesta figura é apresentada a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo utilizando-se a calibração gerada pelo algoritmo HSADE-v1. Esta comparação é apresentada para todo o conjunto de otimização.

Para uma melhor visualização do resultado do processo de otimização, as FIGURAS 24, 25, 26 e 27 ilustram em maiores detalhes a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo para alguns trechos do conjunto de otimização.

Como pode ser observado nas FIGURAS 23 a 27, o resultado final obtido ao final do processo de otimização foi excelente. Nota-se que o valor calculado pelo modelo é bem próximo do valor real. Além disso, vale salientar que o modelo apresentou uma boa estimativa tanto para em condições de transitórios (FIGURA 24 FIGURA 67) quanto para condições mais estabilizadas (FIGURA 26).

Nota-se, entretanto, que o valor estimado não é exatamente igual ao valor real, ou seja, o erro de estimativa não foi nulo (o valor do EMQ obtido ao final do processo de otimização foi de 97,9).

Como pode ser observado nas FIGURAS 24 a 27, existem algumas diferenças entre o valor real e o valor estimado da pressão do *rail*. Entretanto é necessário levar em conta a precisão do modelo. Conforme mencionado anteriormente, o modelo de cálculo da pressão do *rail* é um modelo bem simples e ele não é capaz de determinar com exatidão a pressão em todas as condições de funcionamento do motor (condições de transitórios e estabilizado). Isto faz com que existam erros inerentes ao próprio modelo, independentemente de sua calibração. Deste modo, um erro devido à imprecisão do modelo estará sempre presente.

Sendo assim, tendo em vista a precisão do modelo, os resultados apresentados demonstram que a calibração gerada para os dados de otimização foi corretamente calculada.

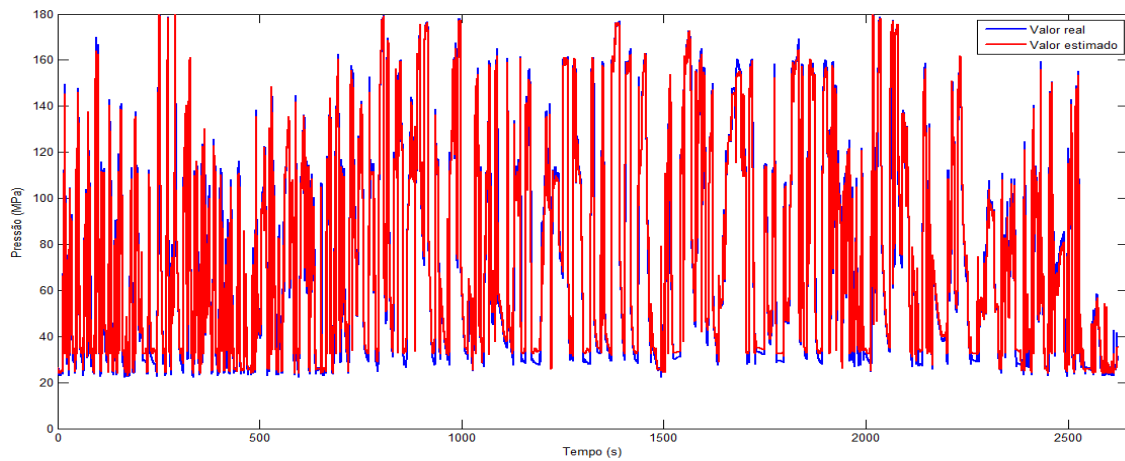


FIGURA 23: Otimização: comparação entre o valor real e o valor estimado para a pressão do rail

FONTE: O autor (2014)

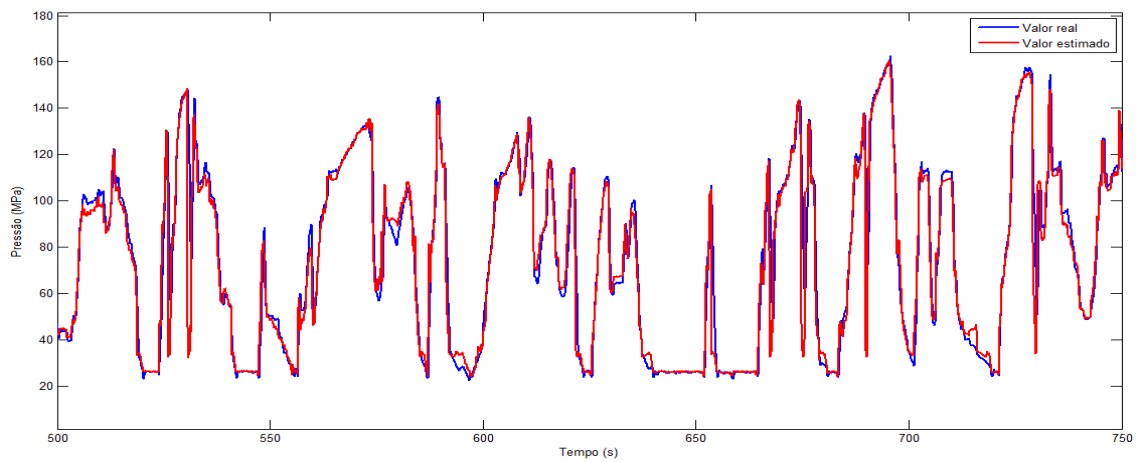


FIGURA 24: Detalhe do resultado processo de otimização entre 500 e 750 segundos

FONTE: O autor (2014)

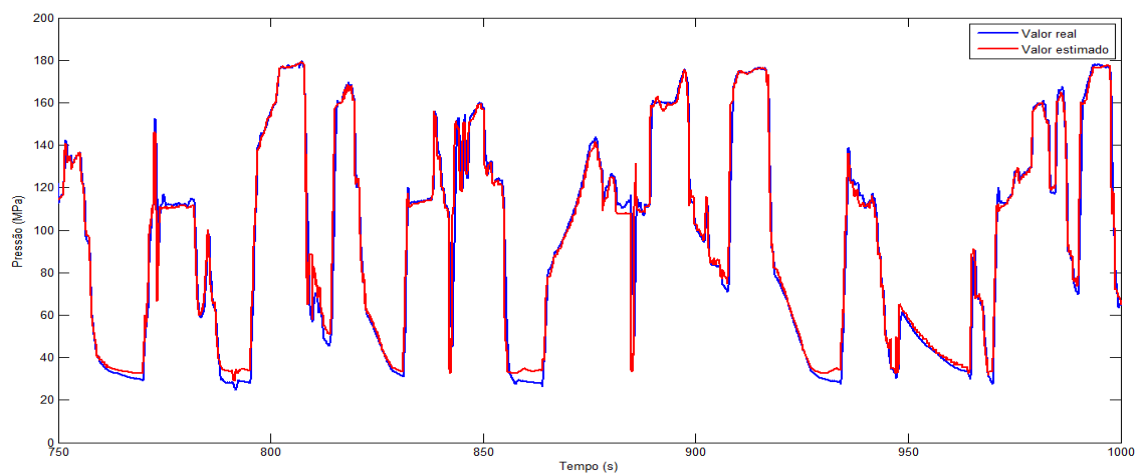


FIGURA 25: Detalhe do resultado processo de otimização entre 750 e 1000 segundos

FONTE: O autor (2014)

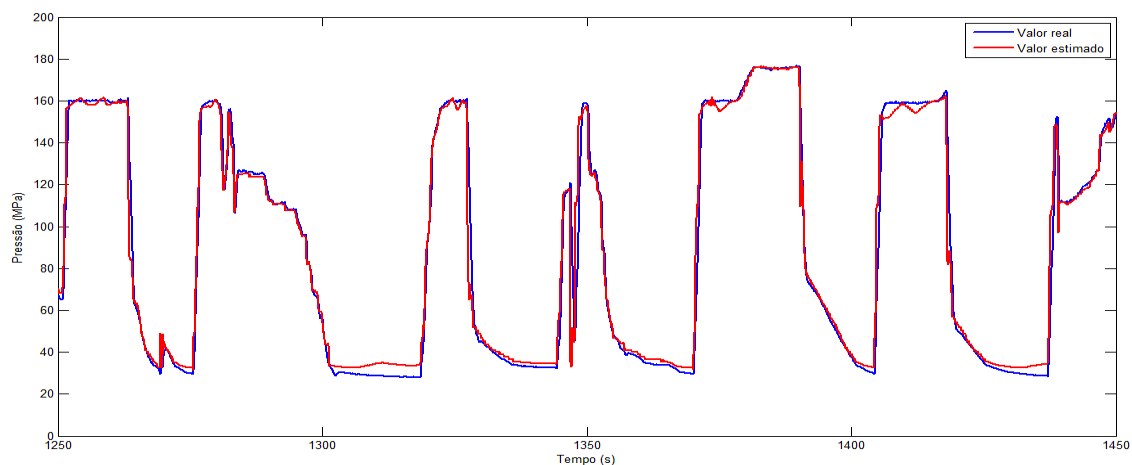


FIGURA 26: Detalhe do resultado processo de otimização entre 1250 e 1450 segundos

FONTE: O autor (2014)

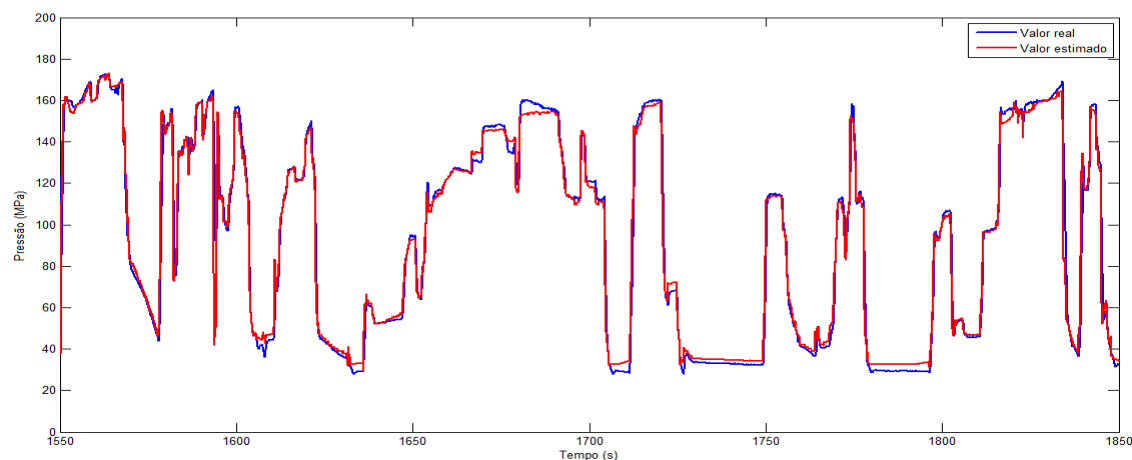


FIGURA 27: Detalhe do resultado processo de otimização entre 1550 e 1850 segundos

FONTE: O autor (2014)

Uma vez gerada a calibração, um segundo conjunto de dados, que não foi empregado no processo de otimização denominado conjunto de validação, foi utilizado para qualificar a calibração do modelo de cálculo da pressão do *rail*.

A FIGURA 28 ilustra a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo a partir dos dados do conjunto de validação, utilizando-se a calibração gerada pelo algoritmo HSADE-v1.

Para uma melhor visualização do resultado da validação, as FIGURAS 29, 30 e 31 ilustram em maiores detalhes a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo

utilizando-se a calibração gerada pelo algoritmo HSADE-v1 para alguns trechos do conjunto de validação.

Como pode ser observado nas FIGURAS 28 a 31, o resultado obtido a partir dos dados de validação foi muito bom. Verifica-se que o valor calculado pelo modelo é bem próximo do valor real, validando, desta forma, a calibração gerada pelo algoritmo HSADE-v1.

Assim como havia sido observado com os dados de otimização, o erro de estimação não foi nulo (o valor do EMQ obtido para o conjunto de dados de validação foi de 86,6). Mas, como já foi discutido, existe um erro intrínseco ao modelo, independente da calibração, o qual não é possível de ser eliminado completamente.

Em suma, os resultados apresentados nesta seção demonstram que o algoritmo HSADE-v1 foi capaz gerar uma calibração adequada e confiável para o modelo de cálculo da pressão do *rail*, comprovando a factibilidade e eficácia do uso do algoritmo HSADE-v1 na realização de calibração automática de funções de *software* de sistema de controle de motor.

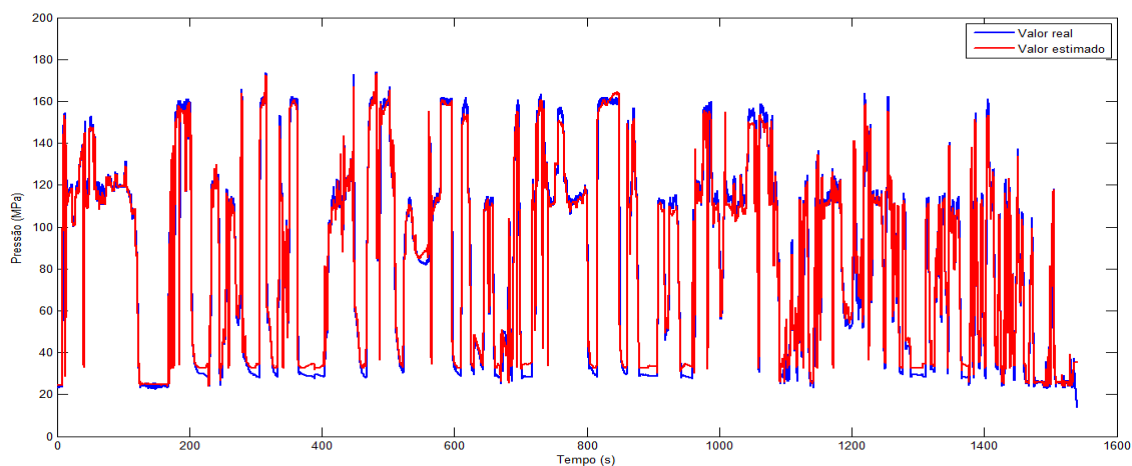


FIGURA 28: Validação: comparação entre o valor real e o valor estimado para a pressão do *rail*

FONTE: O autor (2014)

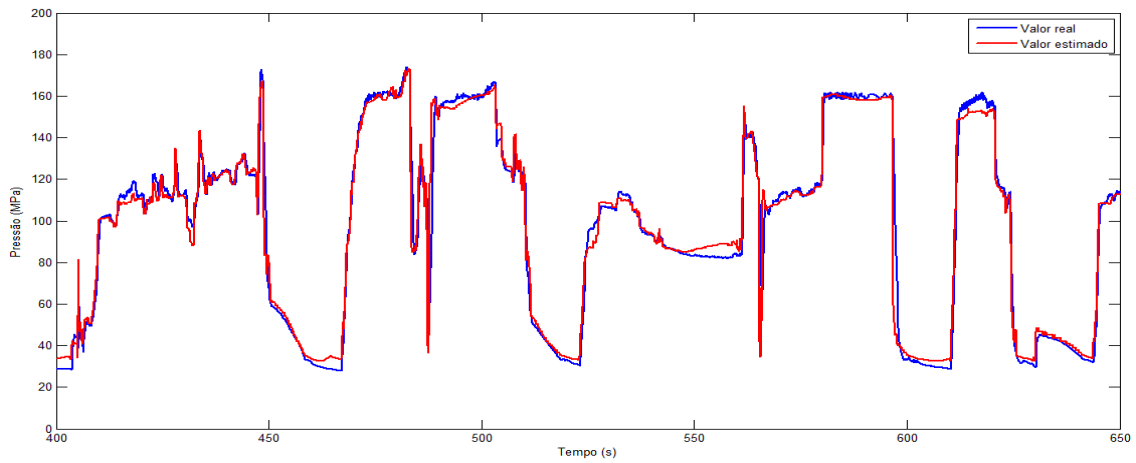


FIGURA 29: Detalhe do resultado da validação entre 400 e 650 segundos

FONTE: O autor (2014)

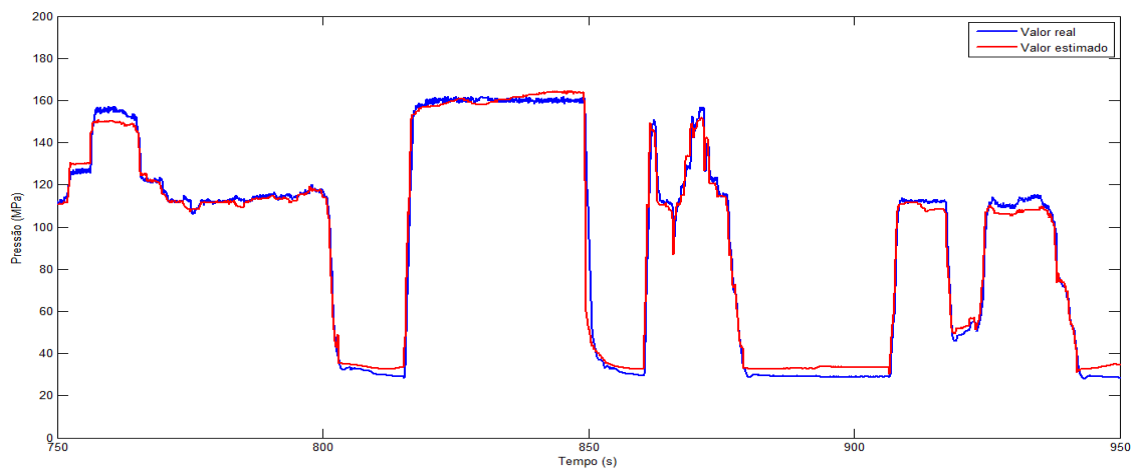


FIGURA 30: Detalhe do resultado da validação entre 750 e 950 segundos

FONTE: O autor (2014)

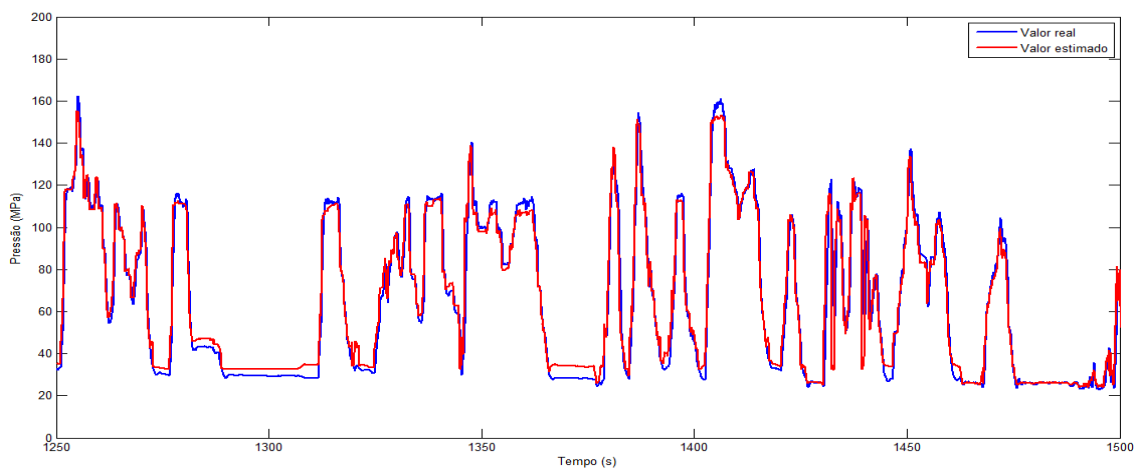


FIGURA 31: Detalhe do resultado da validação entre 1250 e 1500 segundos

FONTE: O autor (2014)

#### 7.1.2.2.2 Análise do algoritmo HSADE-v2

Nesta seção, serão apresentados os resultados experimentais obtidos através da aplicação do algoritmo HSADE-v2 na realização da calibração automática do modelo de cálculo da pressão de *rail*.

A calibração dos parâmetros do modelo de cálculo da pressão de *rail* foi gerada através da aplicação do algoritmo HSADE-v2 sobre o conjunto de dados de otimização.

O resultado final do processo de otimização é ilustrado na FIGURA 32. Nesta figura é apresentada a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo utilizando-se a calibração gerada pelo algoritmo HSADE-v2. Esta comparação é apresentada para todo o conjunto de otimização.

Para uma melhor visualização do resultado do processo de otimização, as FIGURAS 33, 34, 35 e 36 ilustram em maiores detalhes a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo para alguns trechos do conjunto de otimização.

Como pode ser observado nas FIGURAS 32 a 36, o resultado final obtido após o processo de otimização foi muito bom e parecido com o resultado obtido para a primeira versão do algoritmo *HSADE*.

Assim como havia sido observado anteriormente, verifica-se que o valor calculado pelo modelo é bem próximo do valor real. Além disso, observa-se que o modelo apresentou uma boa estimativa tanto para em condições de transitórios (FIGURA 33) quanto para condições mais estabilizadas (FIGURA 35).

Nota-se, também, que o valor estimado não é exatamente igual ao valor real, ou seja, o erro de estimativa não foi nulo (o valor do EMQ obtido ao final do processo de otimização foi de 98,0). Mas, como já discutido na seção 7.1.2.2.1, existe um erro intrínseco ao modelo, independente da calibração, assim não é possível eliminá-lo.

Sendo assim, tendo em vista a precisão do modelo, os resultados experimentais demonstram que a calibração gerada para os dados de otimização foi corretamente calculada.



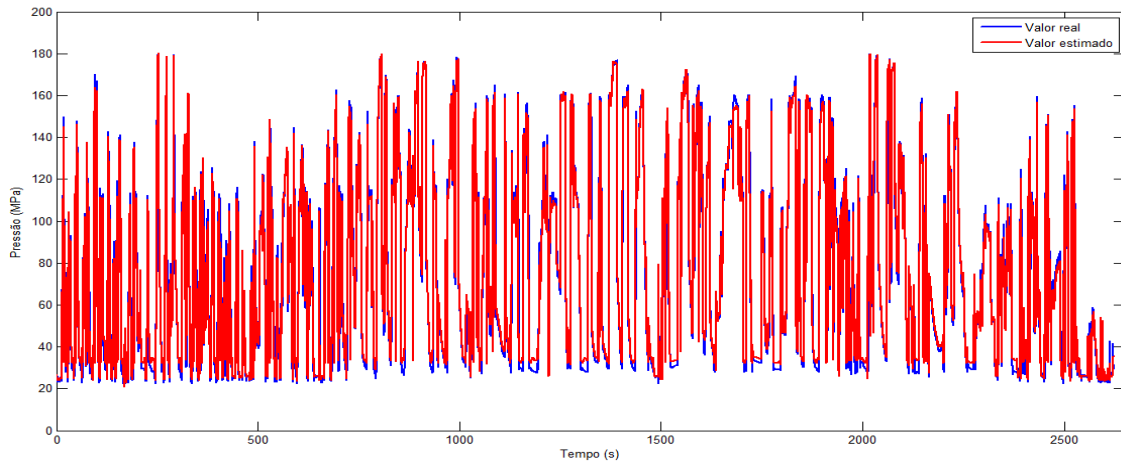


FIGURA 32: Otimização: comparação entre o valor real e o valor estimado para a pressão do rail

FONTE: O autor (2014)

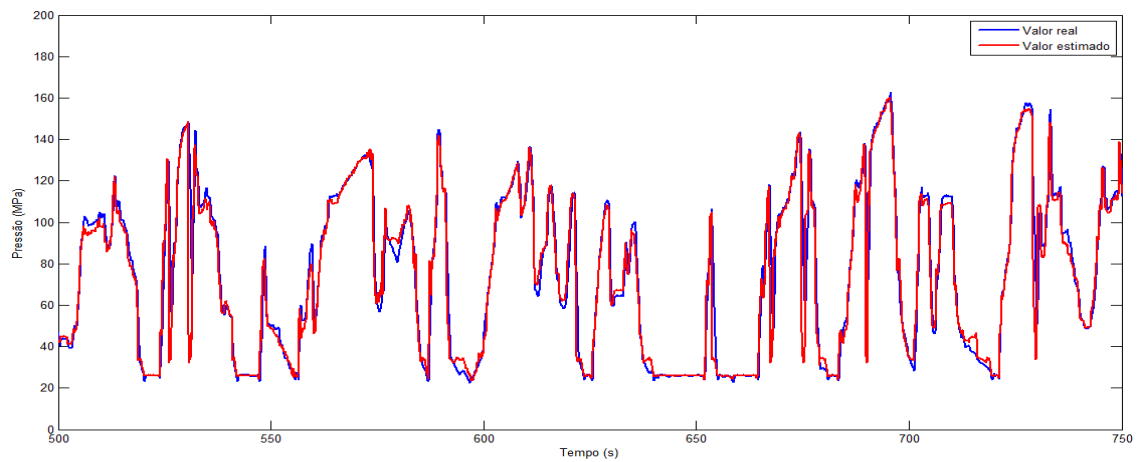


FIGURA 33: Detalhe do resultado processo de otimização entre 500 e 750 segundos

FONTE: O autor (2014)

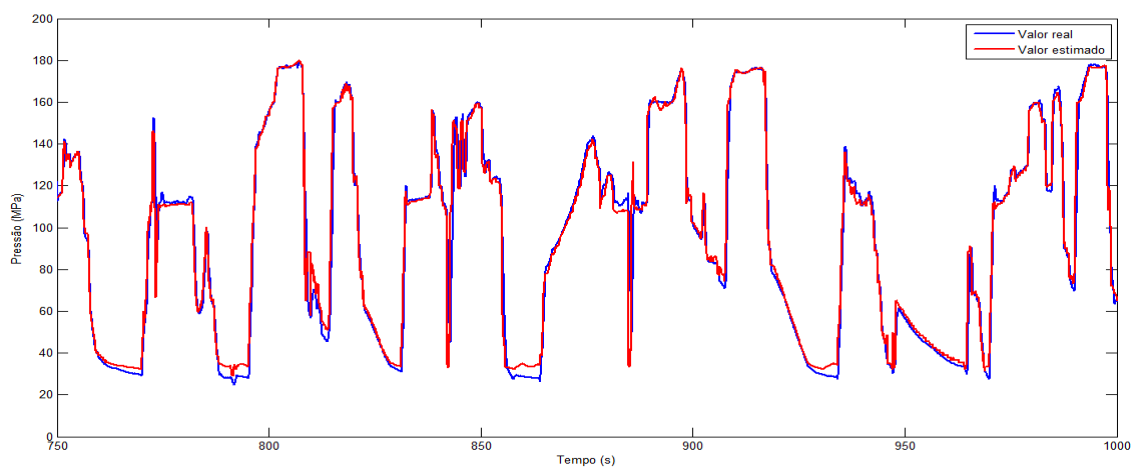


FIGURA 34: Detalhe do resultado processo de otimização entre 750 e 1000 segundos

FONTE: O autor (2014)

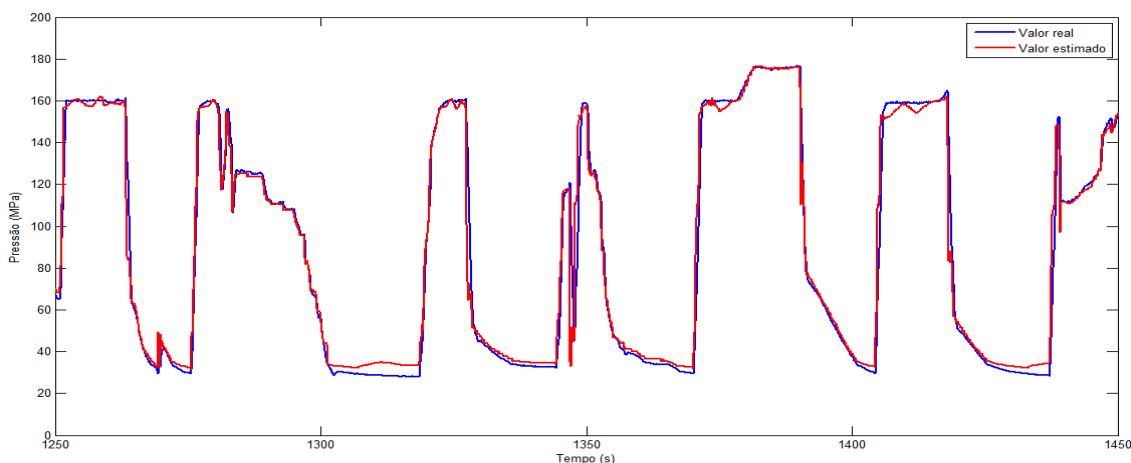


FIGURA 35: Detalhe do resultado processo de otimização entre 1250 e 1450 segundos

FONTE: O autor (2014)

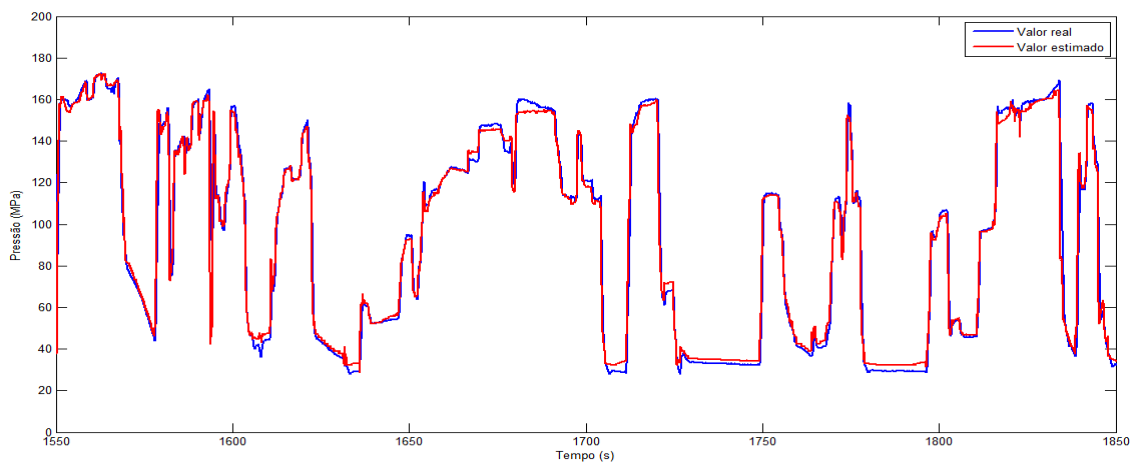


FIGURA 36: Detalhe do resultado processo de otimização entre 1550 e 1850 segundos

FONTE: O autor (2014)

Uma vez gerada a calibração, o conjunto de dados de validação foi utilizado para qualificar a calibração do modelo de cálculo da pressão do *rail*.

A FIGURA 37 apresenta a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo a partir dos dados do conjunto de validação, utilizando-se a calibração gerada pelo algoritmo HSADE-v2.

Para uma melhor visualização do resultado da validação, as FIGURAS 38, 39 e 40 ilustram em maiores detalhes a comparação entre o valor real da pressão do *rail* medida pelo sensor e o valor da pressão do *rail* calculado através do modelo

utilizando-se a calibração gerada pelo algoritmo HSADE-v2 para alguns trechos do conjunto de validação.

Como pode ser observado nas FIGURAS 37 a 40, o resultado obtido a partir dos dados de validação foi muito bom. Verifica-se que o valor calculado pelo modelo é bem próximo do valor real, validando, portanto, a calibração gerada pelo algoritmo HSADE-v2.

Assim como havia sido observado com os dados de otimização, o erro de estimação não foi nulo (o valor do EMQ obtido para o conjunto de dados de validação foi de 86,8). Mas, como já foi discutido, existe um erro intrínseco ao modelo, independente da calibração, assim não é possível eliminá-lo.

Em suma, os resultados apresentados nesta seção demonstram que o algoritmo HSADE-v2 foi capaz gerar uma calibração adequada e confiável para o modelo de cálculo da pressão do *rail*, comprovando a factibilidade e eficácia do uso do algoritmo HSADE-v2 na realização de calibração automática de funções de *software* de sistema de controle de motor.

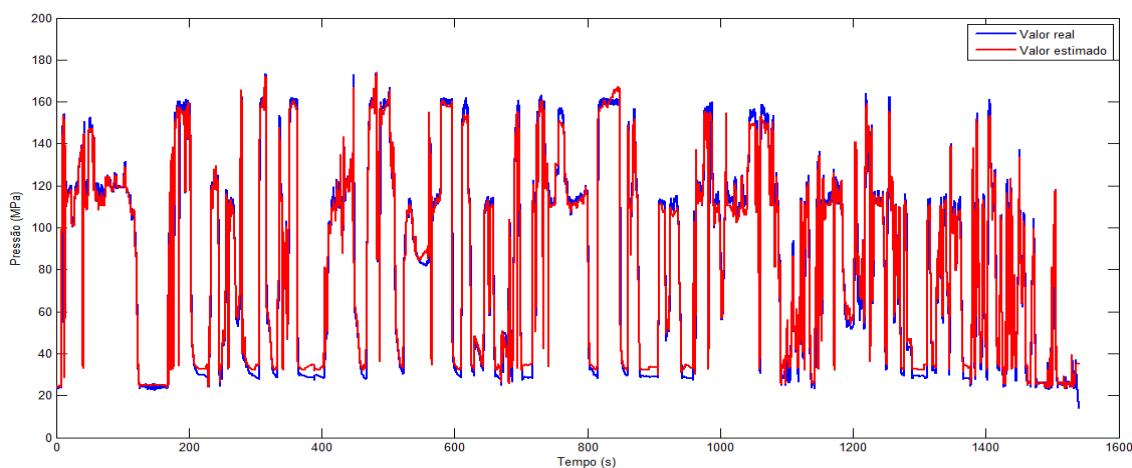


FIGURA 37: Validação: comparação entre o valor real e o valor estimado para a pressão do *rail*

FONTE: O autor (2014)

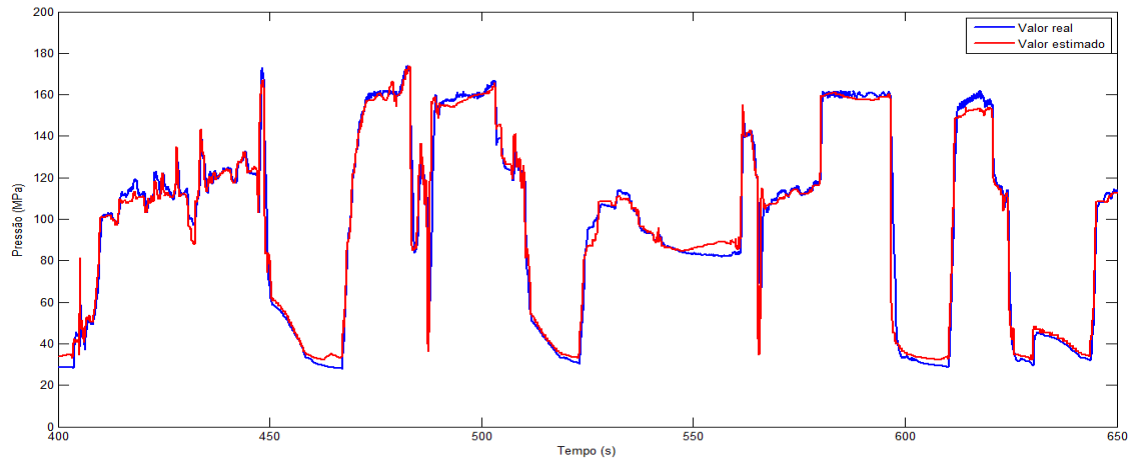


FIGURA 38: Detalhe do resultado da validação entre 400 e 650 segundos

FONTE: O autor (2014)

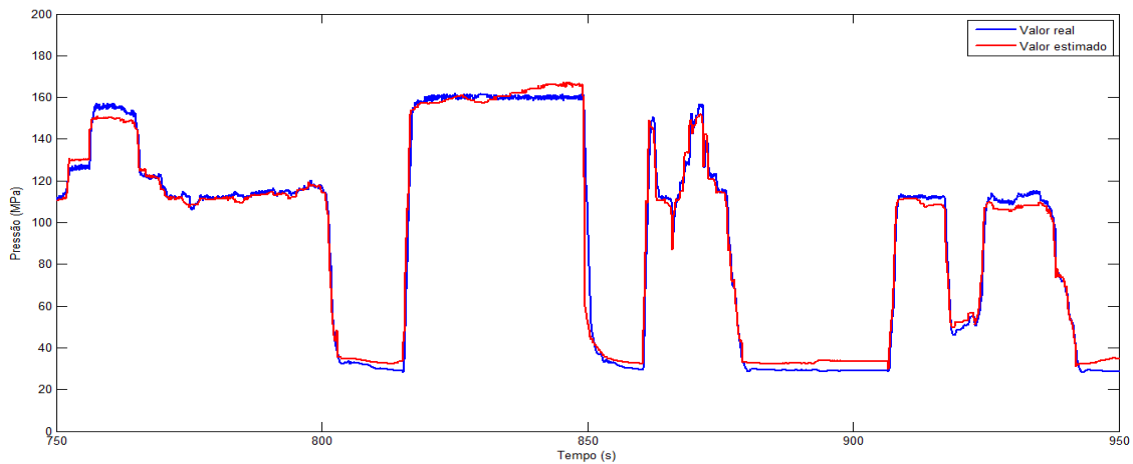


FIGURA 39: Detalhe do resultado da validação entre 750 e 950 segundos

FONTE: O autor (2014)

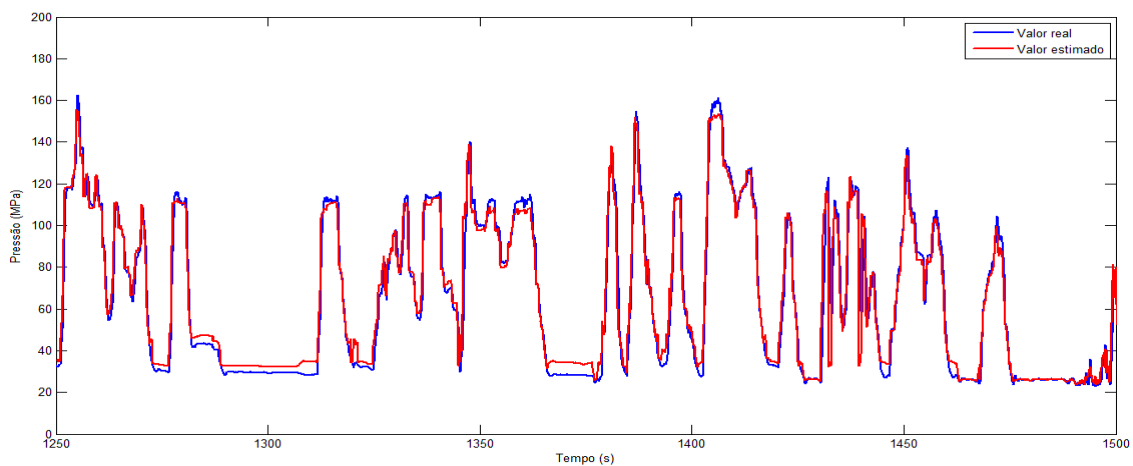


FIGURA 40: Detalhe do resultado da validação entre 1250 e 1500 segundos

FONTE: O autor (2014)

### 7.1.3 Estudo de caso: calibração do modelo de cálculo da pressão do ar no coletor de admissão

O segundo estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão do ar no coletor de admissão em motores a gasolina.

O modelo de cálculo da pressão de ar no coletor de admissão utilizado neste estudo de caso também é baseado em uma tabela *lookup* função da rotação do motor e do ângulo de abertura da borboleta. Sendo assim, a otimização do modelo de da pressão de ar no coletor de admissão corresponde a determinação de valores adequados para cada um dos pontos da tabela *lookup*.

A tabela *lookup* do modelo de pressão de combustível no *rail* possui 400 pontos. Conseqüentemente, a dimensão  $D$  deste estudo de caso será  $D = 400$  (400 parâmetros a serem ajustados).

Assim como no primeiro estudo de caso, vale salientar que este modelo é simples, embora possua um número elevado de parâmetros a serem ajustados.

As restrições laterais impostas aos parâmetros do modelo a serem ajustados foram definidas a fim de ter o espaço de busca alocado em uma área física factível.

As duas versões do algoritmo HSADE serão utilizados para a realização da calibração automática. O detalhamento dos resultados obtidos por cada versão do algoritmo será apresentado nas próximas seções.

Os valores de parâmetros de cada versão do algoritmo HSADE utilizados para este estudo de caso, são os mesmos utilizados para o primeiro estudo de caso e definidos na seção 7.1.2.

Assim como no primeiro estudo de caso, o processo de calibração automática será finalizado após um número máximo de 120000 avaliações da função objetivo. Como já mencionado anteriormente, a função objetivo a ser minimizada será o EMQ. O cálculo do EMQ será realizado baseando-se na diferença entre o valor estimado pelo modelo da pressão do ar no coletor de admissão e o valor de pressão real medido por um sensor localizado no coletor de admissão.

### 7.1.3.1 Aquisição dos dados

Assim como feito no primeiro estudo de caso, dois conjuntos de dados experimentais foram gerados: um conjunto que será utilizado para a realização da calibração automática e um segundo conjunto que será utilizado para a validação.

Os dados experimentais foram obtidos a partir de medições realizadas diretamente em um veículo comercial leve equipado com um motor a gasolina.

Assim como ocorre no primeiro estudo de caso, a robustez da calibração do modelo de cálculo da pressão do coletor de admissão também depende diretamente dos dados utilizados no processo de otimização. Para que a calibração gerada pelo processo de otimização seja robusta e confiável, é necessário que os dados de otimização sejam gerados em diversas condições de funcionamento do motor de tal forma que toda a faixa de funcionamento do motor seja coberta.

A fim de que os dados obtidos fossem gerados em diversas condições de funcionamento do motor, as medições foram realizadas em condições de condução variadas em trechos urbanos e rodoviários. Além disso, de forma a obter uma validação mais robusta, cada conjunto de dados (otimização e validação) foi gerado em diferentes condições de condução.

Os dados para otimização correspondem a uma aquisição realizada durante uma rodagem com o veículo de aproximadamente 18 minutos e os dados de validação correspondem a uma aquisição realizada durante uma rodagem com o veículo de aproximadamente 13 minutos. Além disso, os dados foram adquiridos com uma taxa de 10 amostras por segundo.

Para cada conjunto de dados, os seguintes parâmetros foram gravados:

- rotação do motor;
- ângulo de abertura da borboleta;
- valor do sensor de pressão localizado no coletor de admissão.

### 7.1.3.2 Resultados experimentais e discussão

Os dois algoritmos, HSADE-v1 e HSADE-v2, foram utilizados para gerar uma calibração do modelo de cálculo da pressão do ar no coletor de admissão. Nas seções subsequentes, uma análise mais detalhada dos resultados experimentais obtidos será apresentada para cada uma das duas versões do algoritmo *HSADE*.

### 7.1.3.2.1 Análise do algoritmo HSADE-v1

Nesta seção, serão apresentados os resultados experimentais obtidos através da aplicação do algoritmo HSADE-v1 na realização da calibração automática do modelo de cálculo da pressão do ar no coletor de admissão.

A calibração dos parâmetros do modelo de cálculo da pressão do ar no coletor de admissão foi gerada através da aplicação do algoritmo HSADE-v1 sobre o conjunto de dados de otimização.

O resultado final do processo de otimização é ilustrado na FIGURA 41. Nesta figura é apresentada a comparação entre o valor real da pressão no coletor de admissão medida pelo sensor e o valor da pressão no coletor de admissão calculado através do modelo, utilizando-se a calibração gerada pelo algoritmo HSADE-v1. Esta comparação é apresentada para todo o conjunto de otimização.

Para uma melhor visualização do resultado do processo de otimização, as FIGURAS 42, 43, 44, 45 e 46 ilustram em maiores detalhes a comparação entre o valor real da pressão no coletor de admissão medida pelo sensor e o valor da pressão no coletor de admissão calculado através do modelo para alguns trechos do conjunto de otimização.

Como pode ser observado nas FIGURAS 41 a 46, o resultado final obtido ao final do processo de otimização foi excelente. Nota-se que o valor calculado pelo modelo é bem próximo do valor real.

Nota-se, entretanto, que o valor estimado não é exatamente igual ao valor real, ou seja, o erro de estimação não foi nulo (o valor do EMQ obtido ao final do processo de otimização foi de 1715,5).

Assim como havia sido observado no primeiro estudo de caso, existem algumas diferenças entre o valor real e o valor estimado da pressão no coletor de admissão. Entretanto, neste estudo de caso, também é necessário levar em conta a precisão do modelo. Embora não tenha sido detalhado neste trabalho, o modelo de cálculo da pressão do ar no coletor de admissão é um modelo bem simples, o que faz com que existam erros inerentes ao próprio modelo, independentemente de sua calibração. Deste modo, um erro devido à imprecisão do modelo estará sempre presente.

Sendo assim, tendo em vista a precisão do modelo, os resultados apresentados demonstram que a calibração gerada para os dados de otimização foi corretamente calculada.

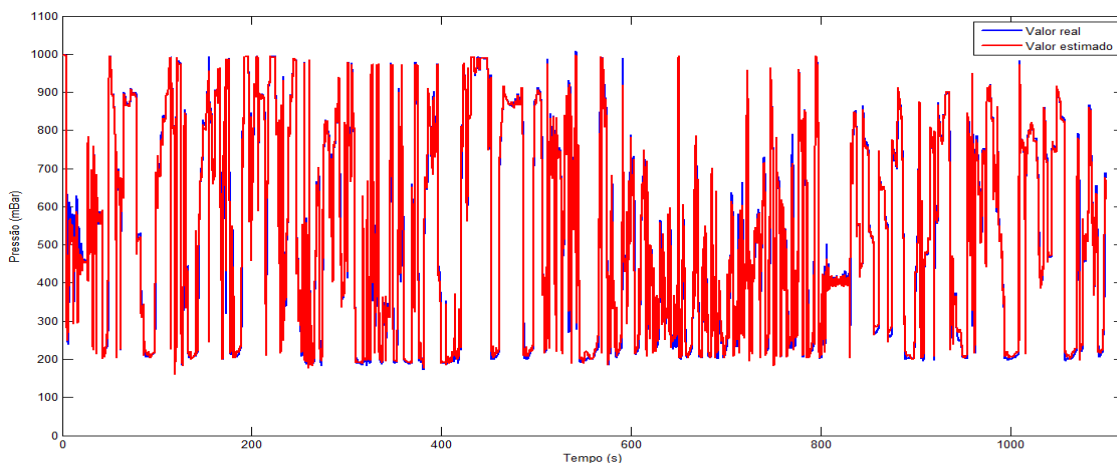


FIGURA 41: Otimização: comparação entre o valor real e o valor estimado para a pressão do ar no coletor de admissão

FONTE: O autor (2014)

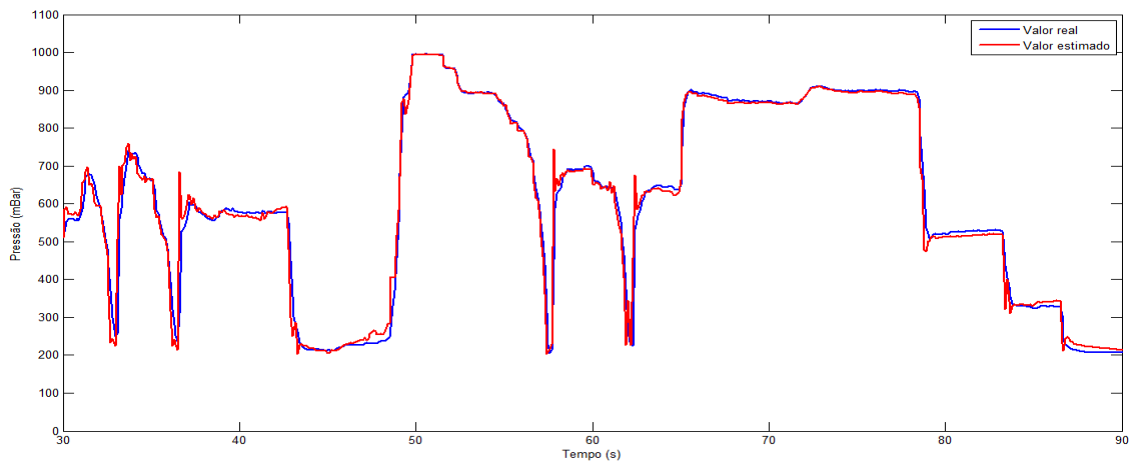


FIGURA 42: Detalhe do resultado processo de otimização entre 30 e 90 segundos

FONTE: O autor (2014)



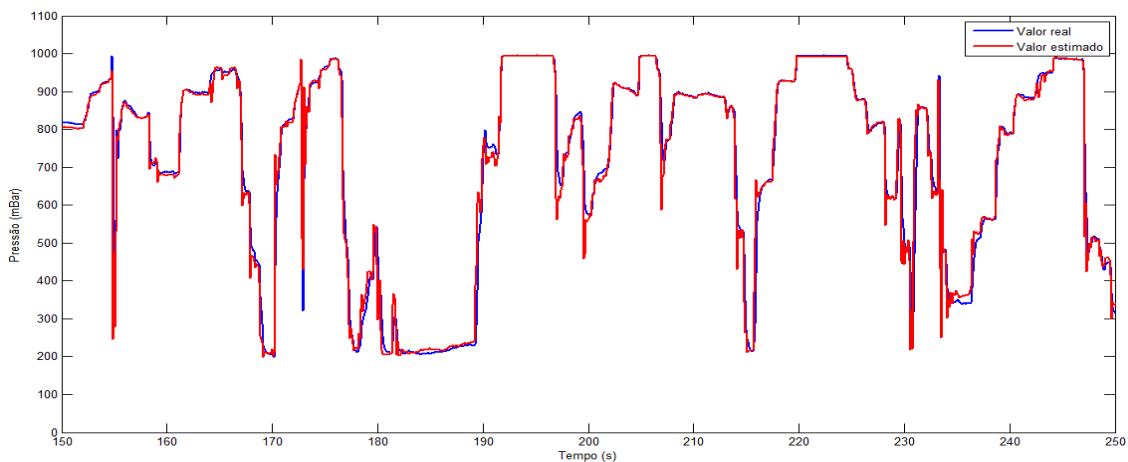


FIGURA 43: Detalhe do resultado processo de otimização entre 150 e 250 segundos

FONTE: O autor (2014)

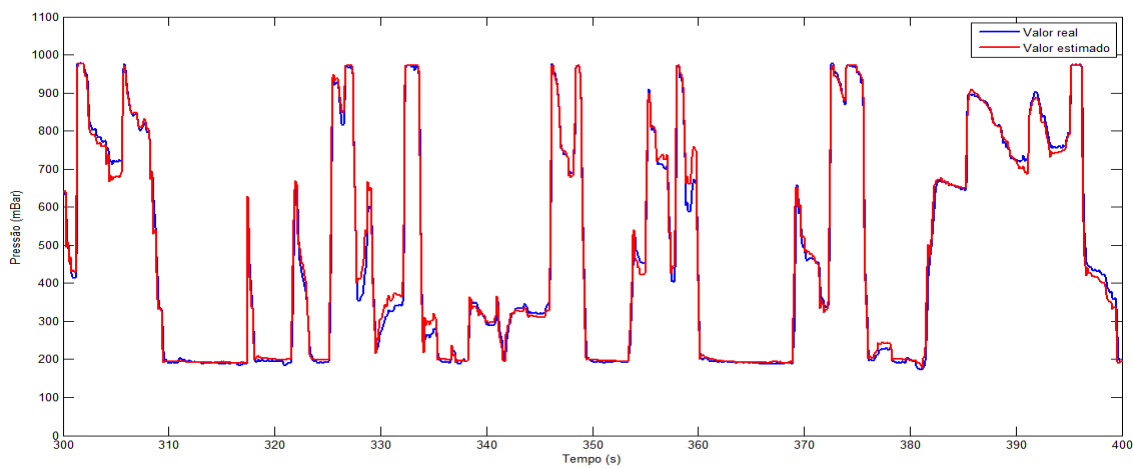


FIGURA 44: Detalhe do resultado processo de otimização entre 300 e 400 segundos

FONTE: O autor (2014)

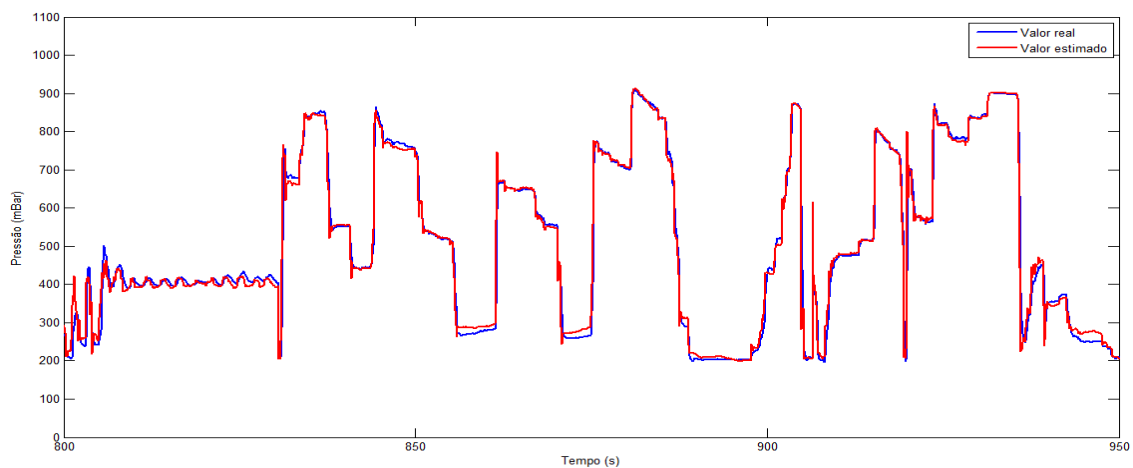


FIGURA 45: Detalhe do resultado processo de otimização entre 800 e 950 segundos

FONTE: O autor (2014)

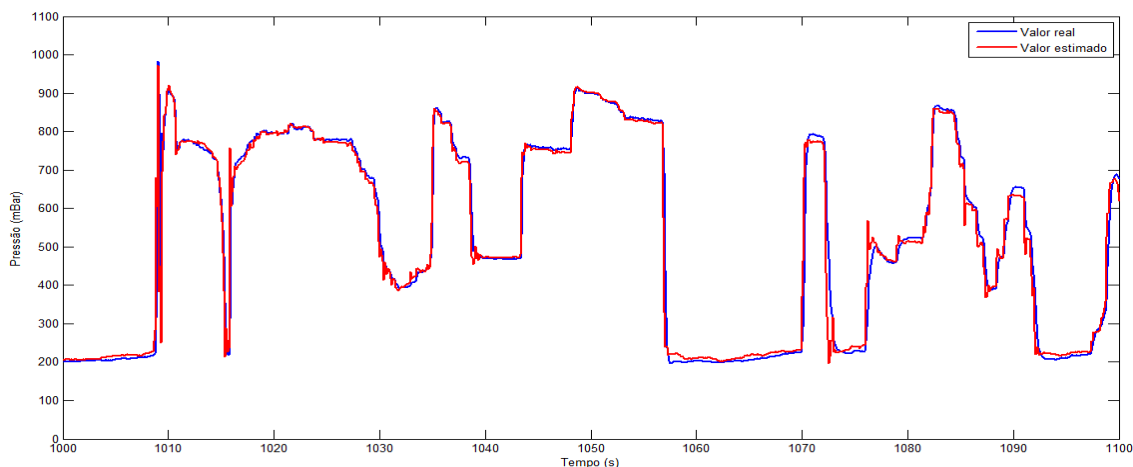


FIGURA 46: Detalhe do resultado processo de otimização entre 1000 e 1100 segundos

FONTE: O autor (2014)

Uma vez gerada a calibração, um segundo conjunto de dados, que não foi empregado no processo de otimização denominado conjunto de validação, foi utilizado para qualificar a calibração do modelo de cálculo da pressão do *rail*.

A FIGURA 47 ilustra a comparação entre o valor real da pressão do coletor de admissão medida pelo sensor e o valor da pressão calculada através do modelo a partir dos dados do conjunto de validação, utilizando-se a calibração gerada pelo algoritmo HSADE-v1.

Para uma melhor visualização do resultado da validação, as FIGURAS 48, 49, 50 e 51 ilustram em maiores detalhes a comparação entre o valor real da pressão do coletor de admissão medida pelo sensor e o valor da pressão do coletor de admissão calculado através do modelo utilizando-se a calibração gerada pelo algoritmo HSADE-v1 para alguns trechos do conjunto de validação.

Como pode ser observado nas FIGURAS 48 a 51, o resultado obtido a partir dos dados de validação foi excelente. Verifica-se que o valor calculado pelo modelo é bem próximo do valor real, validando, desta forma, a calibração gerada pelo algoritmo HSADE-v1.

Assim como havia sido observado com os dados de otimização, o erro de estimativa não foi nulo (o valor do EMQ obtido para o conjunto de dados de validação foi de 2095,5). Mas, como já foi discutido, existe um erro intrínseco ao modelo, independente da calibração, assim não é possível eliminá-lo.

Em suma, os resultados apresentados nesta seção demonstram que o algoritmo HSADE-v1 foi capaz gerar uma calibração adequada e confiável para o

modelo de cálculo da pressão do ar no coletor de admissão, comprovando, desta forma, a factibilidade e eficácia do uso do algoritmo HSADE-v1 na realização de calibração automática de funções de *software* de sistema de controle de motor

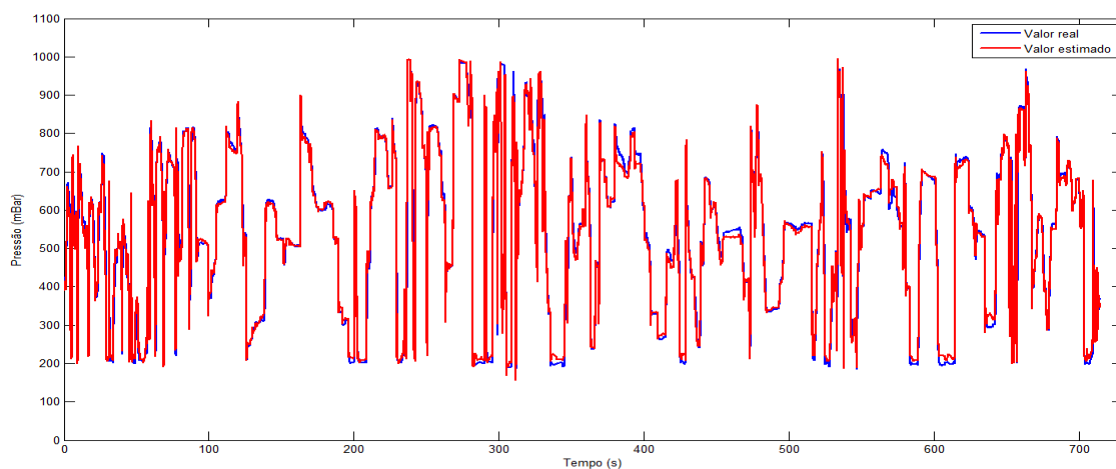


FIGURA 47: Validação: comparação entre o valor real e o valor estimado para a pressão do ar no coletor de admissão

FONTE: O autor (2014)

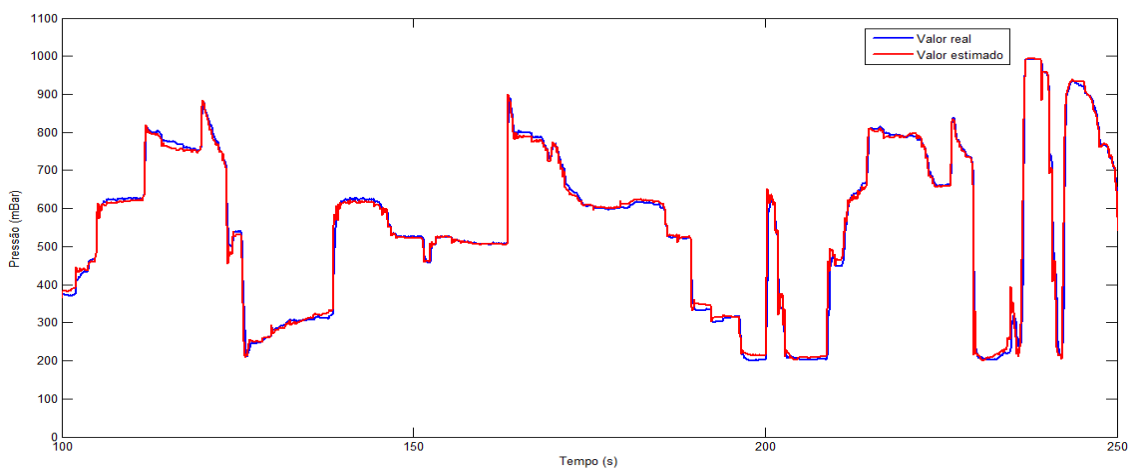


FIGURA 48: Detalhe do resultado da validação entre 100 e 250 segundos

FONTE: O autor (2014)

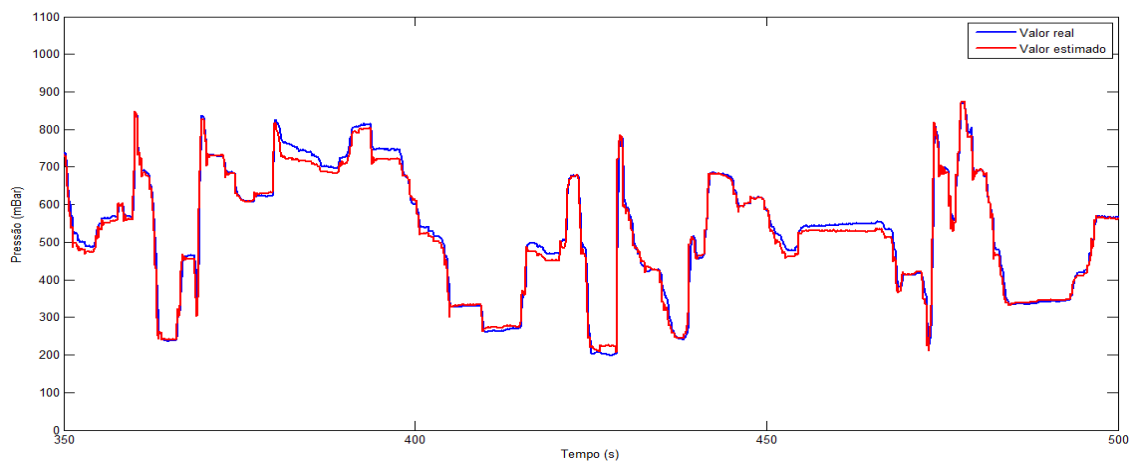


FIGURA 49: Detalhe do resultado da validação entre 350 e 500 segundos

FONTE: O autor (2014)

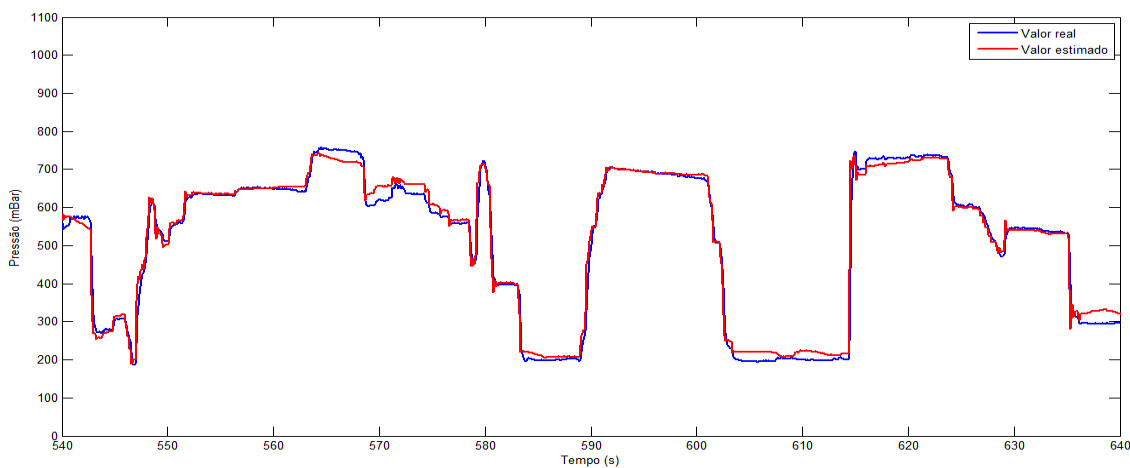


FIGURA 50: Detalhe do resultado da validação entre 540 e 640 segundos

FONTE: O autor (2014)

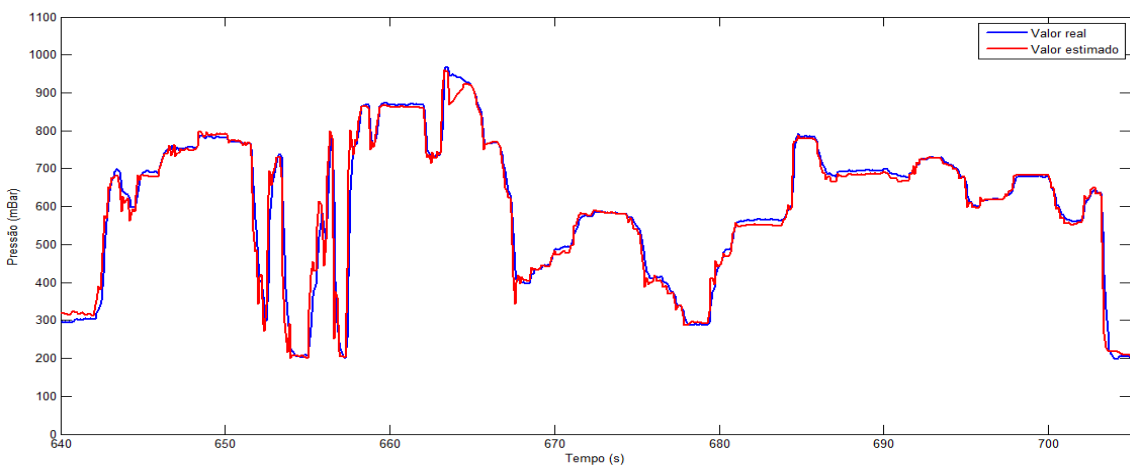


FIGURA 51: Detalhe do resultado da validação entre 640 e 710 segundos

FONTE: O autor (2014)

### 7.1.3.2.2 Análise do algoritmo HSADE-v2

Nesta seção, serão apresentados os resultados experimentais obtidos através da aplicação do algoritmo HSADE-v2 na realização da calibração automática do modelo de cálculo da pressão do ar no coletor de admissão.

A calibração dos parâmetros do modelo de cálculo da pressão do coletor de admissão foi gerada através da aplicação do algoritmo HSADE-v2 sobre o conjunto de dados de otimização.

O resultado final do processo de otimização é ilustrado na FIGURA 52. Nesta figura é apresentada a comparação entre o valor real da pressão do coletor de admissão medida pelo sensor e o valor da pressão do coletor calculado através do modelo, utilizando-se a calibração gerada pelo algoritmo HSADE-v2. Esta comparação é apresentada para todo o conjunto de otimização.

Para uma melhor visualização do resultado do processo de otimização, as FIGURAS 53, 54, 55, 56 e 57 ilustram em maiores detalhes a comparação entre o valor real da pressão no coletor medida pelo sensor e o valor da pressão no coletor calculado através do modelo para alguns trechos do conjunto de otimização.

Como pode ser observado nas FIGURAS 52 a 57, o resultado final obtido após o processo de otimização foi excelente e parecido com o resultado obtido para a primeira versão do algoritmo *HSADE*.

Assim como havia sido observado anteriormente, verifica-se que o valor calculado pelo modelo é bem próximo do valor real

Nota-se, também, que o valor estimado não é exatamente igual ao valor real, ou seja, o erro de estimação não foi nulo (o valor do EMQ obtido ao final do processo de otimização foi de 1720,3). Mas, como já discutido na seção 7.1.3.2.1, existe um erro intrínseco ao modelo, independente da calibração, assim não é possível eliminá-lo.

Sendo assim, tendo em vista a precisão do modelo, os resultados experimentais demonstram que a calibração gerada para os dados de otimização foi corretamente calculada.

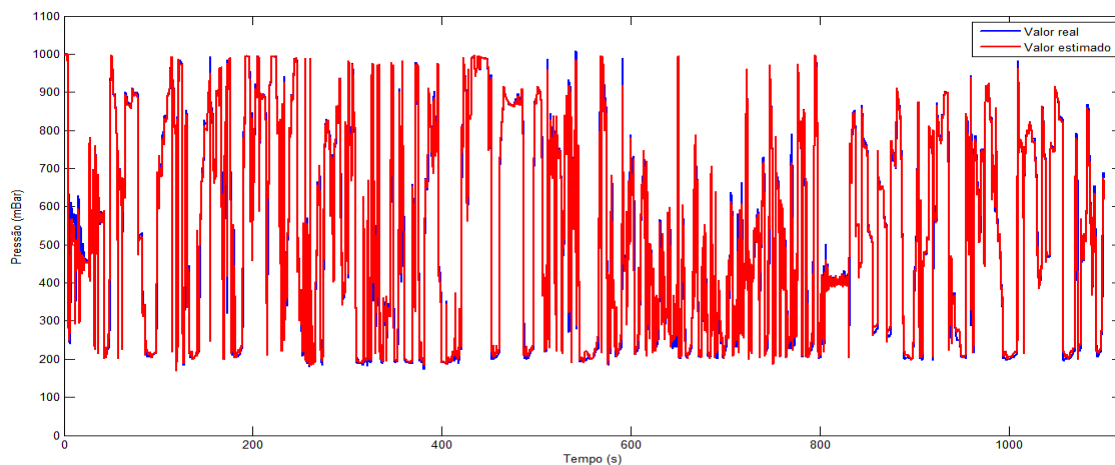


FIGURA 52: Otimização: comparação entre o valor real e o valor estimado para a pressão do ar no coletor de admissão

FONTE: O autor (2014)

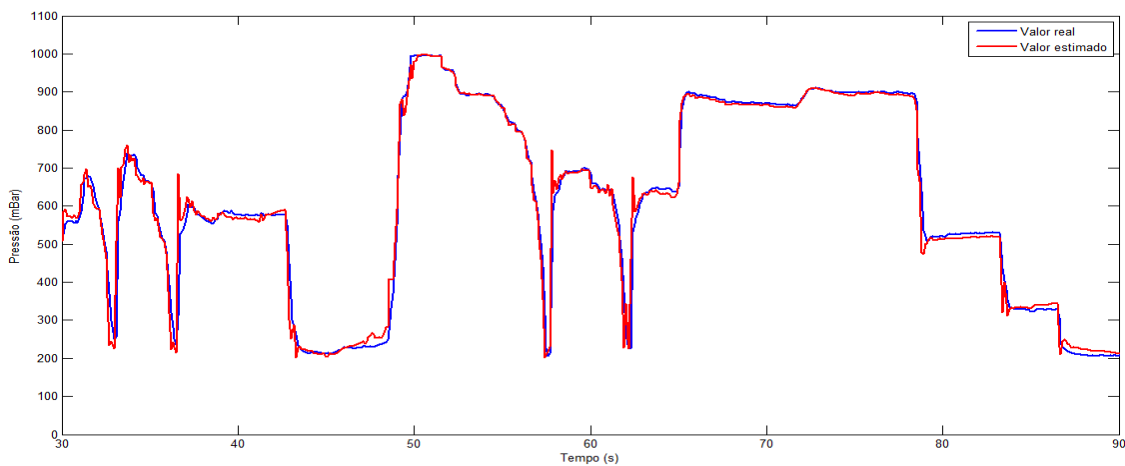


FIGURA 53: Detalhe do resultado processo de otimização entre 30 e 90 segundos

FONTE: O autor (2014)

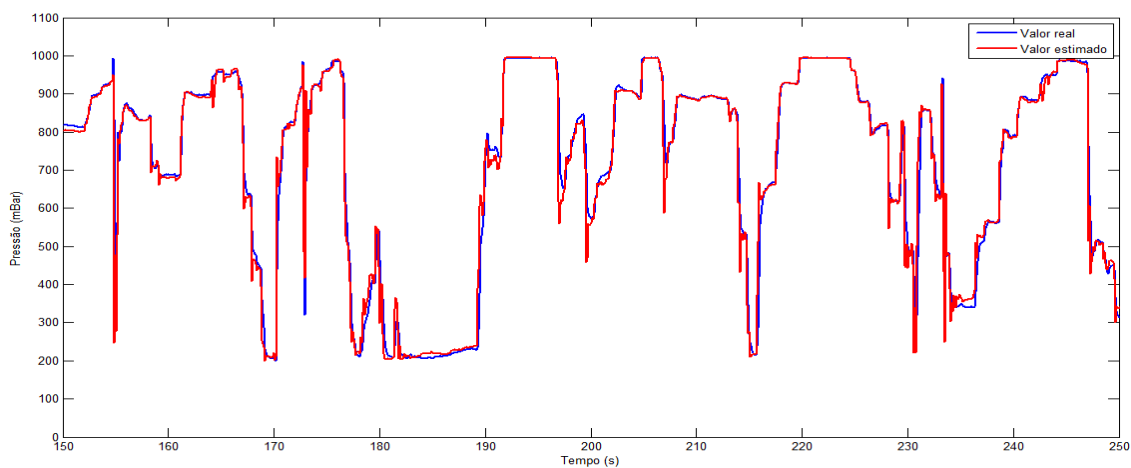


FIGURA 54: Detalhe do resultado processo de otimização entre 150 e 250 segundos

FONTE: O autor (2014)

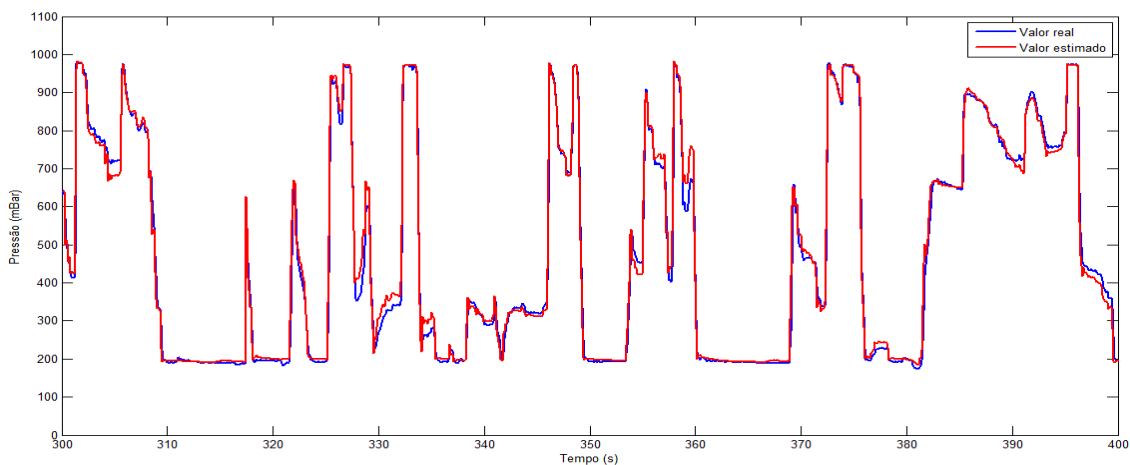


FIGURA 55: Detalhe do resultado processo de otimização entre 300 e 400 segundos

FONTE: O autor (2014)

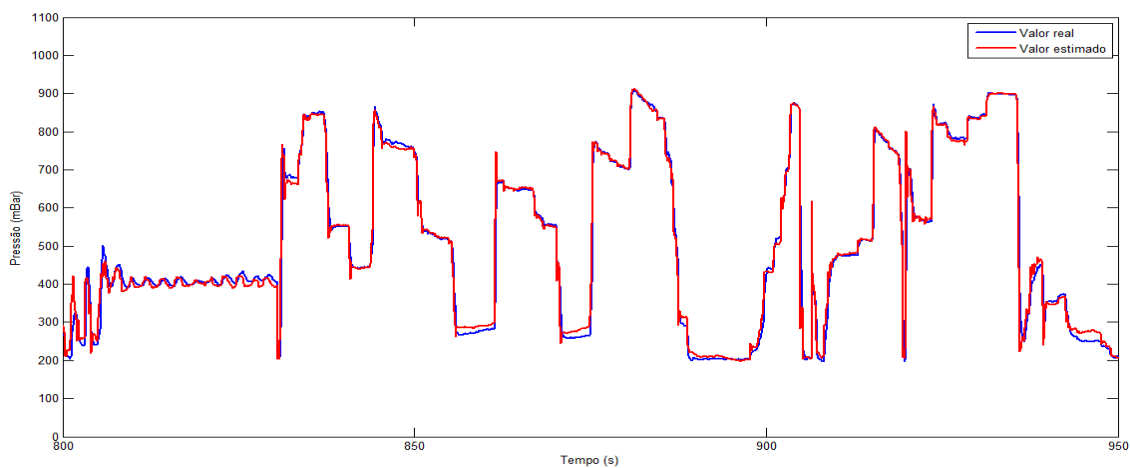


FIGURA 56: Detalhe do resultado processo de otimização entre 800 e 950 segundos

FONTE: O autor (2014)

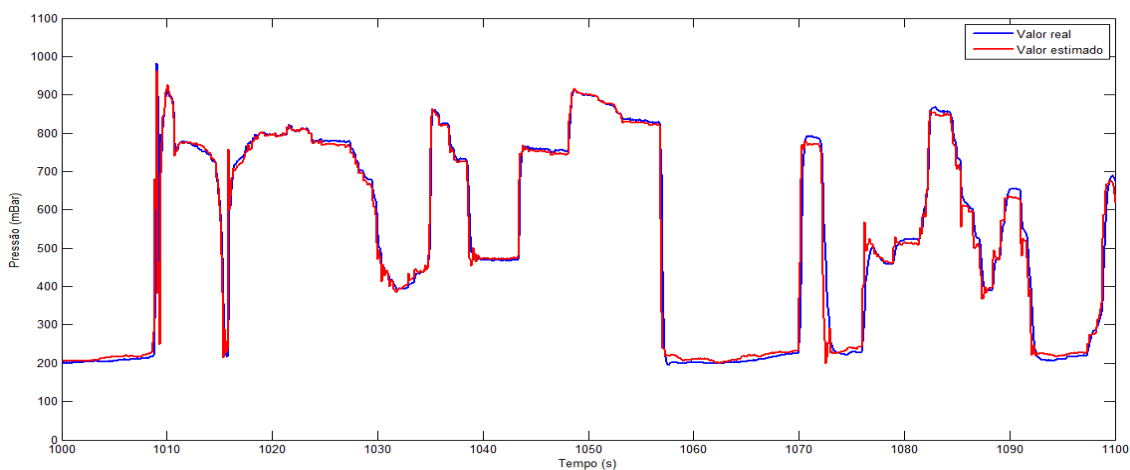


FIGURA 57: Detalhe do resultado processo de otimização entre 1000 e 1100 segundos

FONTE: O autor (2014)

Uma vez gerada a calibração, o conjunto de dados de validação foi utilizado para qualificar a calibração do modelo de cálculo da pressão do ar no coletor de admissão.

A FIGURA 58 apresenta a comparação entre o valor real da pressão do coletor de admissão medida pelo sensor e o valor da pressão do coletor de admissão calculado através do modelo a partir dos dados do conjunto de validação, utilizando-se a calibração gerada pelo algoritmo HSADE-v2.

Para uma melhor visualização do resultado da validação, as FIGURAS 59, 60, 61 e 62 ilustram em maiores detalhes a comparação entre o valor real da pressão do coletor medida pelo sensor e o valor da pressão do coletor calculado através do modelo utilizando-se a calibração gerada pelo algoritmo HSADE-v2 para alguns trechos do conjunto de validação.

Como pode ser observado nas FIGURAS 58 a 62, o resultado obtido a partir dos dados de validação foi muito bom. Verifica-se que o valor calculado pelo modelo é bem próximo do valor real, validando, portanto, a calibração gerada pelo algoritmo HSADE-v2.

Assim como havia sido observado com os dados de otimização, o erro de estimação não foi nulo (o valor do EMQ obtido para o conjunto de dados de validação foi de 1486,5). Mas, como já foi discutido, existe um erro intrínseco ao modelo, independente da calibração, assim não é possível eliminá-lo.

Em suma, os resultados apresentados nesta seção demonstram que o algoritmo HSADE-v2 foi capaz gerar uma calibração adequada e confiável para o modelo de cálculo da pressão do *rail*, comprovando a factibilidade e eficácia do uso do algoritmo HSADE-v2 na realização de calibração automática de funções de *software* de sistema de controle de motor.



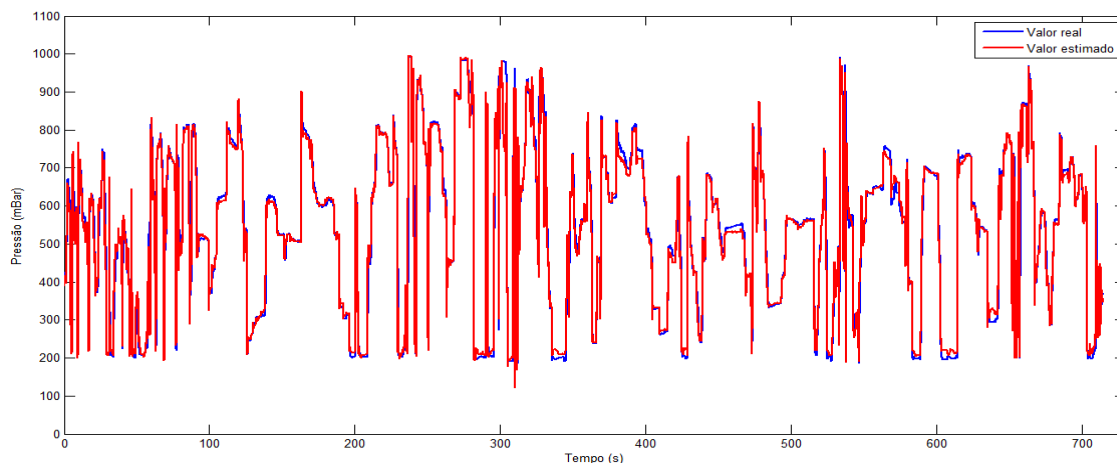


FIGURA 58: Validação: comparação entre o valor real e o valor estimado para a pressão do ar no coletor de admissão

FONTE: O autor (2014)

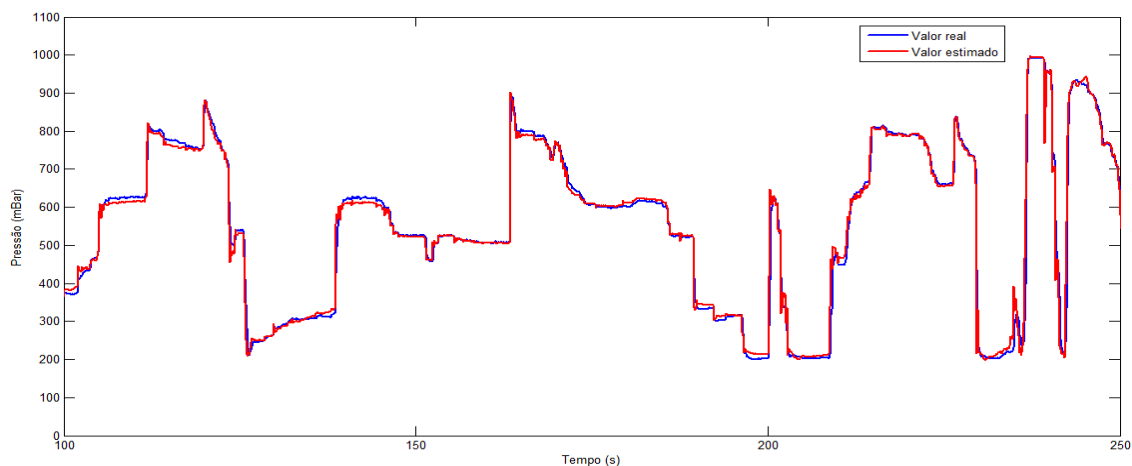


FIGURA 59: Detalhe do resultado da validação entre 100 e 250 segundos

FONTE: O autor (2014)

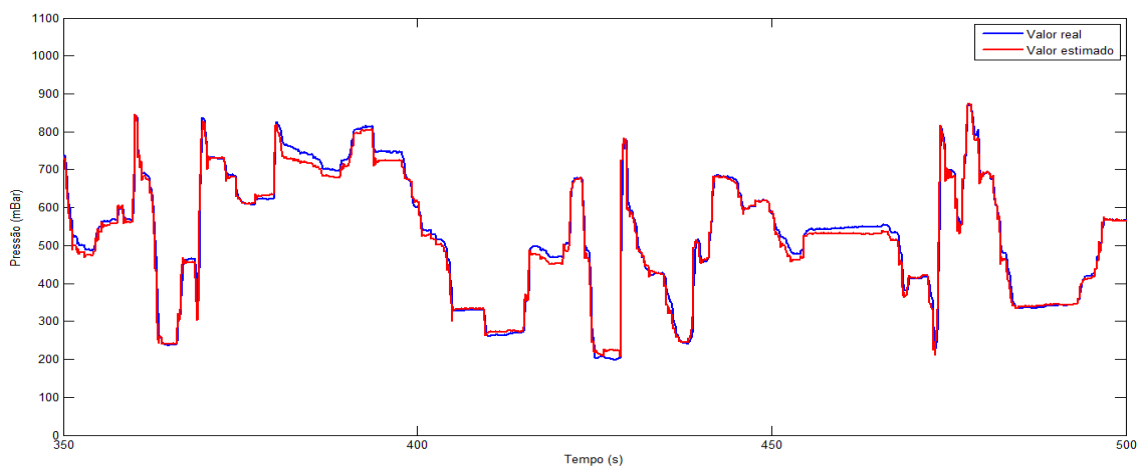


FIGURA 60: Detalhe do resultado da validação entre 350 e 500 segundos

FONTE: O autor (2014)

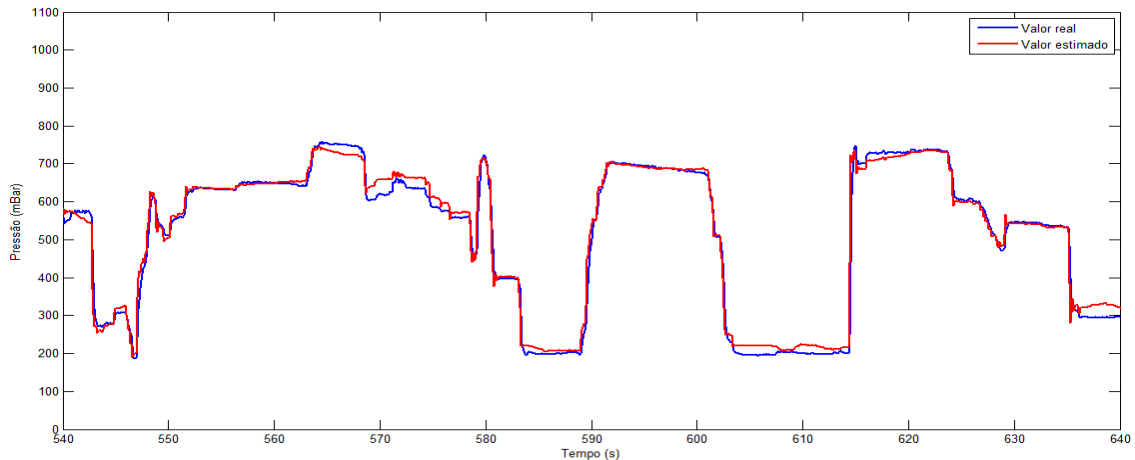


FIGURA 61: Detalhe do resultado da validação entre 540 e 640 segundos

FONTE: O autor (2014)

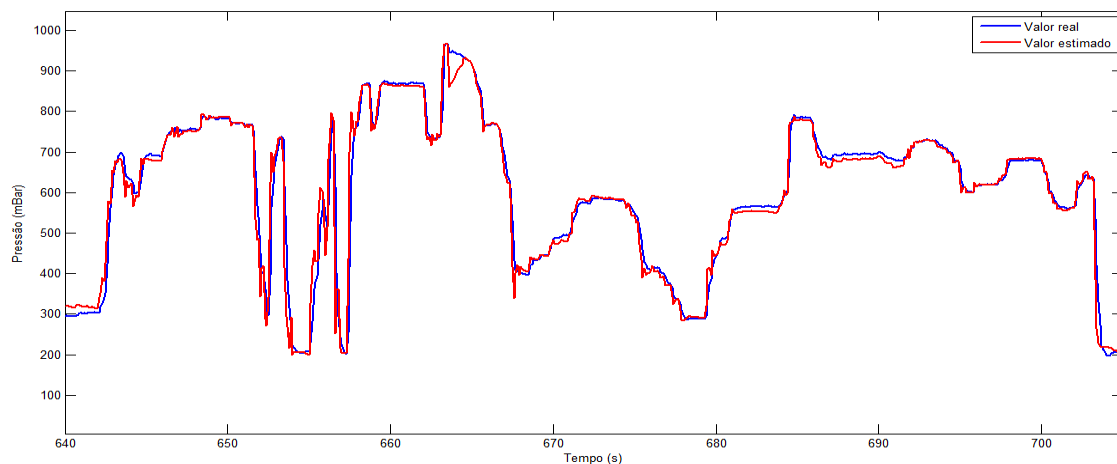


FIGURA 62: Detalhe do resultado da validação entre 640 e 710 segundos

FONTE: O autor (2014)

#### 7.1.4 Análise de desempenho

A fim de realizar uma análise mais detalhada sobre o desempenho das abordagens apresentadas nesta dissertação nos estudos de caso relativos à calibração automática, cada uma das versões do algoritmo HSADE será comparada aos oito algoritmos de comparação utilizados na seção 6 (JADE com e sem arquivo, jDE, CoDE, EPSDE, SaDE, DEGL, DEahcSPX)

Cada uma das duas versões do algoritmo HSADE, juntamente com os oito de comparação, serão aplicados aos dois estudos de caso relativos à calibração

automática e uma análise de desempenho utilizando-se o teste de *Mann-Whitney*, conforme descrito na 6.2.1, será realizado.

Para reduzir o impacto da natureza estocástica dos algoritmos, cada algoritmo foi executado 50 vezes para cada estudo de caso. Além disto, cada experimento foi finalizado após um número máximo de 120000 avaliações da função objetivo.

O desempenho será avaliado diretamente de acordo com o valor do EMQ obtido em cada experimento que é definido como  $EMQ(\vec{X}_{Melhor})$  onde  $\vec{X}_{Melhor}$  representa a melhor solução encontrada pelo algoritmo de otimização em um experimento. Para cada algoritmo e para cada estudo de caso, o valor do EMQ foi gravado ao final de cada experimento.

Com relação aos parâmetros de cada algoritmo, para os algoritmos HSADE-v1 e HSADE-v2 foram utilizados os mesmos parâmetros definidos na seção 7.1.2. Já para os algoritmos de comparação (JADE com e sem arquivo, jDE, SaDE, EPSDE, DEGL, DEahcSPX e CoDE), com exceção ao tamanho da população, todos os outros parâmetros foram definidos segundo a orientação dos autores dos mesmos. Com relação ao tamanho da população, este foi definido como  $NP = 120$  para JADE com e sem arquivo, jDE, SaDE, EPSDE, DEGL, DEahcSPX e  $NP = 40$  para CoDE. O valor de  $NP = 40$  para CoDE foi escolhido devido ao fato de que este algoritmo calcula três vetores experimentais para cada vetor alvo, o que seria equivalente a uma população de 120 indivíduos quando comparado aos outros algoritmos.

Nas próximas seções, uma análise mais detalhada dos resultados experimentais obtidos nos estudos de caso relativos à calibração automática será apresentada.

#### 7.1.4.1 Resultados experimentais e discussão

As TABELAS 60 e 61 apresentam a média e o desvio padrão do valor de EMQ, calculados sobre 50 experimentos, obtidos para cada um dos algoritmos: JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, HSADE-v1 e HSADE-v2

TABELA 60 – MÉDIA E DESVIO PADRÃO DO VALOR DO EMQ OBTIDOS PARA O MODELO DE CÁLCULO DA PRESSÃO DO RAIL

JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE
9,79E+01 ± 3,76E-03	9,80E+01 ± 3,37E-02	9,87E+01 ± 1,50E-01	9,98E+01 ± 3,69E-01	9,81E+01 ± 5,29E-02

EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
9,81E+01 ± 7,21E-02	1,03E+02 ± 1,50E+00	3,56E+02 ± 1,61E+01	9,79E+01 ± 4,98E-03	9,80E+01 ± 4,54E-02

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do valor da função objetivo calculados sobre 50 experimentos, sendo o valor do EMQ de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 120.000.

TABELA 61 – MÉDIA E DESVIO PADRÃO DO VALOR DO EMQ OBTIDOS PARA O MODELO DE CÁLCULO DA PRESSÃO NO COLETOR DE ADMISSÃO

JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE
1,72E+03 ± 4,61E-01	1,72E+03 ± 3,54E+00	1,77E+03 ± 8,86E+00	1,98E+03 ± 4,09E+01	1,73E+03 ± 2,17E+00

EPSDE	DEGL	DEahcSPX	HSaDE-v1	HSaDE-v2
1,78E+03 ± 1,24E+01	2,00E+03 ± 7,61E+01	1,67E+04 ± 7,98E+02	1,72E+03 ± 5,45E-01	1,73E+03 ± 3,53E+00

FONTE: O autor (2014)

NOTA: Dados apresentados na forma *média ± desvio padrão*.

Média e desvio padrão do valor da função objetivo calculados sobre 50 experimentos, sendo o valor do EMQ de cada experimento obtido após um número máximo de avaliações da função objetivo igual a 120.000.

As TABELAS 62 e 63 apresentam o resultado do teste de *Mann-Whitney* para um nível de significância 0,05, considerando HSADE-v1 e HSADE-v2 como algoritmo de controle respectivamente. Os sinais "+", "≈" e "-" indicam se o algoritmo de controle apresentou um desempenho melhor, semelhante ou pior,

respectivamente, do que o algoritmo de comparação para o problema correspondente.

TABELA 62 – RESULTADOS DO TESTE DE *MANN-WHITNEY* CONSIDERANDO HSADE-v1 COMO ALGORITMO DE CONTROLE

<i>Estudo de caso</i>	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Pressão no rail</i>	-	+	+	+	+	+	+	+
<i>Pressão no coletor</i>	-	+	+	+	+	+	+	+

FONTE: O autor (2014)

TABELA 63 – RESULTADOS DO TESTE DE *MANN-WHITNEY* CONSIDERANDO HSADE-v2 COMO ALGORITMO DE CONTROLE

<i>Estudo de caso</i>	JADE com arquivo	JADE sem arquivo	CoDE	SaDE	jDE	EPSDE	DEGL	DEahcSPX
<i>Pressão no rail</i>	-	≈	+	+	+	+	+	+
<i>Pressão no coletor</i>	-	-	+	+	≈	+	+	+

FONTE: O autor (2014)

Adicionalmente aos resultados acima apresentados, as FIGURAS 63 e 64 ilustram a evolução do valor do EMQ durante o processo de otimização para a modelo de cálculo da pressão do rail e para o modelo de cálculo da pressão no coletor, respectivamente.

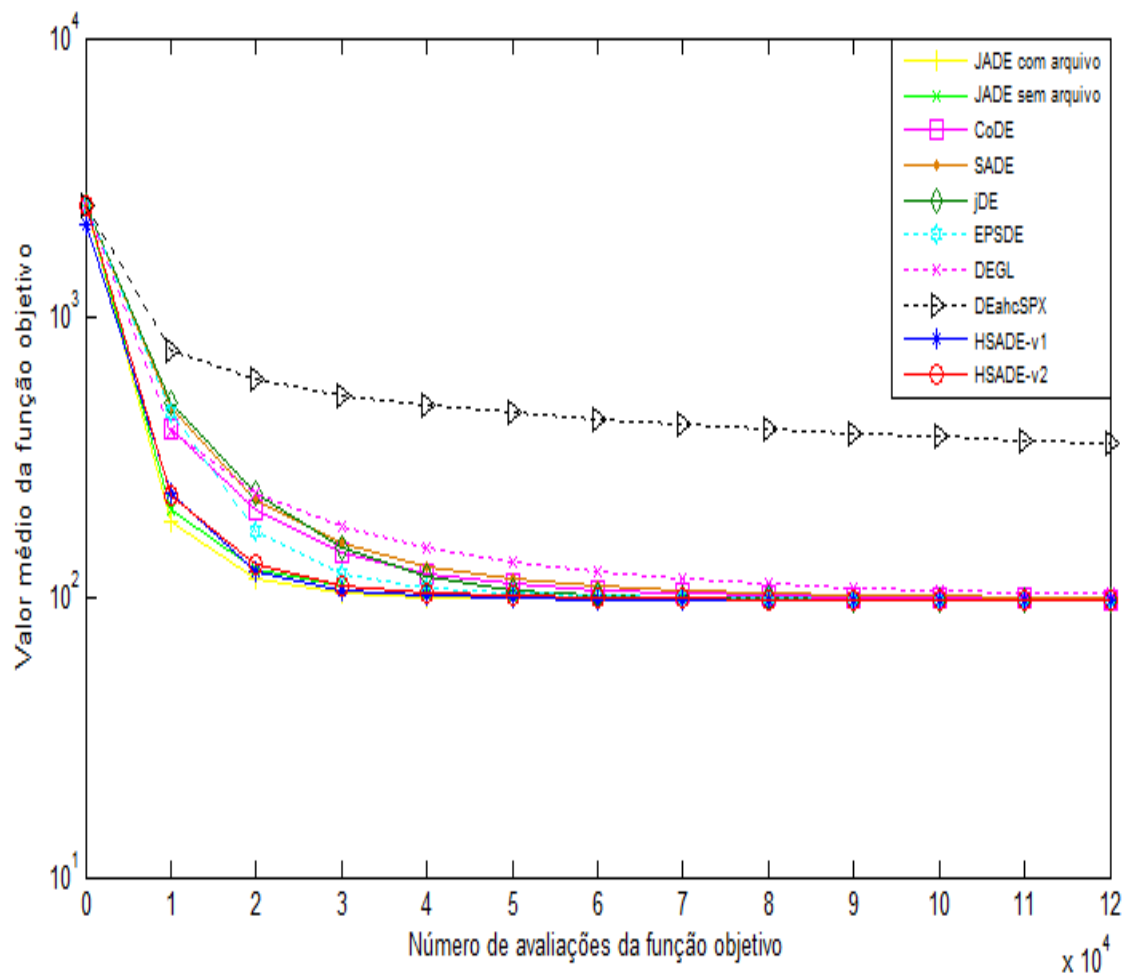


FIGURA 63: Evolução do EMQ para o modelo de pressão do *rail* durante o processo de otimização  
 FONTE: O autor (2014)

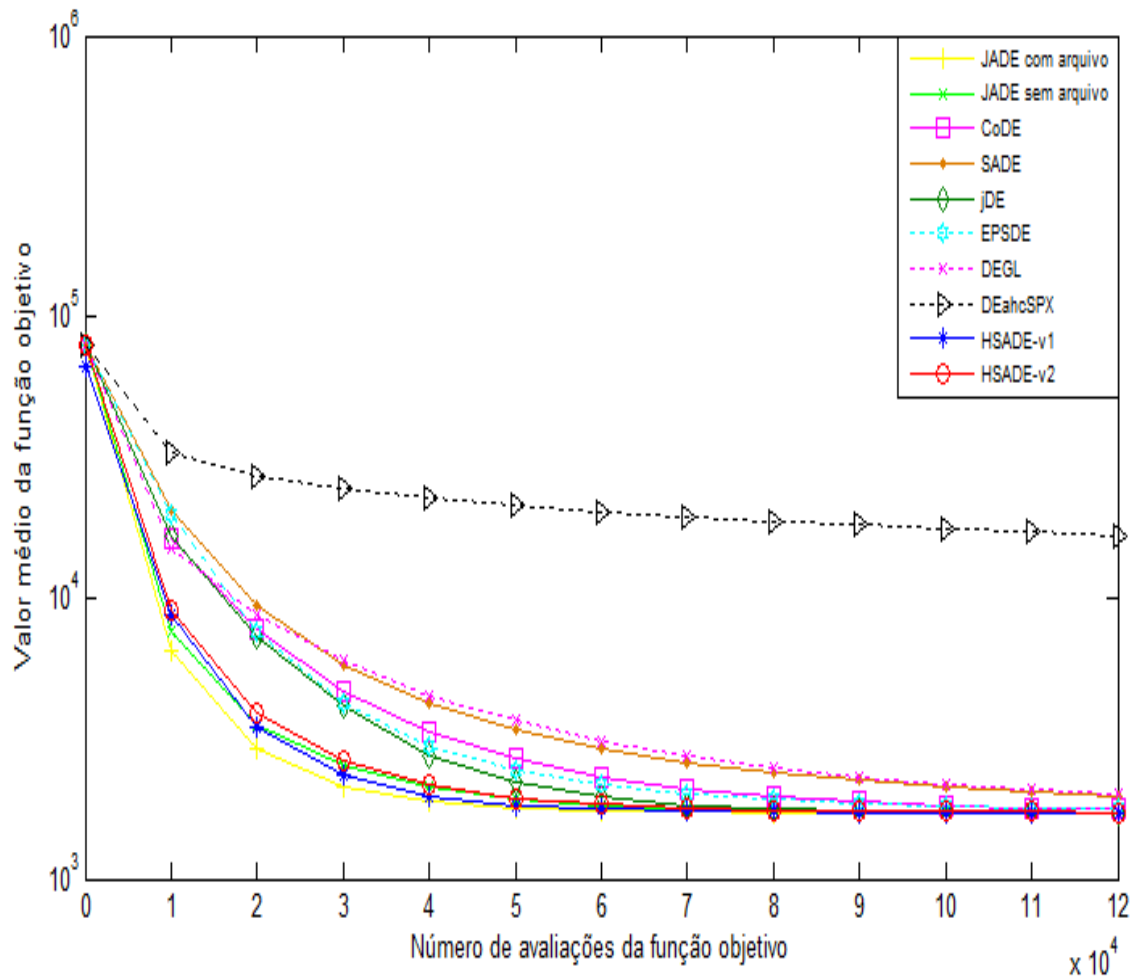


FIGURA 64: Evolução do EMQ para o modelo da pressão no coletor durante o processo de otimização

FONTE: O autor (2014)

Primeiramente, com relação ao algoritmo HSADE-v1, observa-se que este algoritmo apresentou um desempenho muito bom. Nota-se a partir da TABELA 62 que HSADE-v1 foi superior aos sete algoritmos de comparação (JADE *sem arquivo*, CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX) nos dois estudos de casos relativos à calibração automática abordados nesta dissertação, de acordo com o teste de *Mann-Whitney*.

De acordo com o teste de *Mann-Whitney*, o algoritmo HSADE-v1 foi superado apenas por JADE *com arquivo*. Entretanto, analisando-se o valor médio do EMQ obtido nos dois estudos de casos (TABELA 62) e a evolução do EMQ durante o processo de otimização para os dois algoritmos HSADE-v1 e JADE *com arquivo*

(FIGURAS 63 e 64), pode-se dizer que não há uma diferença de desempenho significativa entre estes dois algoritmos.

Este fato fica claro quando se compara, por exemplo, o valor estimado da pressão do coletor de admissão utilizando-se a calibração gerada pelo algoritmo HSADE-v1 e a calibração gerada pelo algoritmo JADE *com arquivo*. Esta comparação é apresentada na FIGURA 65 para os dados do conjunto de otimização.

Como pode ser observado na FIGURA 65, basicamente não existe uma diferença significativa entre os resultados obtidos por HSADE-v1 e JADE *com arquivo*.

Sendo assim, pode-se dizer, de uma maneira global, que o algoritmo HSADE-v1 apresentou um desempenho, melhor, ou no mínimo similar que os algoritmos de comparação, comprovando, desta forma, seu bom desempenho na realização de calibração automática.

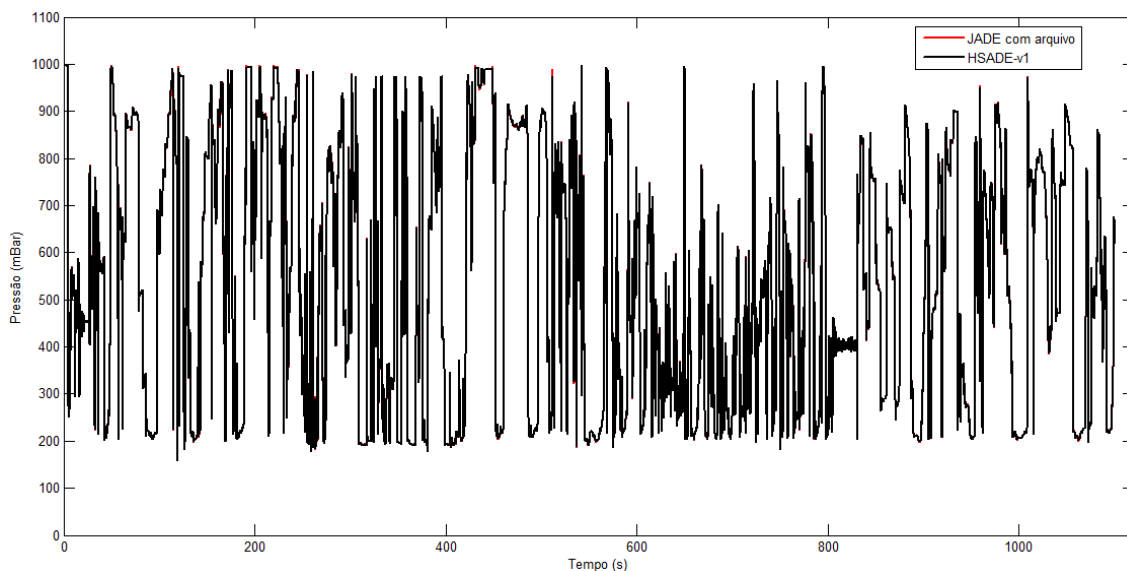


FIGURA 65: Comparação do valor estimado da pressão do *rail* utilizando-se a calibração gerada pelo algoritmo HSADE-v1 e pelo algoritmo JADE *com arquivo*

FONTE: O autor (2014)

Com relação ao algoritmo HSADE-v2, observa-se que este algoritmo também apresentou um desempenho muito bom. Como pode ser verificado a partir da TABELA 63, HSADE-v2 foi superior ou no mínimo apresentou desempenho semelhante que sete algoritmos de comparação (JADE *sem arquivo*, CoDE, SaDE,



jDE, EPSDE, DEGL e DEahcSPX) para o estudo de caso relativo a calibração do modelo de cálculo da pressão do *rail* e foi superior ou no mínimo apresentou desempenho semelhante que seis algoritmos de comparação (CoDE, SaDE, jDE, EPSDE, DEGL e DEahcSPX) para o estudo de caso relativo a calibração do modelo de cálculo da pressão do coletor de admissão.

De acordo com o teste de *Mann-Whitney*, o algoritmo HSADE-v2 foi superado apenas por JADE *com arquivo* no primeiro estudo de caso e superado por JADE com e sem arquivo no segundo estudo de caso. Mas assim, como foi observado para a primeira versão de HSADE, analisando-se o valor médio do EMQ obtido nos dois estudos de casos (TABELA 63) e a evolução do EMQ durante o processo de otimização para os dois algoritmos HSADE-v2 e JADE com e sem arquivo (FIGURAS 63 e 64), pode-se dizer que não há uma diferença de desempenho significativa entre estes três algoritmos.

Este fato também fica claro quando se compara, por exemplo, o valor estimado da pressão do coletor de admissão utilizando-se a calibração gerada pelo algoritmo HSADE-v2 e pelo algoritmo JADE com e sem arquivo. Esta comparação é apresentada na FIGURA 68 para os dados do conjunto de otimização.

Como pode ser observado na FIGURA 68, basicamente não existe uma diferença significativa entre os resultados obtidos por HSADE-v12 e JADE com e sem arquivo.

Sendo assim, pode-se dizer, de uma maneira global, que o algoritmo HSADE-v2 também apresentou um desempenho, melhor, ou no mínimo similar que os algoritmos de comparação, comprovando, desta forma, seu bom desempenho na realização de calibração automática.

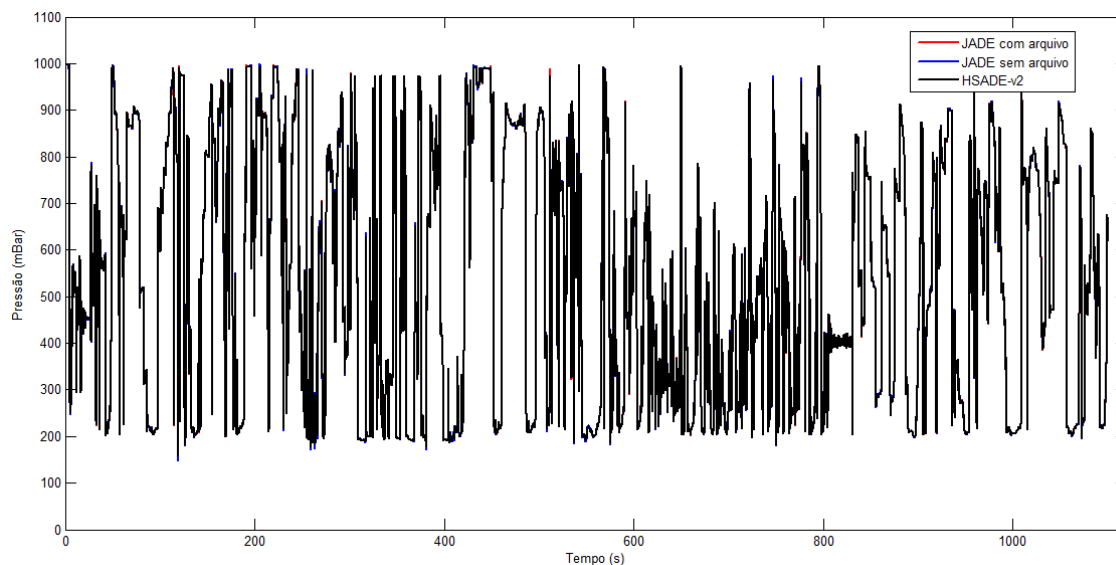


FIGURA 66: Comparação do valor estimado da pressão do *rail* utilizando-se a calibração gerada pelo algoritmo HSADE-v1 e pelo algoritmo JADE *com arquivo*

FONTE: O autor (2014)

#### 7.1.5 Análise geral dos algoritmos propostos

Os resultados apresentados nas seções 7.1.2 e 7.1.3 mostram que os algoritmos de otimização HSADE-v1 e HSADE-v2 são ferramentas úteis no processo de calibração. Além disto, estes resultados mostram que ambos os algoritmos, HSADE-v1 e HSADE-v2, foram capazes de gerar uma calibração adequada e confiável para os modelos da pressão do combustível no *rail* e da pressão do ar no coletor de admissão, comprovando, desta forma, a factibilidade e eficácia do uso dos algoritmos HSADE-v1 e HSADE-v2, na realização de calibração automática de funções de *software* de sistema de controle de motor.

Como já abordado anteriormente, os objetivos de custo, qualidade e prazo são fatores cada vez mais importantes no processo de calibração, ou seja, cada vez mais é necessário realizar a calibração de funções de *software* de sistemas de controle de motor em um curto prazo, com custos cada vez mais reduzidos, mas sempre com a qualidade necessária. Neste sentido, a utilização dos algoritmos HSADE-v1 e HSADE-v2 no processo de calibração demonstraram ser ferramentas úteis para que estes objetivos possam ser alcançados.

Na sequência é apresentada com mais detalhes a contribuição da utilização dos algoritmos HSADE-v1 e HSADE-v2 no processo de calibração para cada um dos objetivos de custo, qualidade e prazo:

- Prazo:

A utilização dos algoritmos HSADE-v1 e HSADE-v2 na realização de calibração automática permite a realização da calibração de um número elevado de parâmetros em um curto período de tempo, diminuindo, portanto, o tempo do processo de calibração.

Em termos de número de parâmetros, no primeiro estudo de caso (estimação da pressão do *rail*) existiam 300 parâmetros a serem calibrados, enquanto no segundo estudo de caso (estimação da pressão do coletor de admissão) existiam 400 parâmetros a serem calibrados.

Para se ter uma idéia do potencial de redução de tempo de calibração, caso as funções abordadas nesta dissertação fossem calibradas manualmente, seriam necessários por volta de dois a três dias para realização da calibração. Já com a utilização dos algoritmos HSADE-v1 e HSADE-v2 para a realização da calibração automática, foram necessárias em média duas horas. Neste tempo de duas horas, foram consideradas as seguintes atividades:

- Geração dos dados para calibração (dados de otimização e validação);
- Realização da calibração automática;
- Análise e validação da calibração.

Vale salientar que no caso da calibração automática dos estudos de caso utilizados neste trabalho, o maior tempo está relacionado à geração de dados e na análise e validação da calibração. O tempo para a realização da calibração automática em si foi muito curto. Os algoritmos HSADE-v1 e HSADE-v2 levaram em média 5 minutos para realizar o processo de otimização e gerar, portanto, uma calibração.

Sendo assim, no caso dos estudos de caso utilizados neste trabalho, a redução de tempo do processo de calibração foi de aproximadamente 90% o que demonstra o grande benefício da utilização dos algoritmos HSADE-v1 e HSADE-v2 para realização de calibração automática.

- Custo:

A utilização dos algoritmos HSADE-v1 e HSADE-v2 na realização de calibração automática gera uma redução de custos no processo de calibração devido ao fato de não ser necessário para realização da calibração, o uso de meios de ensaios que possuem um alto custo de utilização, tais como dinamômetro de rolo ou dinamômetro de motores.

Todo o processo de calibração dos dois estudos de caso abordados nesta dissertação foi realizado utilizando-se apenas dados gerados diretamente em veículo.

Além disso, a redução do tempo necessário para calibração também gera uma redução de custo com recursos humano, pois menos tempo de calibração implica em menos tempo de recursos de engenharia para realizar a calibração.

- Qualidade

Além de uma redução no custo, e uma diminuição no tempo necessário para realizar a calibração, uma ferramenta de calibração automática deve ser capaz de gerar uma calibração robusta e confiável.

Como pode ser observado nas seções 7.1.2 e 7.1.3, os resultados obtidos a partir dos dados de validação demonstraram que as calibrações geradas pelos algoritmos HSADE-v1 e HSADE-v2 foram adequadas e robustas, confirmando, desta forma, a eficiência e a confiabilidade dos algoritmos propostos neste trabalho.

## 8 CONCLUSÕES E TRABALHOS FUTUROS

Um dos objetivos deste trabalho era o desenvolvimento de uma nova versão de um algoritmo baseado na evolução diferencial para solução de problemas mono-objetivo sujeito apenas a restrições laterais. Neste sentido, uma nova versão de um algoritmo de abordagem de evolução diferencial híbrida auto adaptativa, denominado HSADE foi apresentado. Mais especificamente, duas abordagens do algoritmo HSADE foram desenvolvidas e denominadas HSADE-v1 e HSADE-v2. O desenvolvimento das duas abordagens apresentadas nesta dissertação teve como finalidade a implementação de um algoritmo capaz de ter um bom equilíbrio entre o *exploration* e *exploitation* e que também fosse capaz de adaptar os seus valores de parâmetros de controle durante o processo de otimização.

Para que isto fosse possível, em ambos os algoritmos HSADE-v1 e HSADE-v2, uma nova abordagem híbrida que visa uma sinergia entre uma busca voltada à *exploitation* e uma abordagem de evolução diferencial de forma a aprimorar o seu desempenho foi desenvolvida.

Com relação à busca voltada à *exploitation*, um novo operador foi implementado com a finalidade de aumentar a velocidade de convergência do processo de otimização. Neste operador, os vetores experimentais são gerados na vizinhança da população atual.

Com relação à abordagem de evolução diferencial, um novo método de adaptação dos parâmetros de controle durante o processo de otimização foi desenvolvido para ambos os algoritmos HSADE-v1 e HSADE-v2. Cada algoritmo possui um processo de adaptação diferente, mas em ambos os casos, o fator de mutação e taxa de cruzamento são evoluídos baseando-se nos valores dos parâmetros bem sucedidos de gerações passadas.

Além da adaptação dos parâmetros de controle, no algoritmo HSADE-v1 um processo de adaptação da estratégia de geração do vetor experimental durante o processo de otimização foi desenvolvido visando melhorar o equilíbrio entre o *exploration* e *exploitation*. Neste processo, duas estratégias de geração do vetor experimental foram escolhidas de tal forma que uma priorizasse o *exploration* e a outra priorizasse o *exploitation*. A escolha de qual estratégia utilizar é baseada em

uma probabilidade que é adaptada durante o processo de otimização baseando-se no histórico de vetores experimentais bem sucedidos nas gerações passadas

O desempenho dos dois algoritmos propostos foi avaliado utilizando-se para isto três estudos de caso compostos de conjuntos de funções propostas para o CEC2005, CEC2011 e CEC2013. Estes conjuntos de funções correspondem a problemas de minimização mono-objetivos sujeito apenas a restrições laterais, especificamente propostas para avaliar o desempenho de algoritmos evolutivos. Para validar os resultados obtidos nos três estudos de caso, os testes não paramétricos de *Mann-Whitney* e o teste *post hoc* de *Friedman* foram utilizados.

Os resultados experimentais demonstraram que os dois algoritmos HSADE-v1 e HSADE-v2 apresentaram um desempenho global muito bom e consistente na solução dos problemas de otimização propostos no CEC2005, CEC2011 e CEC2013 quando comparado aos algoritmos JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL e DEahcSPX, demonstrando a eficácia e a eficiência dos algoritmos propostos nesta dissertação. Além disso, ressalta-se que para determinados casos o algoritmo HSADE-v1 apresentou melhor desempenho e que para outros casos o algoritmo HSADE-v2 apresentou melhor desempenho, mas, fazendo-se uma análise de maneira global, pode-se dizer que o desempenho dos dois algoritmos HSADE-v1 e HSADE-v2 foram parecidos.

O segundo objetivo deste trabalho era a aplicação dos algoritmos propostos nesta dissertação na realização de calibração automática de parâmetros de funções de *softwares* utilizados para o controle e funcionamento de motores automotivos.

Neste sentido, dois estudos de caso foram selecionados. O primeiro estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão de combustível no *rail* em motores diesel com sistema *common rail* e o segundo estudo de caso refere-se à calibração automática de um modelo responsável por calcular a pressão do ar no coletor de admissão em motores a gasolina.

Os resultados experimentais mostram que os algoritmos de otimização HSADE-v1 e HSADE-v2 foram capazes de gerar de forma autônoma uma calibração adequada e robusta para ambos os estudos de casos, comprovando, desta forma, a factibilidade e a eficiência da utilização dos algoritmos desenvolvidos neste trabalho na realização de calibração automática.

No cenário automotivo no qual os objetivos de prazo, custo e qualidade são importantes no desenvolvimento de projetos, os algoritmos de otimização HSADE-v1 e HSADE-v2 demonstraram ser ferramentas computacionais úteis no processo de calibração, podendo, assim, ser utilizados para substituir o ajuste manual dos parâmetros de funções do *software* da ECU, ajudando, desta forma, para que os objetivos de prazo, custo e qualidade possam ser alcançados.

Entre os possíveis trabalhos futuros destacam-se:

- Neste trabalho, os algoritmos propostos foram aplicados na realização da calibração automática de funções que representam modelos de estimação. Sendo assim, uma proposta de trabalho futuro seria a extensão da aplicação dos algoritmos de otimização na calibração de outros tipos de funções utilizadas para o controle e gerenciamento de motores automotivos.
- Desenvolver um algoritmo baseado na evolução diferencial para solução de problemas multi objetivo.
- Aplicação de um algoritmo multi objetivo para realização de calibração automática de funções utilizadas para o controle e gerenciamento de motores automotivos visto que alguns casos de calibração correspondem a uma otimização multi objetivo.
- Evoluir os algoritmos HSADE-v1 e HSADE-v2 de tal forma que restrições de igualdade e desigualdade sejam consideradas durante a otimização visto que muitos problemas reais possuem este tipo de restrições.

## REFERÊNCIAS

ARNOLD, D. V; BEYER, H. G. Evolution strategies with cumulative step length adaptation on the noisy parabolic ridge. **Natural Computing**. Berlin, Germany: Springer, v. 7, p. 555-587, 2008.

ANUAR, N. I. The role of time, cost and quality in project management. **Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)**. Singapore, 2011, p.630-634.

BASU, M. Improved differential evolution for short-term hydrothermal scheduling. **International Journal of Electrical Power & Energy Systems**, v. 58, p. 91–100, 2014.

BAZARAA, M. S.; JARVIS, J. J.; SHERALI, H. D. **Linear programming and network flows**. New Jersey , USA: Wiley, 1990.

BINITHA, S.; SATHYA, S.S. A survey of bio inspired optimization algorithms. **International Journal of Soft Computing and Engineering (IJSCE)**, v. 2, n. 2, p. 137–151, 2012.

BREST, J.; GREINER, S.; BOSKIVIC, B.; MERNIK, M.; ZUMER, V. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. **IEEE Transactions on Evolutionary Computation**, v. 10, n. 6, p. 646–657, 2006.

CHELLIAHA, T. R.; THANGARAJA, R.; ALLAMSETTYA, S.; PANT, M. Coordination of directional overcurrent relays using opposition based chaotic differential evolution algorithm. **International Journal of Electrical Power & Energy Systems**, v. 55, p. 341–350, 2014.

CHOTARD, A; AUGER, A.; HANSEN, N. Cumulative Step-size Adaptation on Linear Functions: Technical Report, Laboratoire de Recherche en Informatique, France, 2012. Disponível em: <[https://www.lri.fr/~chotard/Homepage\\_files/chotard2012TRcumulative.pdf](https://www.lri.fr/~chotard/Homepage_files/chotard2012TRcumulative.pdf)>. Acesso em: 10 abr. 2014.

CREPINSEK, M.; LUI, S.; MERNIK, M. Exploration and exploitation in evolutionary algorithms: A survey. **ACM Computing Surveys**, v. 45, n. 3, artigo n. 35, 2013.

DAS, S.; ABRAHAM, A.; CHAKRABORTY, U. K.; KONAR, A. Differential evolution using a neighborhood-based mutation operator. **IEEE Transactions on Evolutionary Computation**, v.13, n. 3, p. 526-553, 2009.

DAS, S.; SUGANTHAN, P. N. Differential evolution: A survey of the state-of-the-Art. **IEEE Transactions on Evolutionary Computation**, v.15, n. 1, p. 4-31, 2011.



DERRAC, J.; GARCIA, S.; MOLINA, D.; HERRERA, F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. **Swarm and Evolutionary Computation**, v. 1, n. 1, p. 3-18, 2011.

FRANCOIS, G. **New approaches in virtualization of ECU software development**. SAE Technical Paper 2013-01-0429, 2013.

GÄMPERLE, R.; MULLER, S. D.; KOUMOUTSAKOS, P. A parameter study for differential evolution, In: **Advances in intelligent systems, fuzzy systems, evolutionary computation**. WSEAS Press, 2002, p. 293–298.

HANSEN, N. The CMA evolution strategy: A comparing review. **Towards a new evolutionary computation**. Berlin, Germany: Springer, v. 192, p. 75-102, 2006.

HANSEN, N. **The CMA evolution strategy: A tutorial**. Laboratoire de Recherche en Informatique, France, 2011. Disponível em: <<https://www.lri.fr/~hansen/cmatutorial.pdf>> Acesso em: 10 abr. 2014.

HANSEN, N.; OSTERMEIER, A. Completely derandomized self-adaptation in evolution strategies. **Evolutionary Computation**, v. 9, n. 2, p. 159-195, 2001.

HUANG, V. L.; QIN, A. K.; SUGANTHAN, P. N., Self-adaptive differential evolution algorithm for constrained real-parameter optimization. In: IEEE Congress on Evolutionary Computation, 2006, Vancouver, Canada. **Proceedings of the IEEE Congress on Evolutionary Computation, 2006. CEC2006**. IEEE Press, 2006, p. 17-24.

LIANG, J. J.; QU, B. Y.; SUNGANTHAN, P. N.; HERNANDEZ-DIAS, A. G. Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter Optimization. **Technical Report** 201212, Zhengzhou University, China; Technical Report, Nanyang Technological University, Singapore, 2013. Disponível em: <<http://www.ntu.edu.sg/home/EPNSugan/>>. Acesso em: 03 nov. 2013.

JIANG, S.; NUTTER, D.; GULLITTI, A. **Implementation of model-based calibration for a gasoline engine**. SAE Technical Paper 2012-01-0722, 2012.

KAJI, H. **Automotive engine calibration with experiment-based evolutionary multi-objective optimization**. 2008, 160 f. Dissertação (Doctoral Thesis), Kyoto University, Japão, 2008.

LANGOUËT, K. H.; METIVIER, L.; SINOQUET, D.; TRAN, Q.H. Optimization for engine calibration, In: **EngOpt 2008 - International Conference on Engineering Optimization**, Rio de Janeiro, 2008. Disponível em: <[http://www.engopt.org/nukleo/pdfs/0265\\_engopt\\_calib2.pdf](http://www.engopt.org/nukleo/pdfs/0265_engopt_calib2.pdf)>. Acesso em: 13 nov. 2013.

LIBERATORE, M. J. Improving project management decision making by modeling quality, time, and cost continuously. **IEEE Transactions on Engineering Management**, v. 60, n. 3, p. 518-528, 2013.

MAIO, F. D.; BARONCHELLI, S.; ZIO, E. Hierarchical differential evolution for minimal cut sets identification: Application to nuclear safety system. **European Journal of Operational Research**, v.238, n. 2, p. 645-652, 2014.

MALLIPEDDI, R.; SUGANTHAN, P. N.; PAN, Q. K.; TASGETIREN, M. F. Differential evolution algorithm with ensemble of parameters and mutation strategies. **Applied Soft Computing**, v. 11, n. 2, p. 1679–1696, 2011.

MENON, S.; VENUGOPAL, P. **AUTOSAR software platform adoption: Systems engineering strategies**. SAE Technical Paper 2014-01-0289, 2014.

MONTGOMERY, D. C.; RUNGER, G. C. **Applied statistics and probability for engineers**. 6<sup>th</sup> ed. USA: Wiley, 2010

NOMAN, N.; IBA, H. Accelerating differential evolution using an adaptive local search. **IEEE Transaction on Evolutionary Computation**, v.12, n. 1, p. 107-125, 2008.

PENG, F.; TANG, K.; CHEN, G.; YAO, X. Multi-start JADE with knowledge transfer for numerical optimization. In: IEEE Congress on Evolutionary Computation, 2009. CEC '09. Norway. **Proceedings of the IEEE Congress on Evolutionary Computation, 2009**. IEEE Press, 2009, p. 1889-1895.

PRICE, K. V. An introduction to differential evolution. **New ideas in optimization**, London: McGraw-Hill, 1999, p. 79–108.

PRICE, K. V.; STORN, R. M.; LAMPINEN, J. A. **Differential evolution - A practical approach to global optimization**. Berlin, Germany: Springer, 2005.

QIN, A. K.; HUANG, V. L.; SUGANTHAN, P. N., Differential evolution algorithm with strategy adaptation for global numerical optimization. **IEEE Transactions on Evolutionary Computation**, v.13, n. 2, p. 398-417, 2009.

RAHNAMAYAN, S.; TIZHOOSH, H. R.; SALAMA, M. M. A. Quasi-oppositional differential evolution. In: IEEE Congress on Evolutionary Computation, 2007, Singapore. **Proceedings of the IEEE Congress on Evolutionary Computation, 2007, CEC2007**, 2007, p. 2229-2236.

ROGALSKY, T.; DERKSEN, R. W.; KOCABIYIK, S. Differential evolution in aerodynamic optimization. **Proceedings of 46th Annual Conference of Canadian Aeronautics and Space Institute**, Montreal, Quebec, Canada, 1999, p. 29–36.

RÖNKKÖNEN, J.; KUKKONEN, S.; PRICE, K. V. Real parameter optimization with differential evolution. In: IEEE Congress on Evolutionary Computation, 2005. Edinburgh, Scotland. **Proceedings of the IEEE Congress on Evolutionary Computation, 2005**. 2005, p. 506-513.

SALMAN, A. A.; AHMAD, I.; OMRAN, M. G. H; MOHAMMAD, M. G. Frequency assignment problem in satellite communications using differential evolution. **Computers & Operations Research**, v. 37, n. 12, p. 2152–2163, 2010.

SAMPATH, A. HEGDE, S. **Interoperability challenges in automotive embedded software**. SAE Technical Paper 2013-01-2847, 2013.

SARUHAN, H. Differential evolution and simulated annealing algorithms for mechanical systems design. **Engineering Science and Technology, an International Journal**. V. 17, n. 13, p. 131-136, 2014.

STORN, R. On the usage of differential evolution for function optimization. **Proceedings of the 1996 biennial conference of the North American fuzzy information processing society – NAFIPS**. New York: IEEE Press, 1996. p. 519–523,

STORN, R.; PRICE, K. Differential evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. **Technical Report TR-95-012**, International Computer Science Institute, Berkeley, USA, 1995. Disponível em: <<http://www1.icsi.berkeley.edu/~storn/litera.html>>. Acesso em: 02 set. 2013.

STORN, R.; PRICE, K. Differential evolution - A simple and efficient heuristic for global optimization over continuous. **Journal of Global Optimization**, v. 11, n. 4, p. 341–359, 1997.

SUGANTHAN, P. N. Differential evolution algorithm: Recent advances. **Theory and Practice of Natural Computing**, v. 7505, p. 30-46, 2012.

SWAGATAM, D.; SUGANTHAN, P. N. Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. **Technical Report**, Nanyang Technological University, Singapore, 2010. Disponível em: <<http://www.ntu.edu.sg/home/EPNSugan/>>. Acesso em: 03 nov. 2013.

SUGANTHAN, P. N.; HANSEN, N.; LIANG, J. J.; DEB, K.; CHEN, Y. P.; AUGER, A.; TIWARI, S. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. **Technical Report**, Nanyang Technological University, Singapore, KanGAL Report Number 2005005, Indian Institute of Technology Kanpur, India, 2005. Disponível em: <<http://www.ntu.edu.sg/home/EPNSugan/>>. Acesso em: 03 nov. 2013.

TSUTSUI, S.; YAMAMURA, M.; HIGUCHI, T. Multi-parent recombination with simplex crossover in real coded genetic algorithms. **Proceedings of the Genetic and Evolutionary Computation Conference**. Orlando, USA: Morgan Kaufmann, 1999, p. 657–664.

VÁZQUEZ-OJEDA, M; SEGOVIA-HERNÁNDEZ, J. G.; HERNÁNDEZ, S.; HERNÁNDEZ-AGUIRRE, A.; KISS, A. A. Optimization of an ethanol dehydration process using differential evolution algorithm. **Computer Aided Chemical Engineering**, v. 32, p. 217–222, 2013.

WALPOLE, R. E.; MYERS, R. H.; MYERS, S. L.; YE, K. **Probability and Statistics for Engineers and Scientists**. 9th ed. USA: Prentice Hall, 2011.

WANG, Y.; CAI, Z.; ZHANG, Q. Differential evolution with composite trial vector generation strategies and control parameters. **IEEE Transactions on Evolutionary Computation**, v. 15, n. 1, p. 55-66, 2011.

ZHANG, J.; SANDERSON, A. C. JADE: adaptive differential evolution with optional external archive. **IEEE Transactions on Evolutionary Computation**, v. 13, n. 5, p. 945–958, 2009a.

ZHANG, J.; SANDERSON, A. C. **Adaptive differential evolution - A robust approach to multimodal problem optimization**. Berlin, Germany: Springer, 2009b.

## APÊNDICES

### APÊNDICE A - GRÁFICOS DE CONVERGÊNCIA PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2005

Neste apêndice é apresentado os gráficos de convergência dos onze algoritmos avaliados (JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1, HSADE-v2) para cada uma das 25 funções propostas no CEC2005.

Nestes gráficos pode-se observar a evolução do valor médio do erro da função objetivo em função do número de avaliações da função objetivo. Para uma melhor visualização, um gráfico semi-log foi utilizado, ou seja, a escala logarítmica é aplicada ao valores médios do erro.

As FIGURAS 67, 68, 69 e 70 ilustram o gráfico de convergência para o caso  $D = 10$ .

As FIGURAS 71, 72, 73 e 74 ilustram o gráfico de convergência para o caso  $D = 30$ .

As FIGURAS 75, 76, 77 e 78 ilustram o gráfico de convergência para o caso  $D = 50$ .

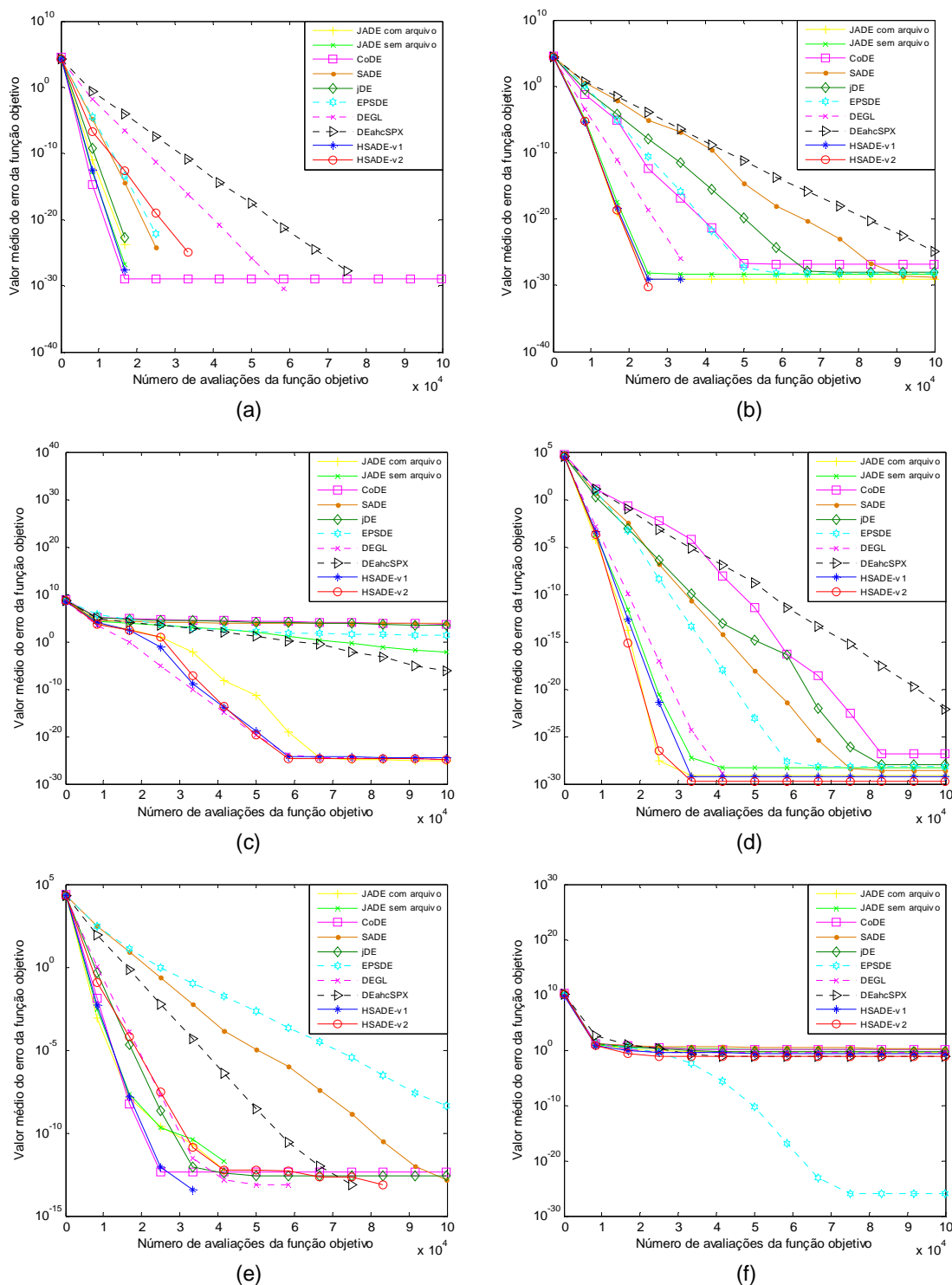


FIGURA 67: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 10$ .

FONTE: O autor (2014)

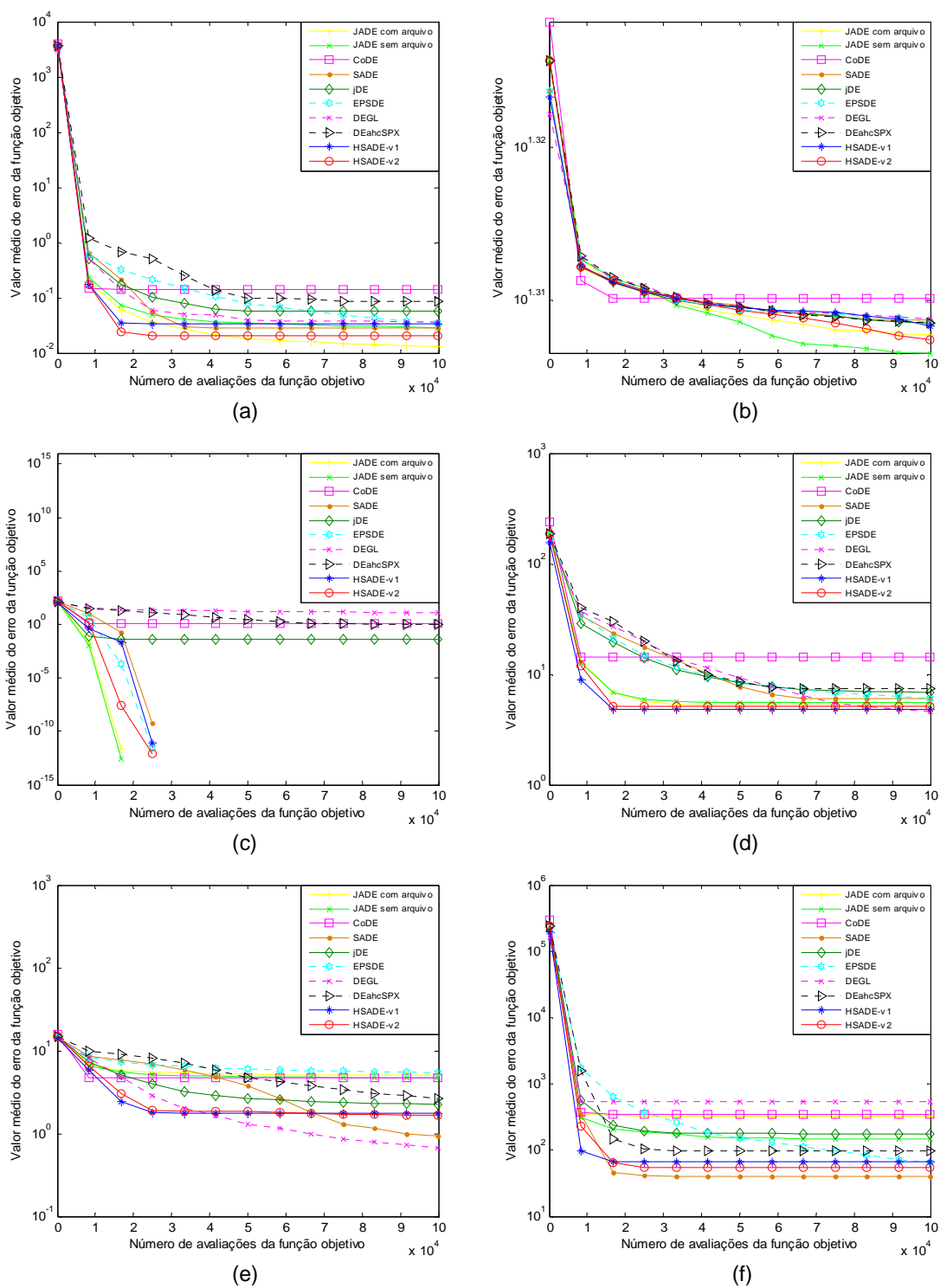


FIGURA 68: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADe-v1 e HSADe-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

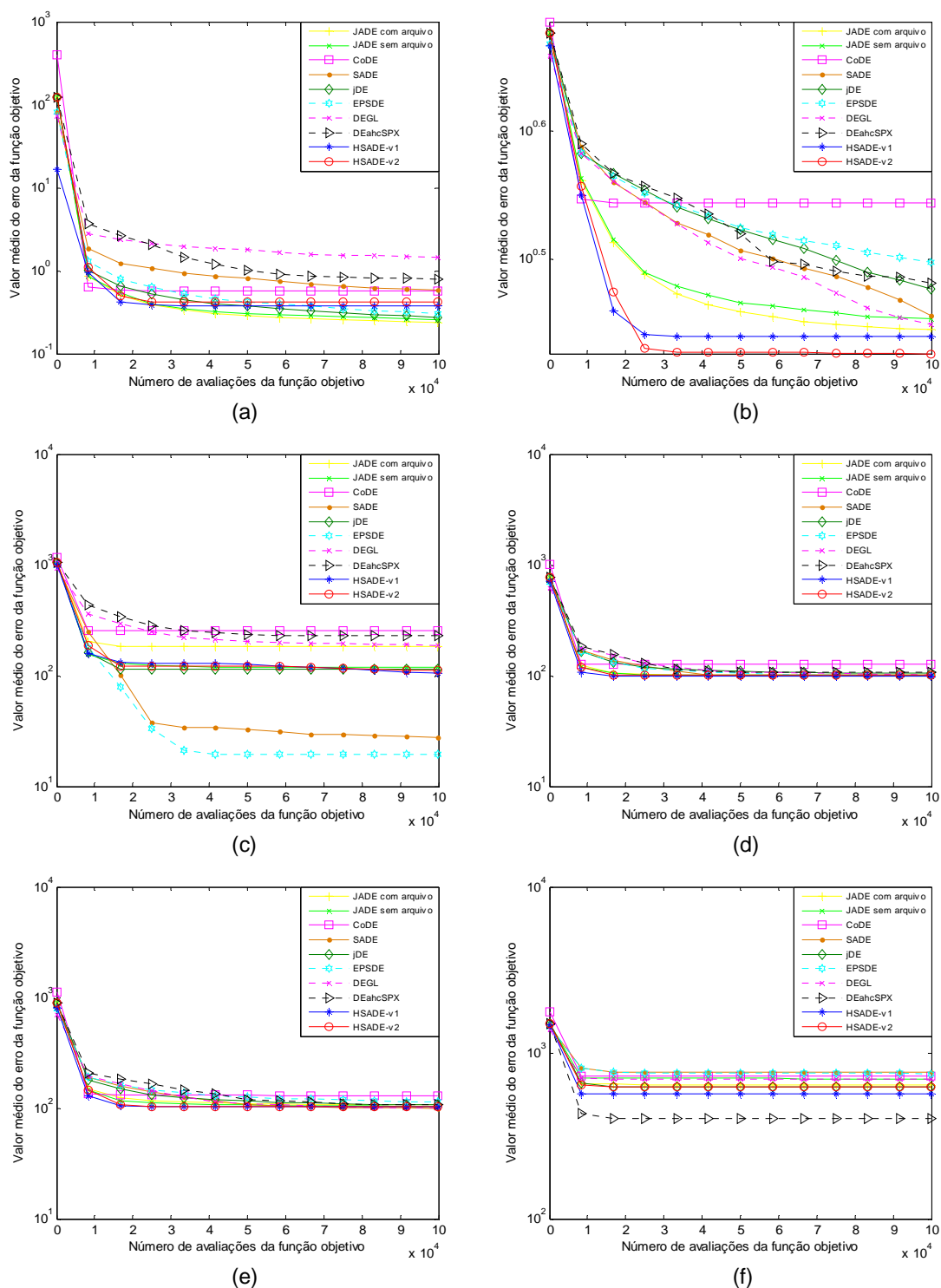


FIGURA 69: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 10$ .

FONTE: O autor (2014)



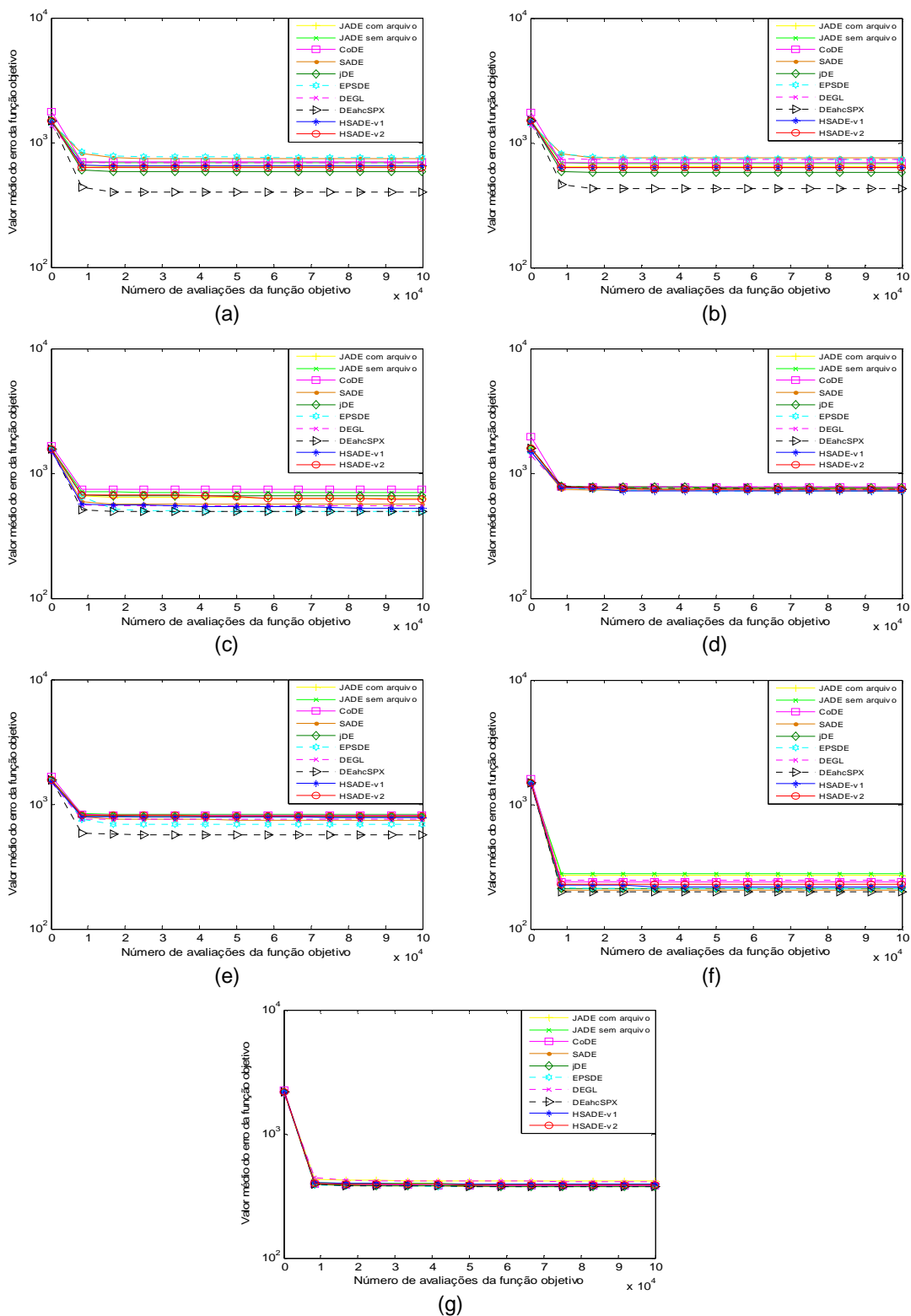


FIGURA 70: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$ , (g)  $F_{25}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

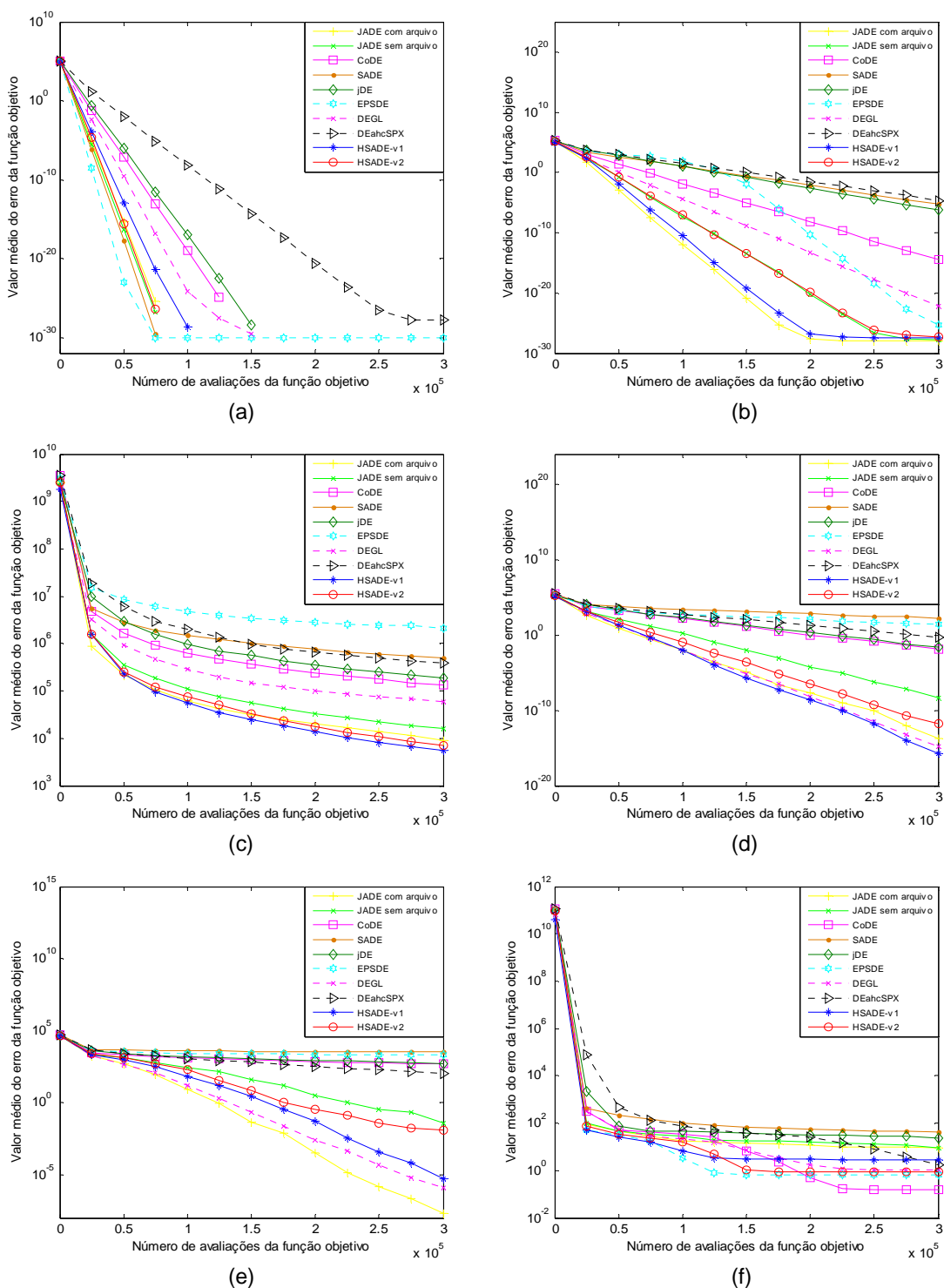


FIGURA 71: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 30$ .

FONTE: O autor (2014)

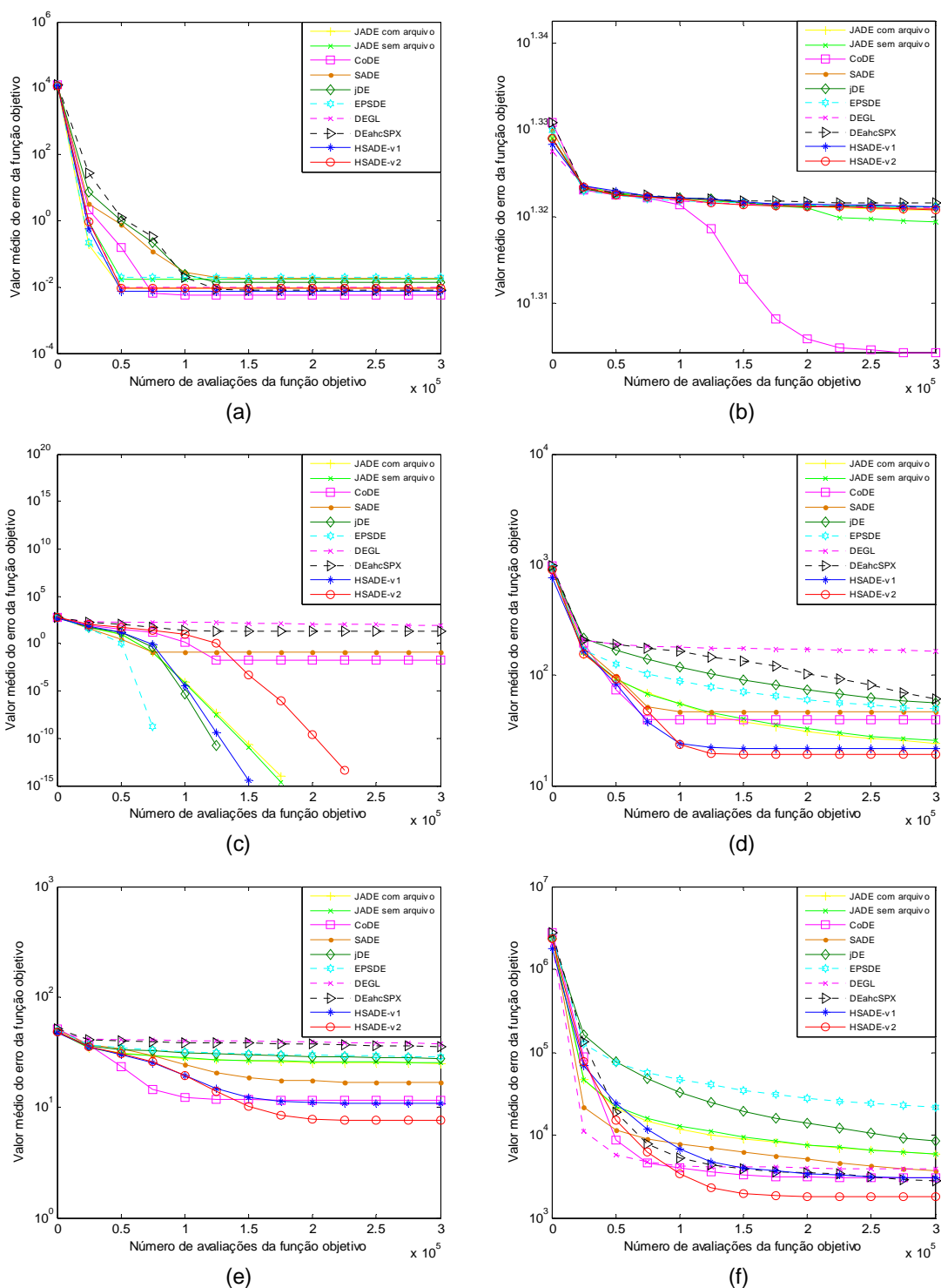


FIGURA 72: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

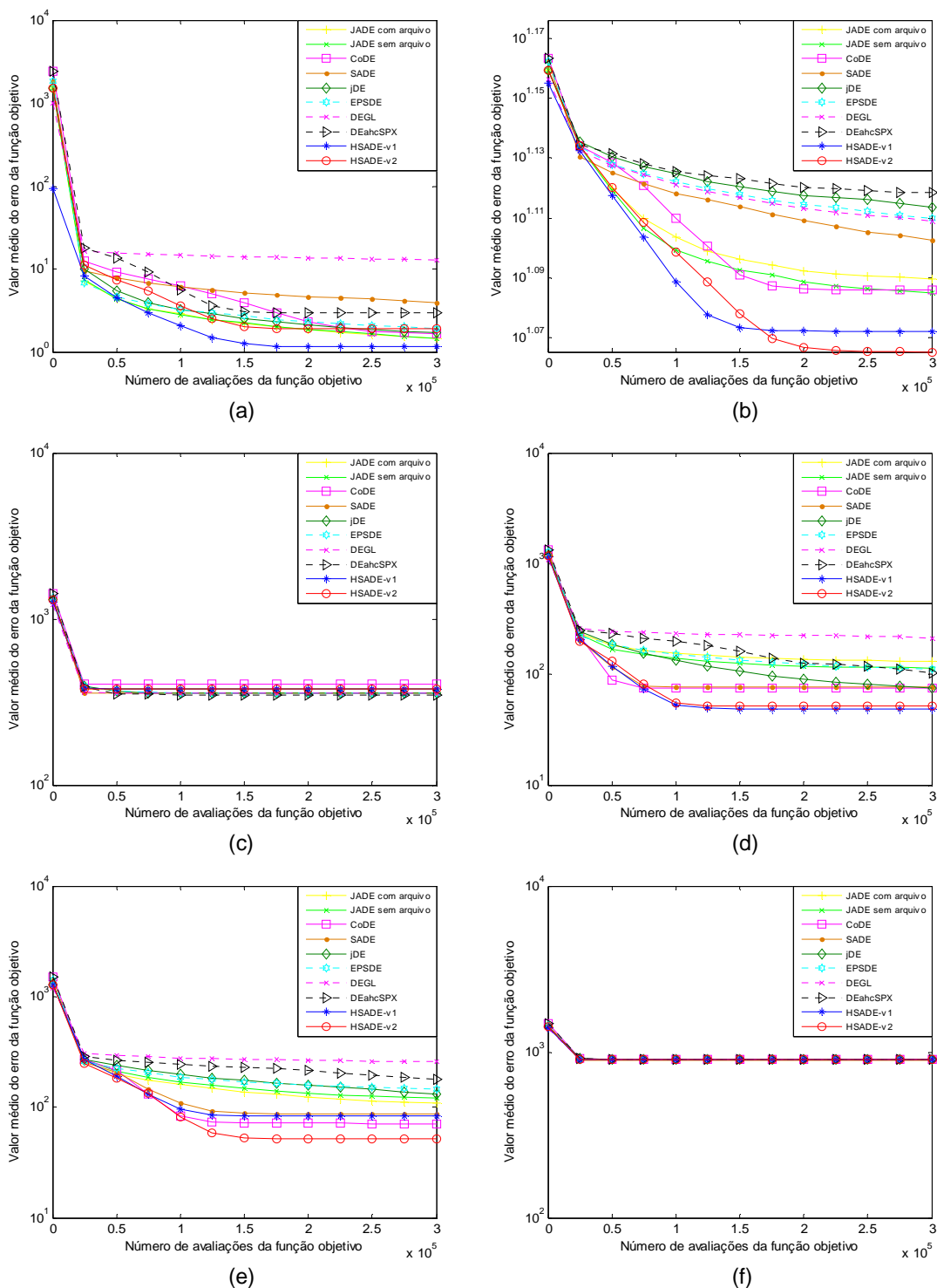


FIGURA 73: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

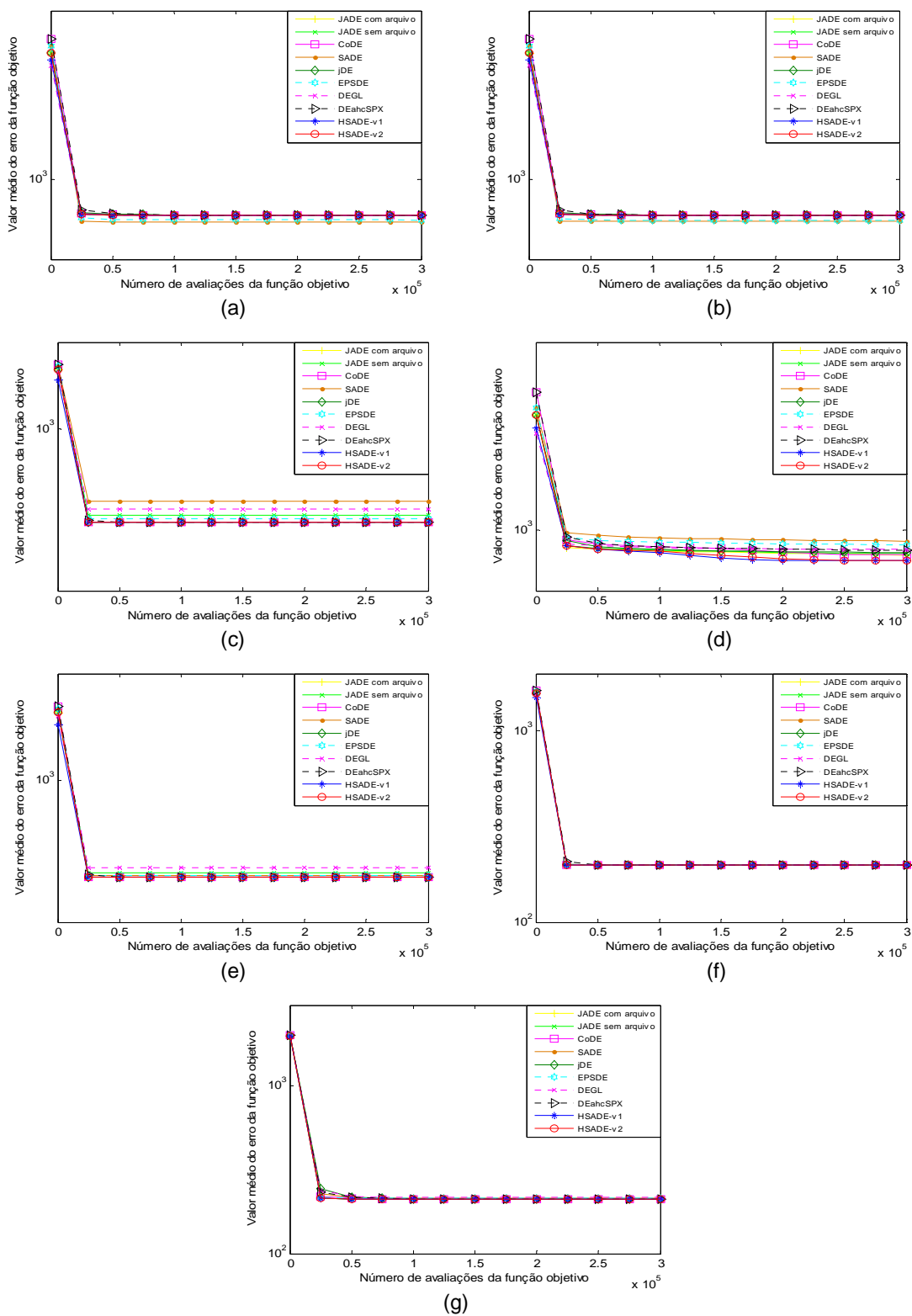


FIGURA 74: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$ , (g)  $F_{25}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

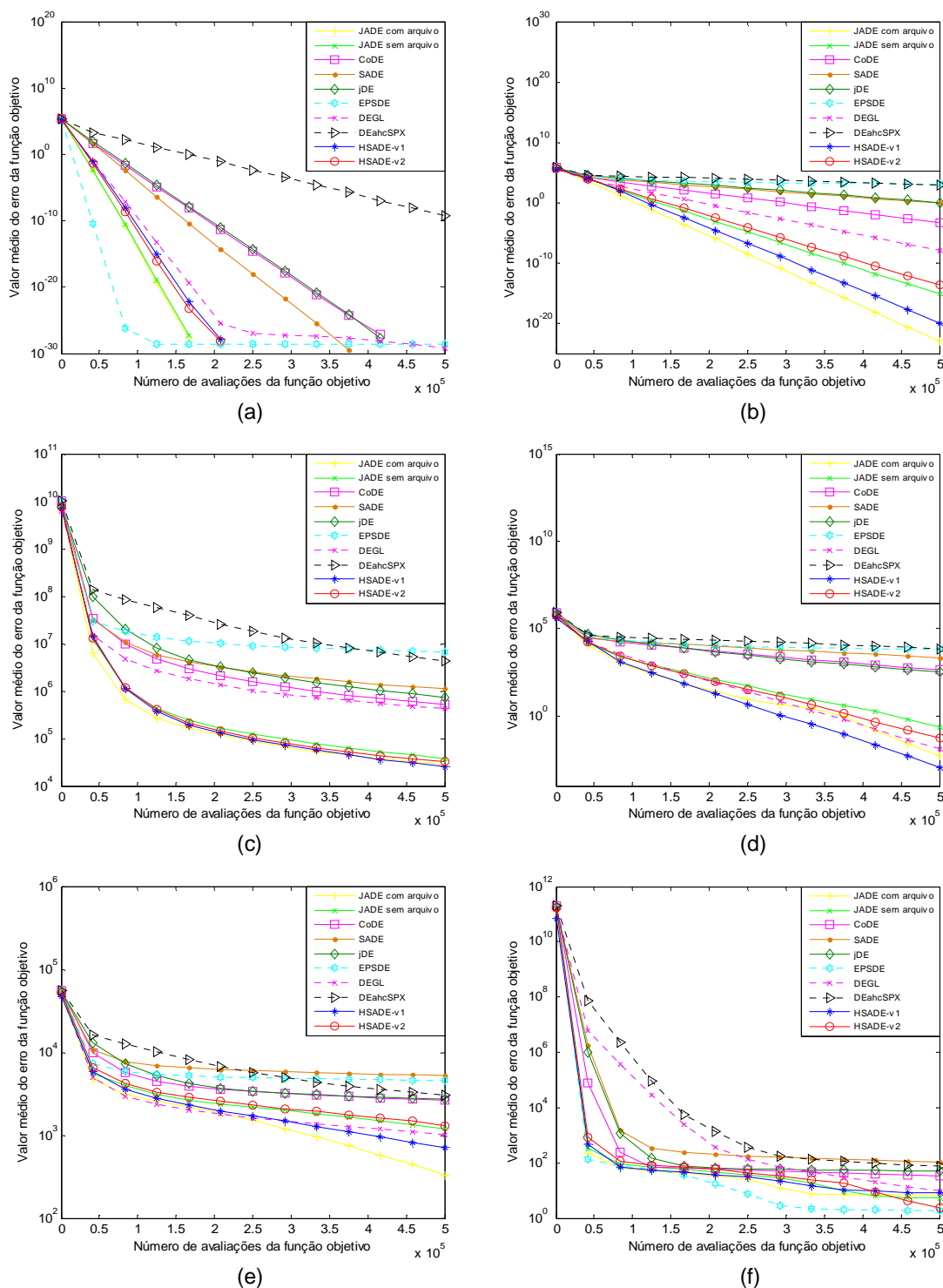


FIGURA 75: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 50$ .

FONTE: O autor (2014)

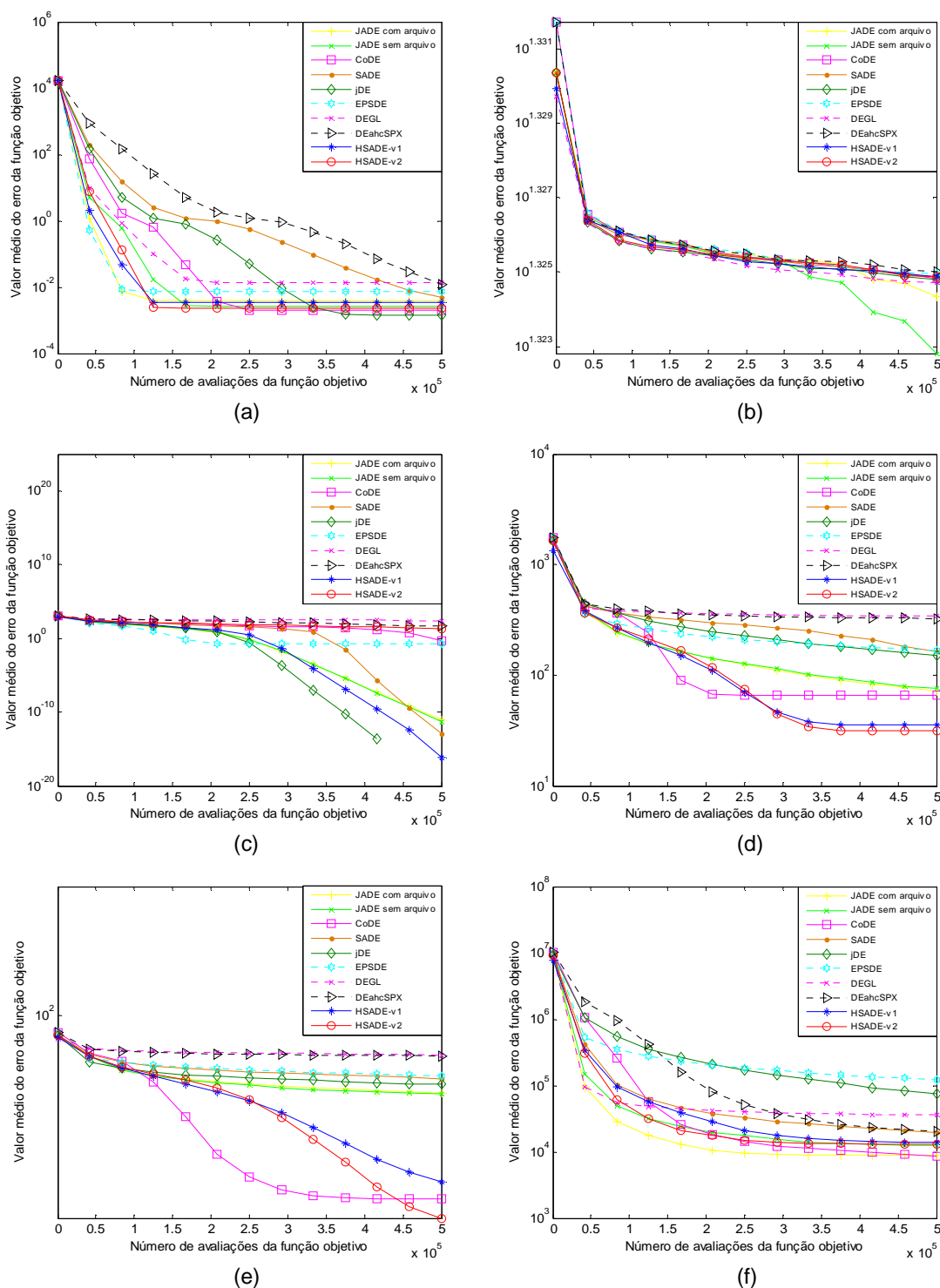


FIGURA 76: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 50$ .

FONTE: O autor (2014)

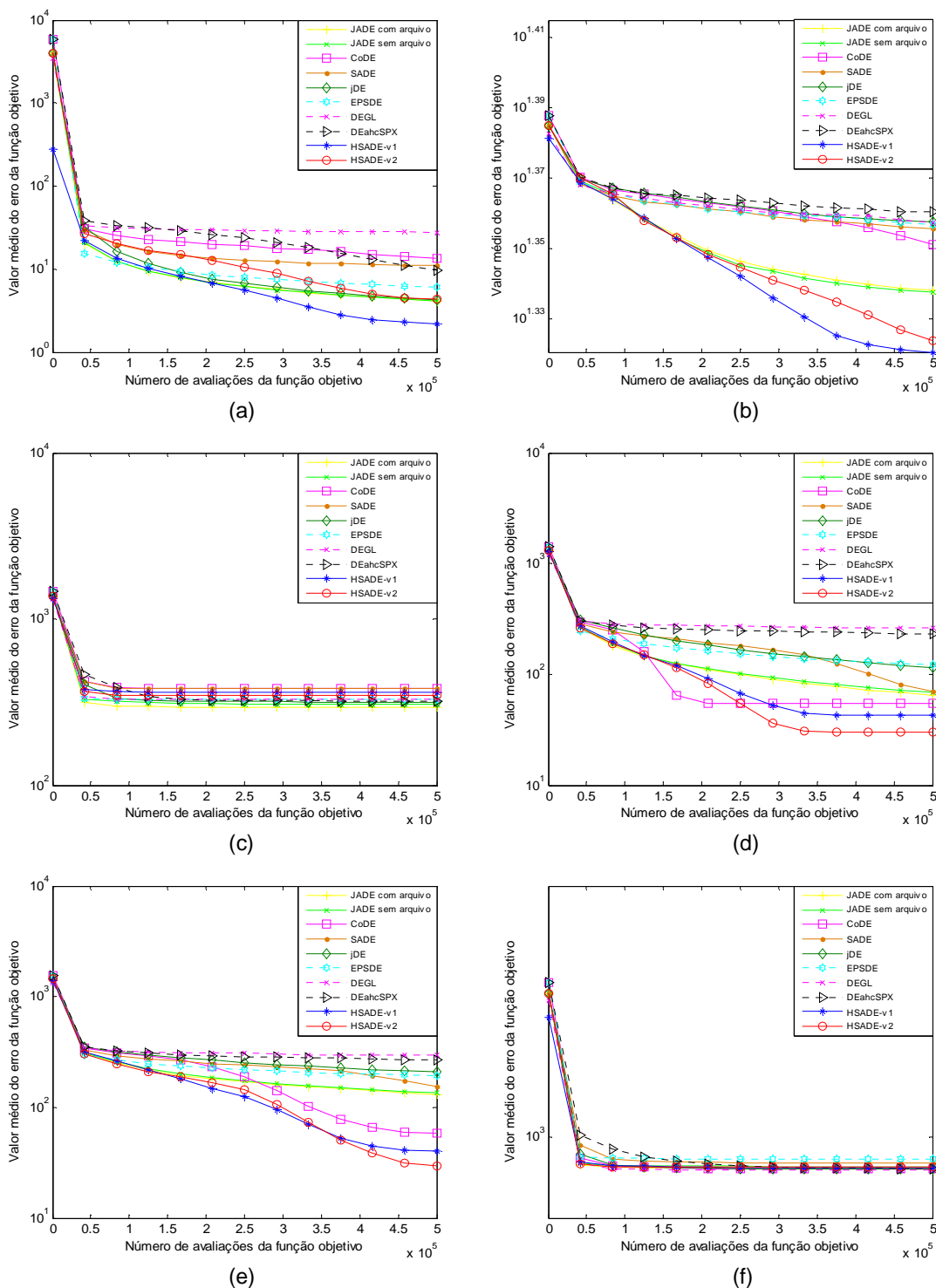


FIGURA 77: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 50$ .

FONTE: O autor (2014)



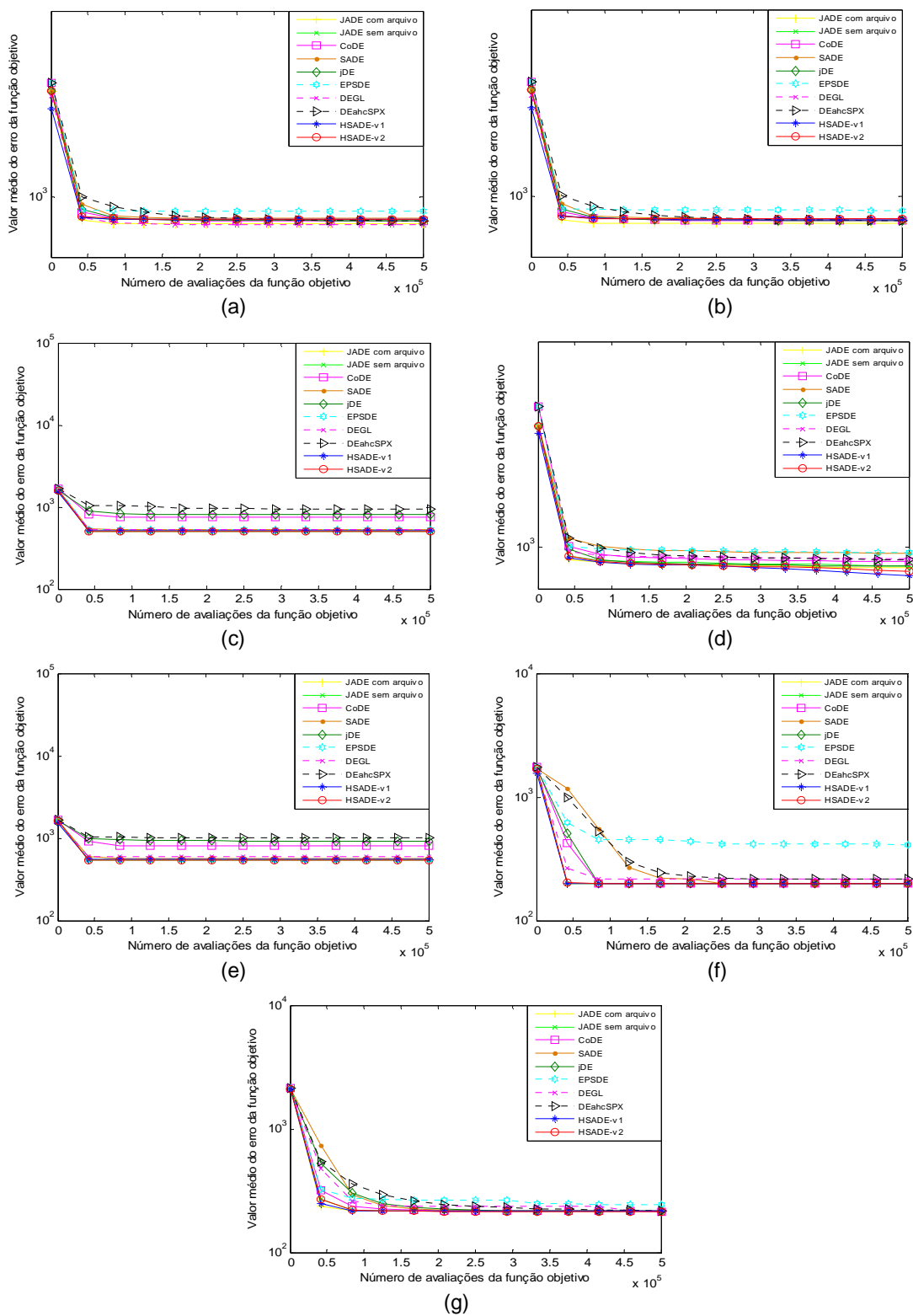


FIGURA 78: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$ , (g)  $F_{25}$  para o caso  $D = 50$ .

FONTE: O autor (2014)

## **APÊNDICE B - GRÁFICOS DE CONVERGÊNCIA PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2011**

Neste apêndice é apresentado os gráficos de convergência dos onze algoritmos avaliados (JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1, HSADE-v2) para cada um dos 22 problemas reais de otimização propostos no CEC2011.

Nestes gráficos pode-se observar a evolução do valor médio da função objetivo em função do número de avaliações da função objetivo. Para uma melhor visualização, um gráfico semi-log foi utilizado, ou seja, a escala logarítmica é aplicada ao valores médios da função objetivo.

As FIGURAS 79, 80, 81 e 82 ilustram o gráfico de convergência de cada um dos 22 problemas testes propostos no CEC2011.

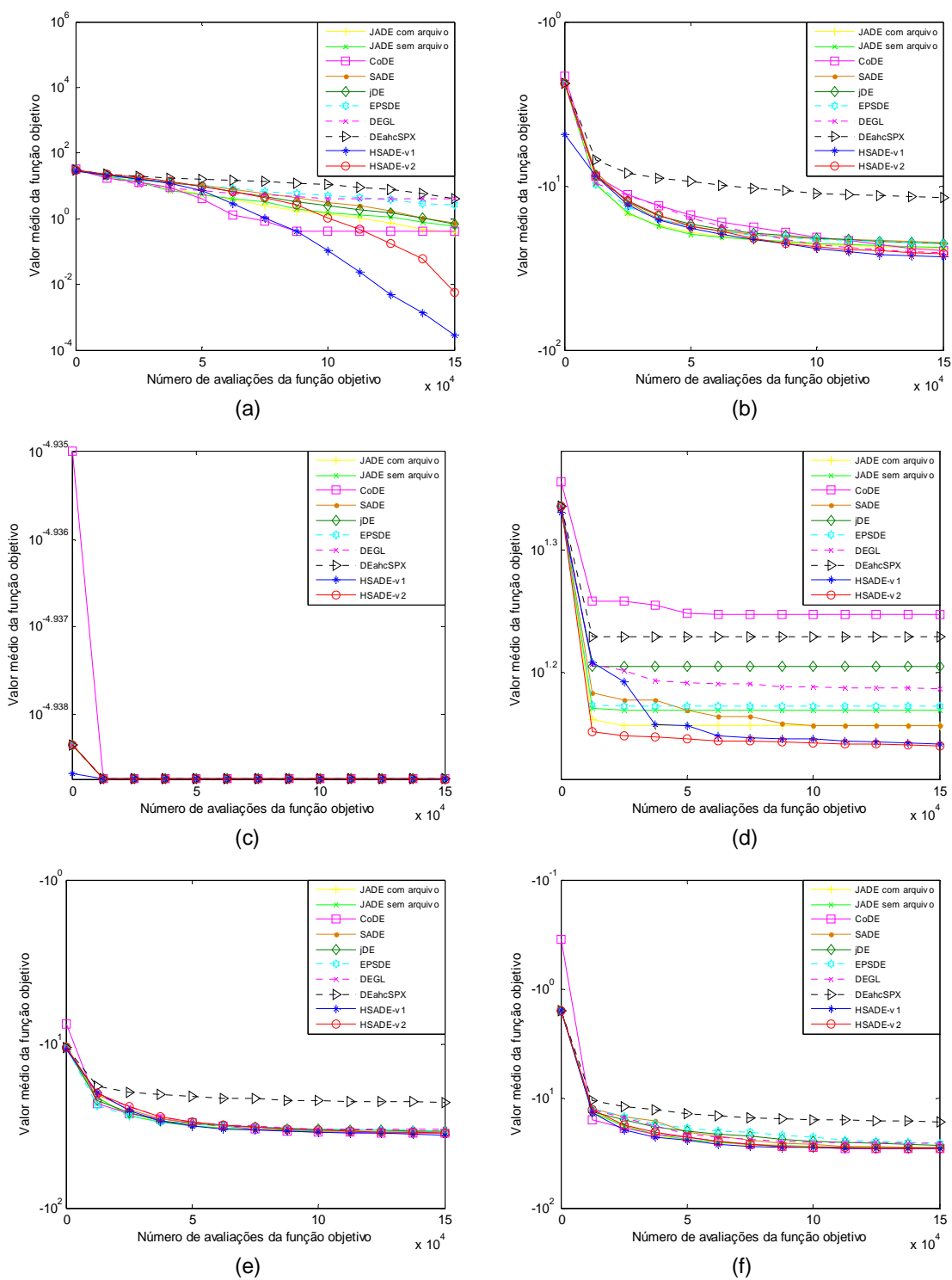


FIGURA 79: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, *EPpSDE*, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a)  $P_1$ , (b)  $P_2$ , (c)  $P_3$ , (d)  $P_4$ , (e)  $P_5$ , (f)  $P_6$ .

FONTE: O autor (2014)

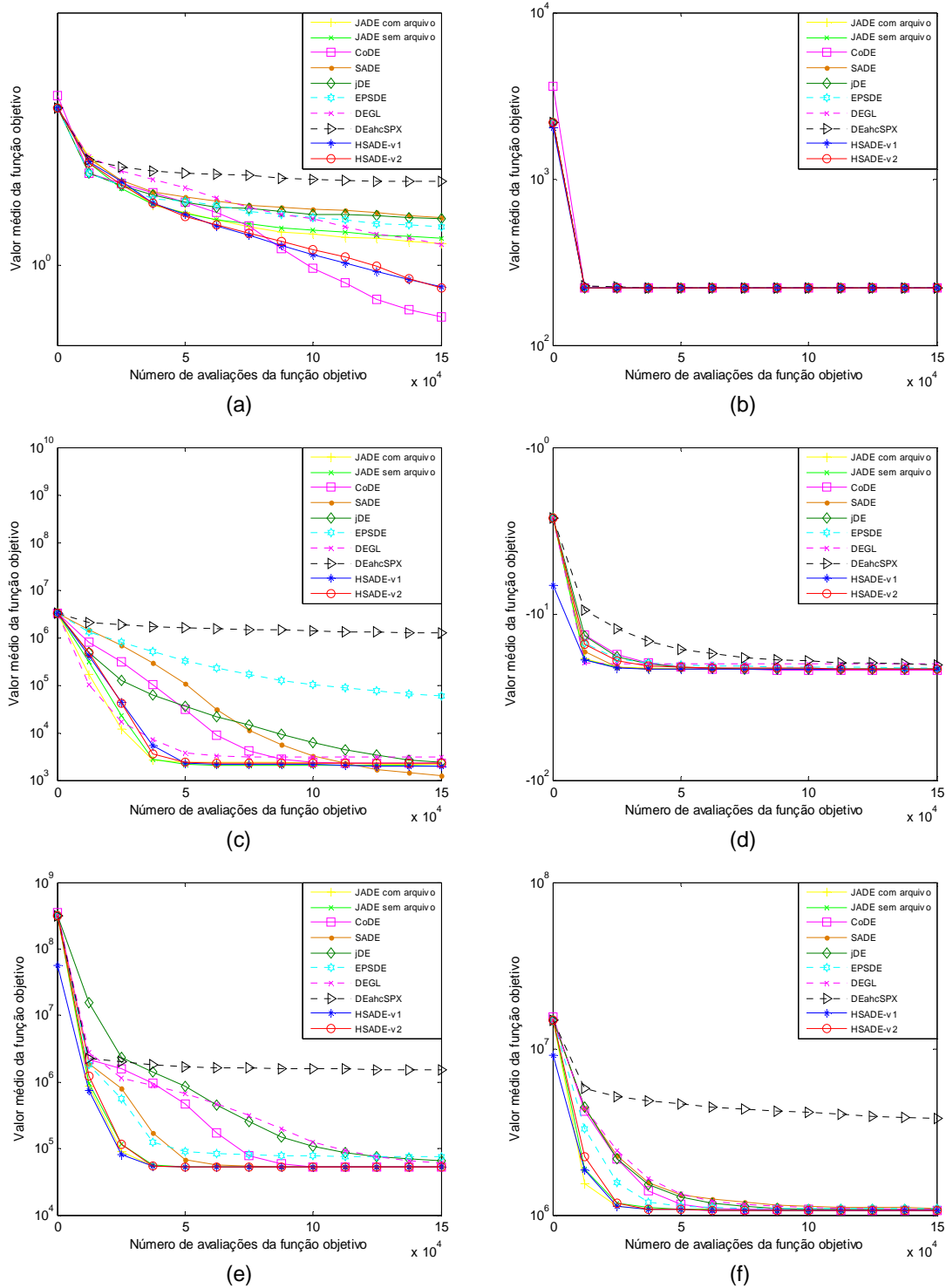


FIGURA 80: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a)  $P_7$ , (b)  $P_8$ , (c)  $P_9$ , (d)  $P_{10}$ , (e)  $P_{11}$ , (f)  $P_{12}$ .

FONTE: O autor (2014)

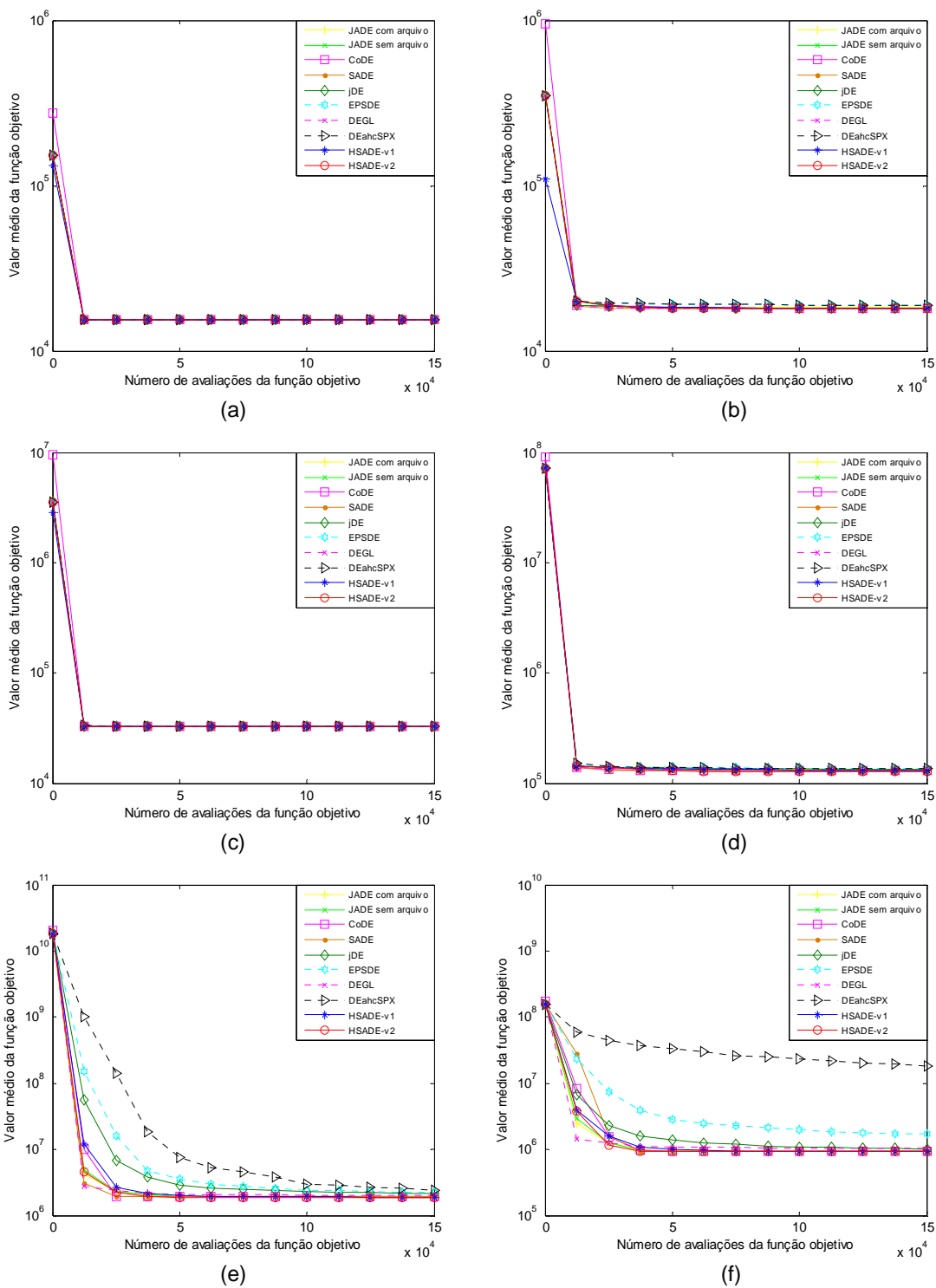


FIGURA 81: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para os problemas testes (a)  $P_{13}$ , (b)  $P_{14}$ , (c)  $P_{15}$ , (d)  $P_{16}$ , (e)  $P_{17}$ , (f)  $P_{18}$ .

FONTE: O autor (2014)

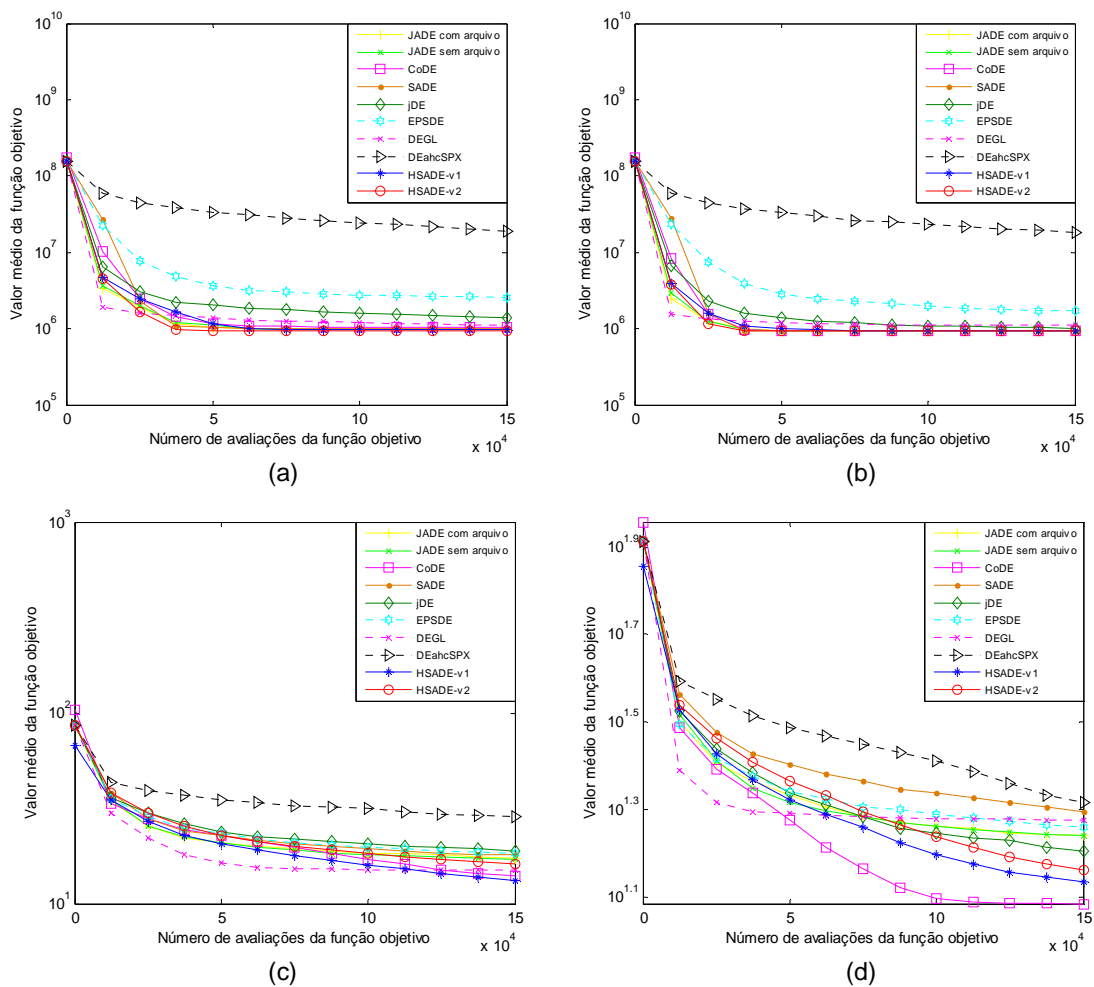


FIGURA 82: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADe-v1 e HSADe-v2 para os problemas testes (a)  $P_{19}$ , (b)  $P_{20}$ , (c)  $P_{21}$ , (d)  $P_{22}$ .  
 FONTE: O autor (2014)

## APÊNDICE C - GRÁFICOS DE CONVERGÊNCIA PARA AS FUNÇÕES TESTES PROPOSTAS NO CEC2011

Neste apêndice é apresentado os gráficos de convergência dos onze algoritmos avaliados (JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1, HSADE-v2) para cada uma das 28 funções propostas no CEC2013.

Nestes gráficos pode-se observar a evolução do valor médio do erro da função objetivo em função do número de avaliações da função objetivo. Para uma melhor visualização, um gráfico semi-log foi utilizado, ou seja, a escala logarítmica é aplicada ao valores médios do erro.

As FIGURAS 83, 84, 85, 86 e 87 ilustram o gráfico de convergência para o caso  $D = 10$ .

As FIGURAS 88, 89, 90, 91 e 92 ilustram o gráfico de convergência para o caso  $D = 30$ .

As FIGURAS 93, 94, 95, 96 e 97 ilustram o gráfico de convergência para o caso  $D = 50$ .

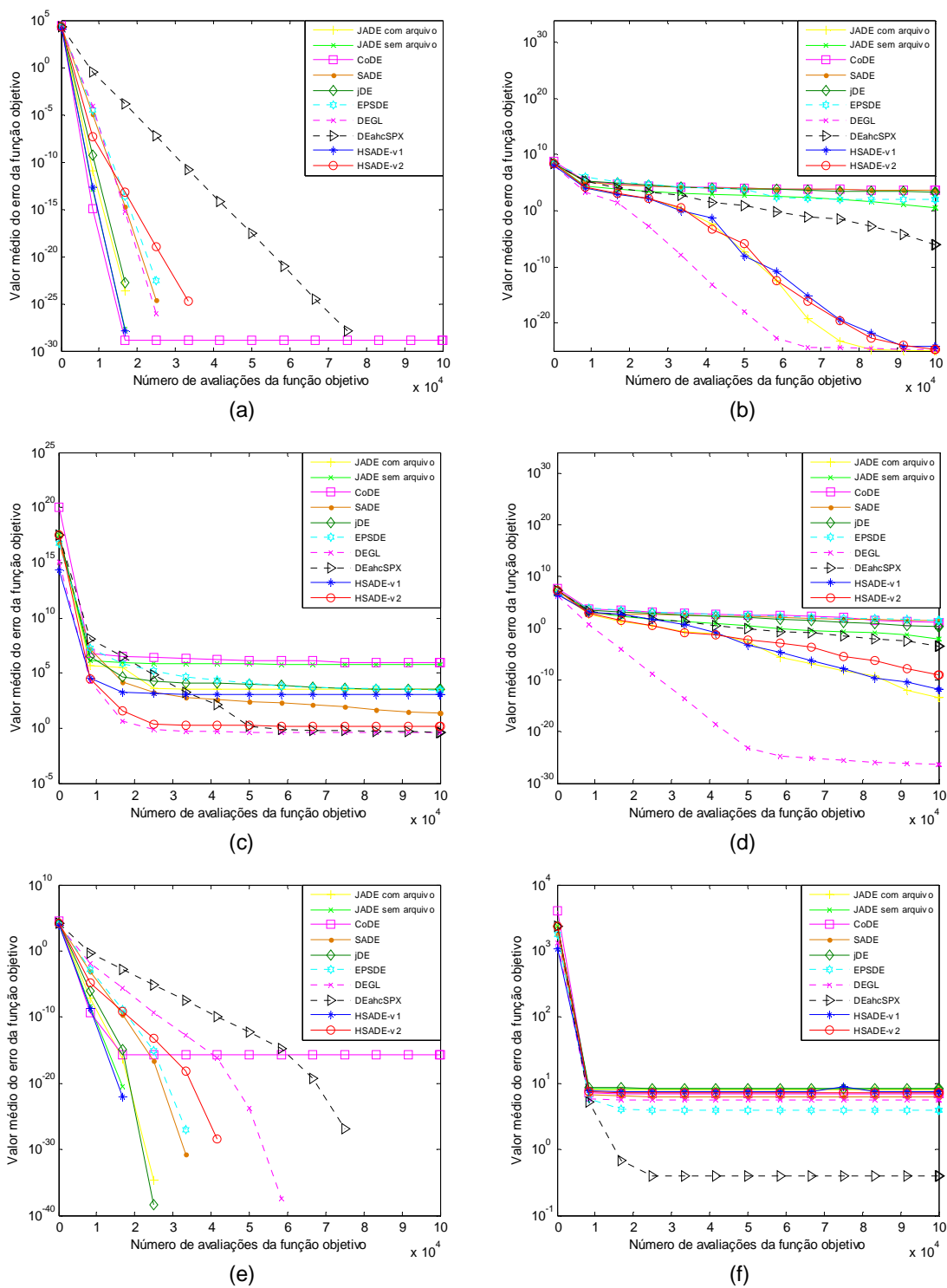


FIGURA 83: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 10$ .

FONTE: O autor (2014)



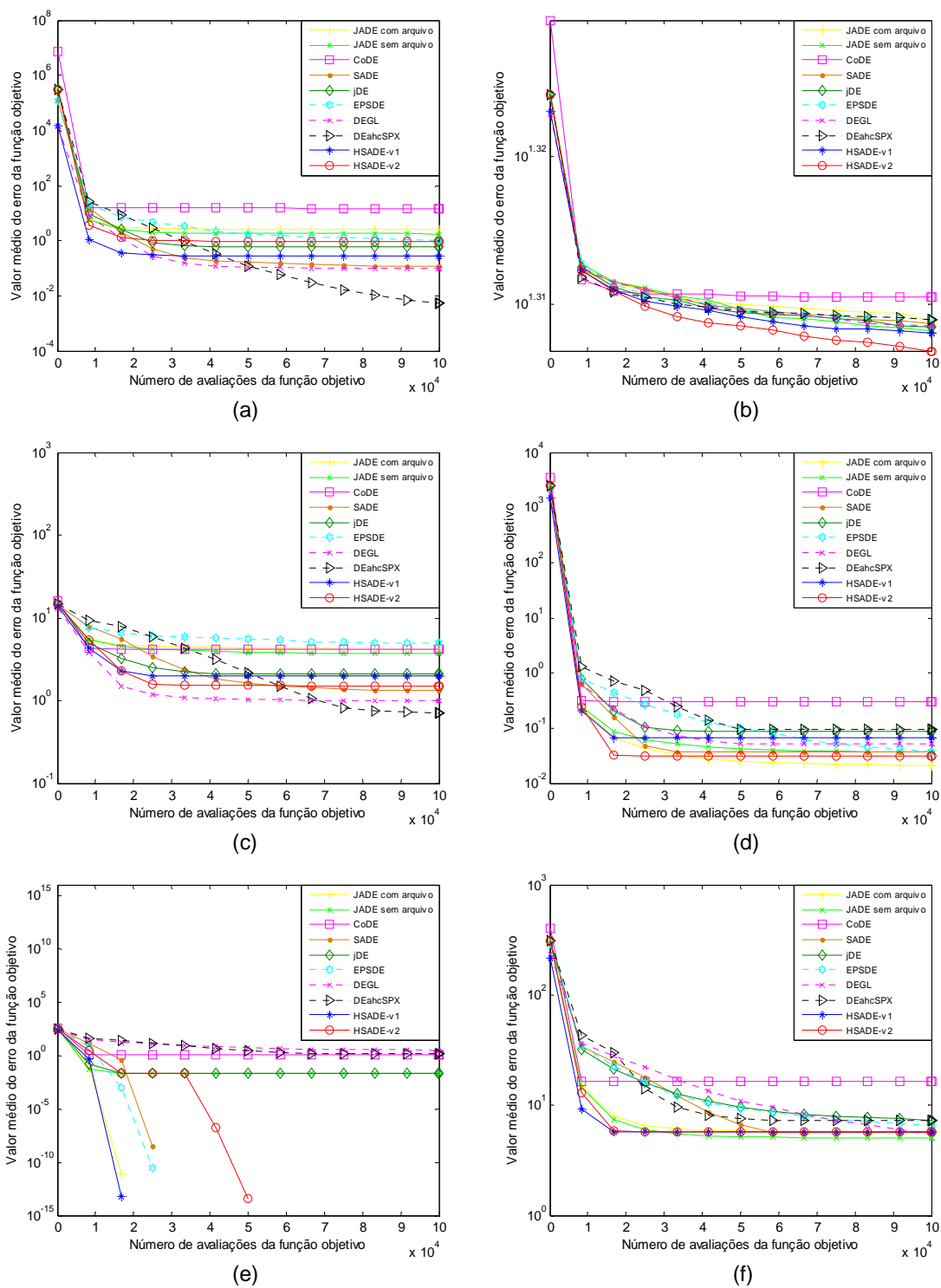


FIGURA 84: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

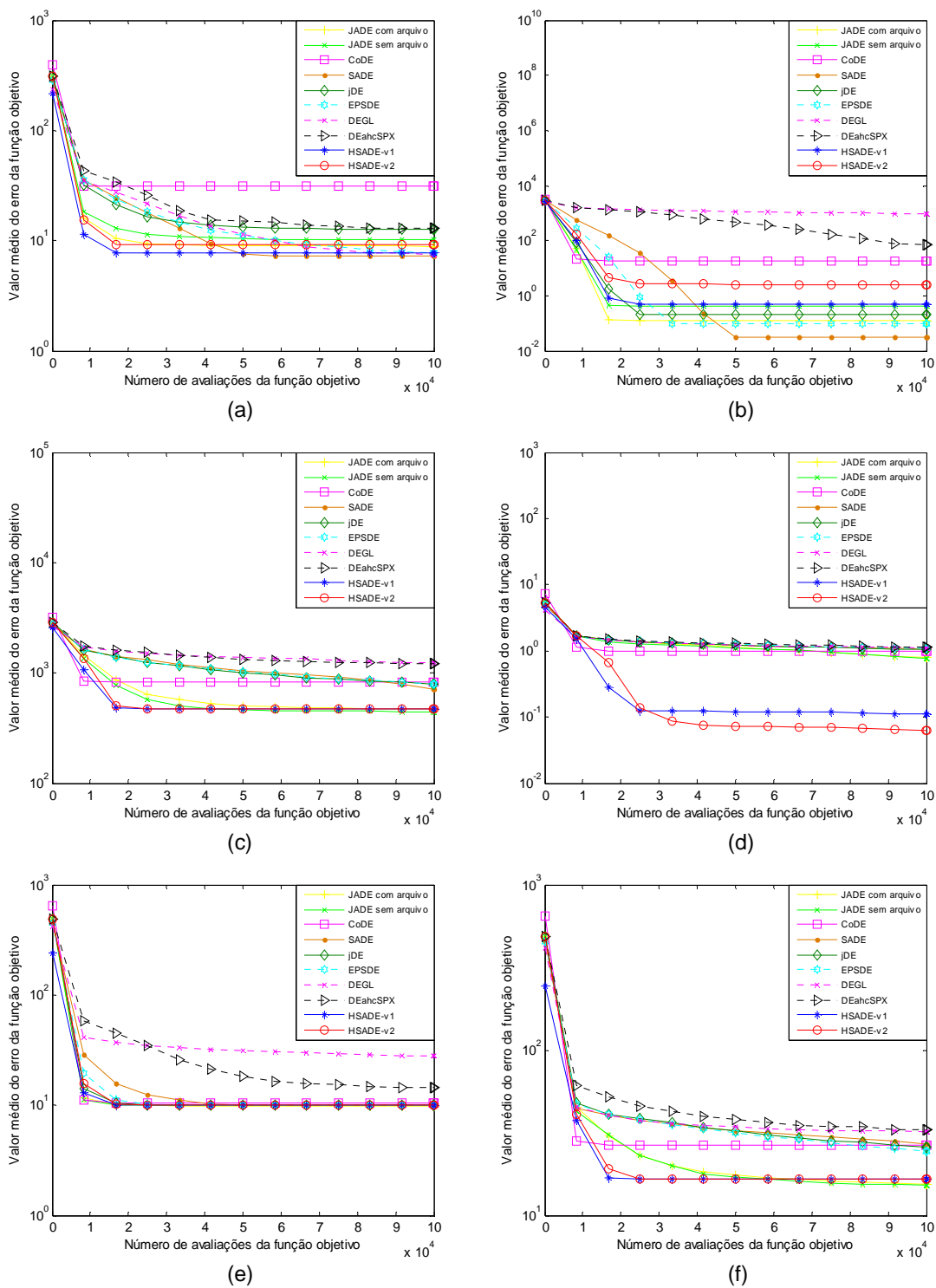


FIGURA 85: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

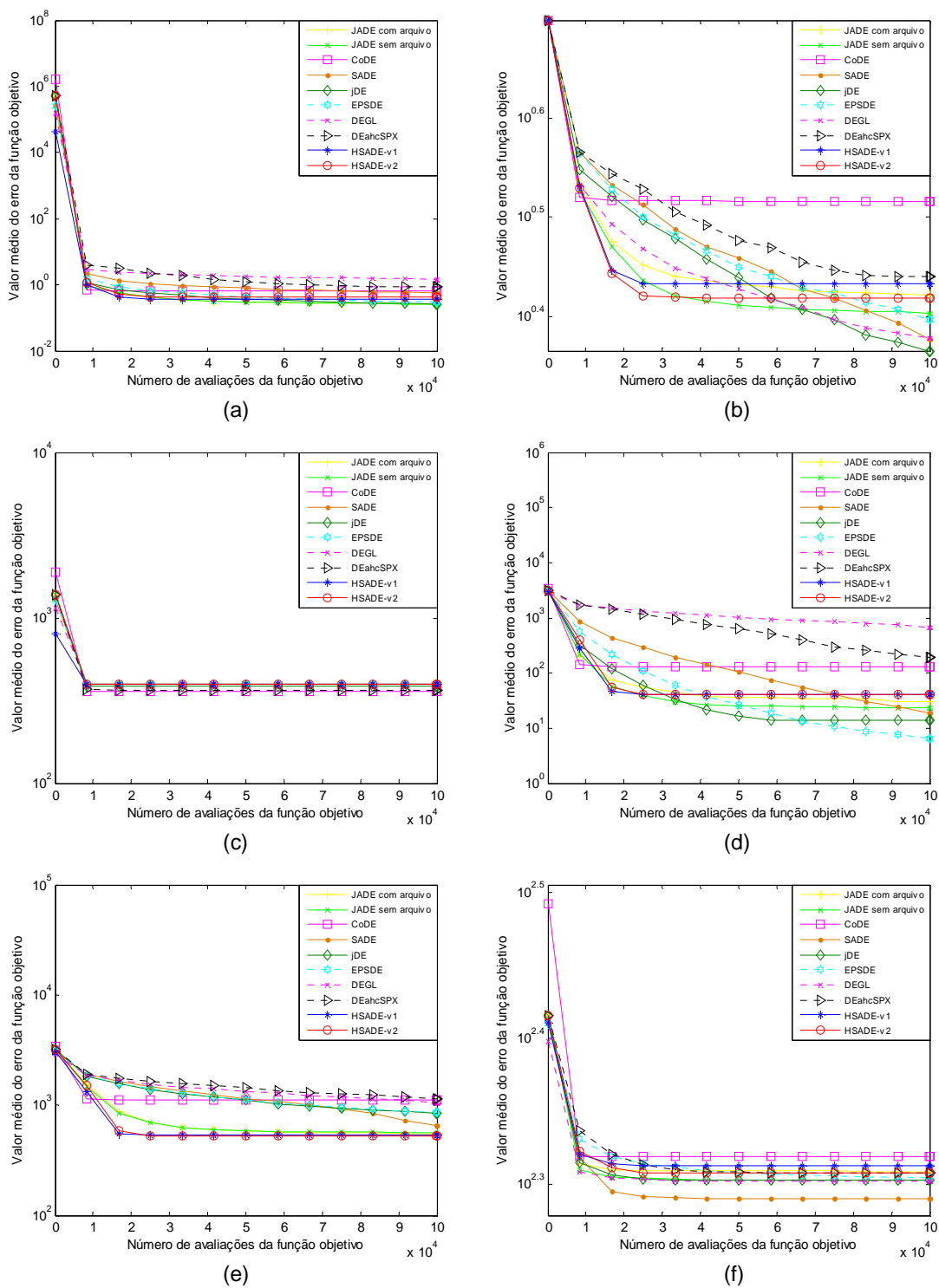


FIGURA 86: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

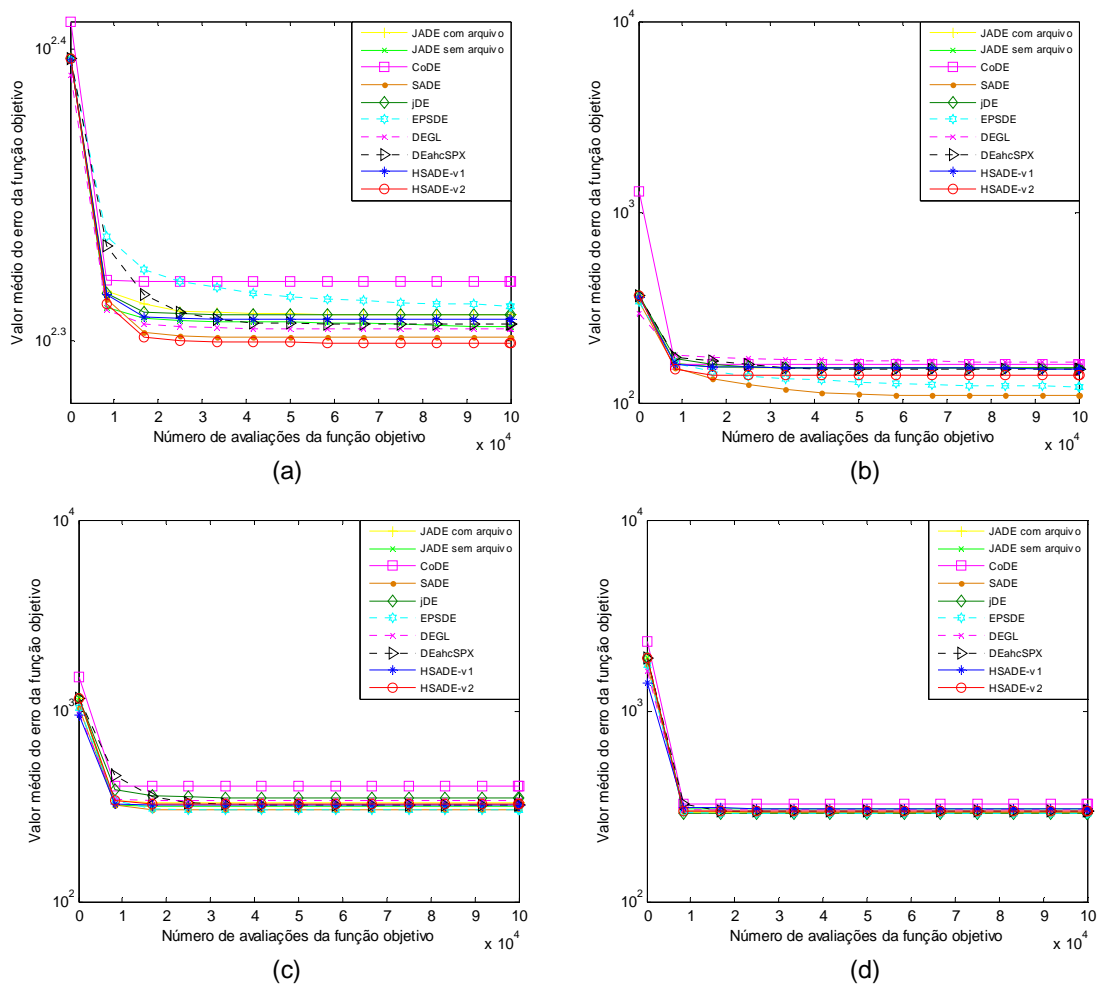


FIGURA 87: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{25}$ , (b)  $F_{26}$ , (c)  $F_{27}$ , (d)  $F_{28}$  para o caso  $D = 10$ .

FONTE: O autor (2014)

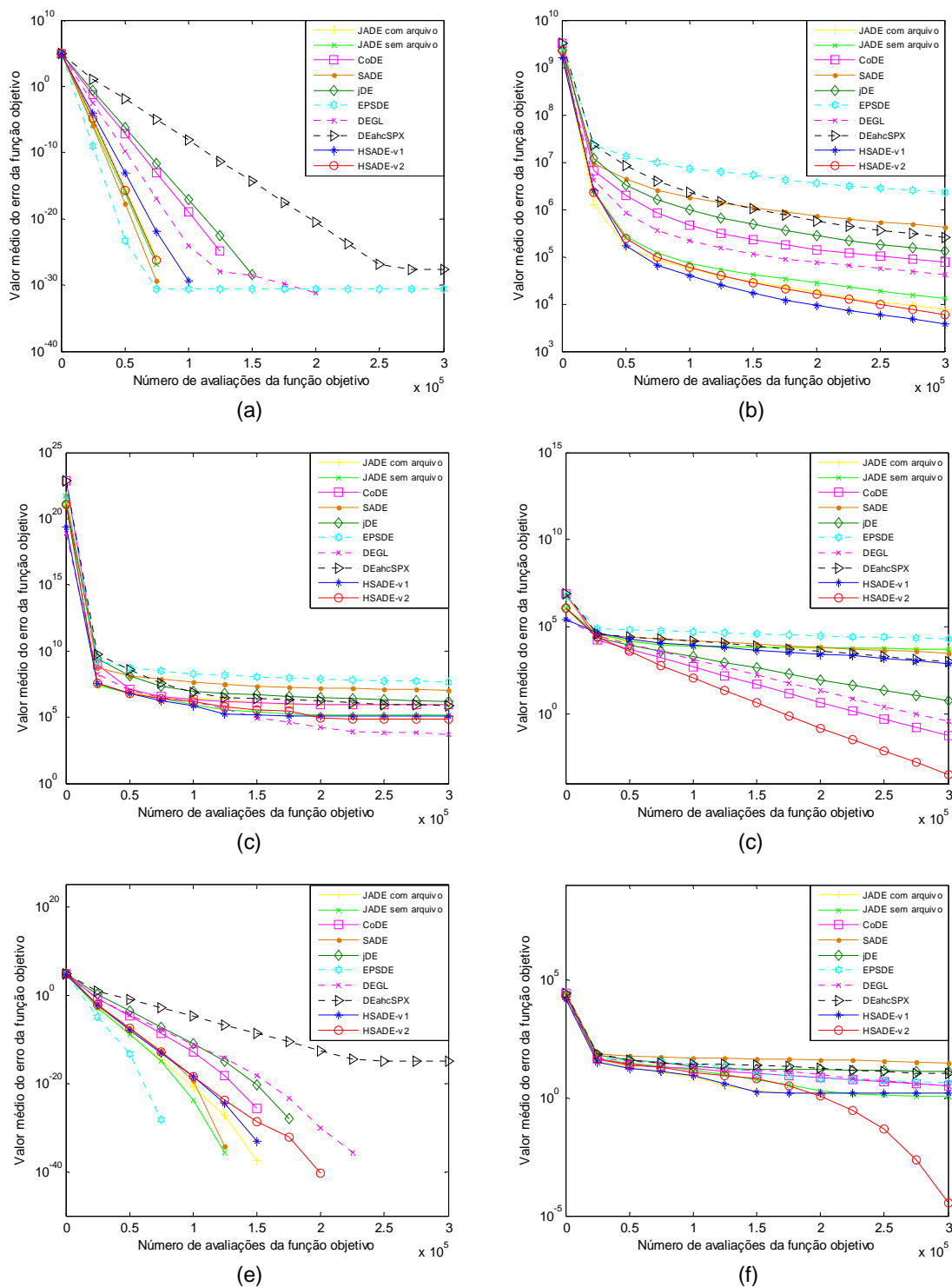


FIGURA 88: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 30$ .

FONTE: O autor (2014)

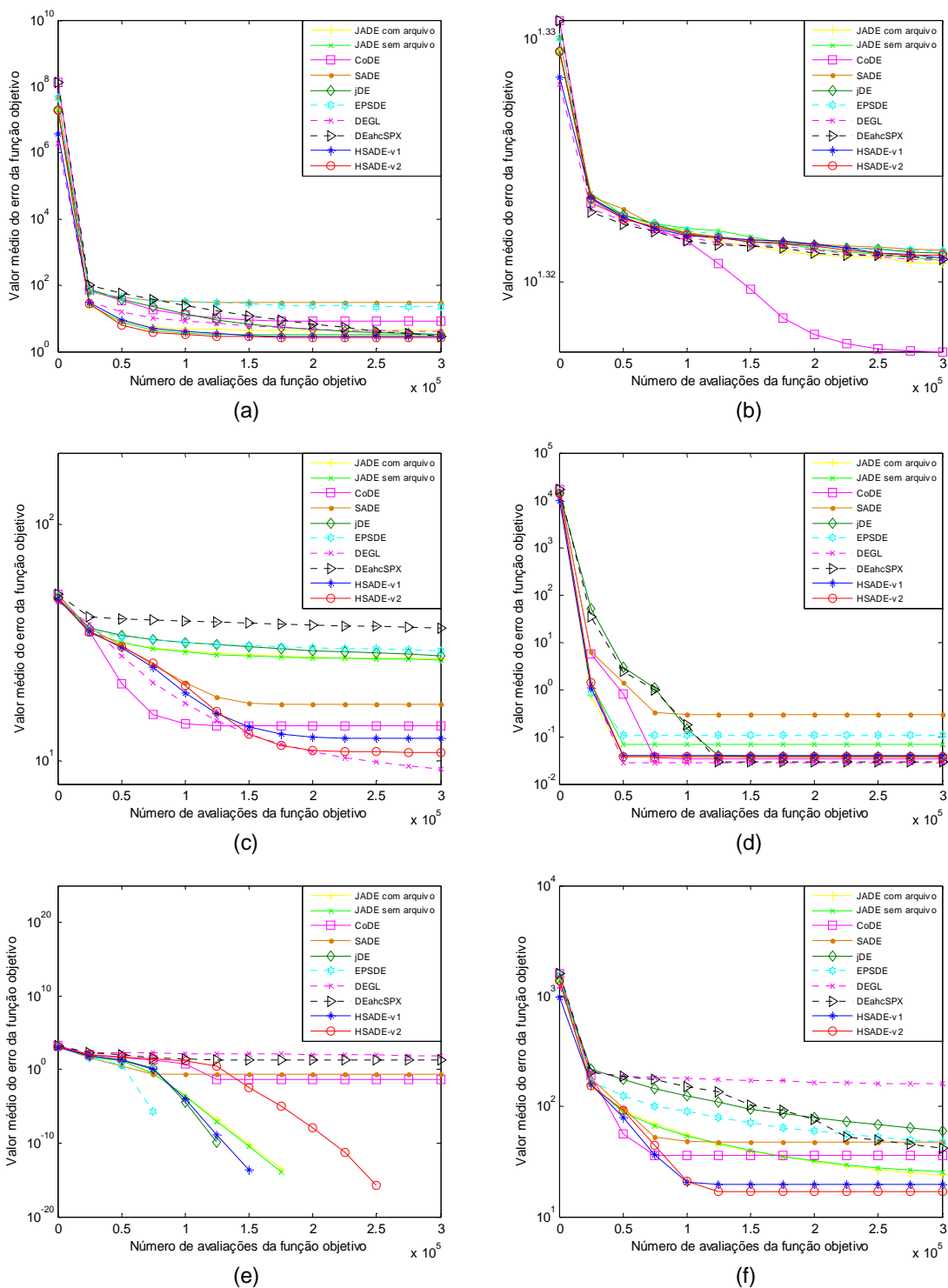


FIGURA 89: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

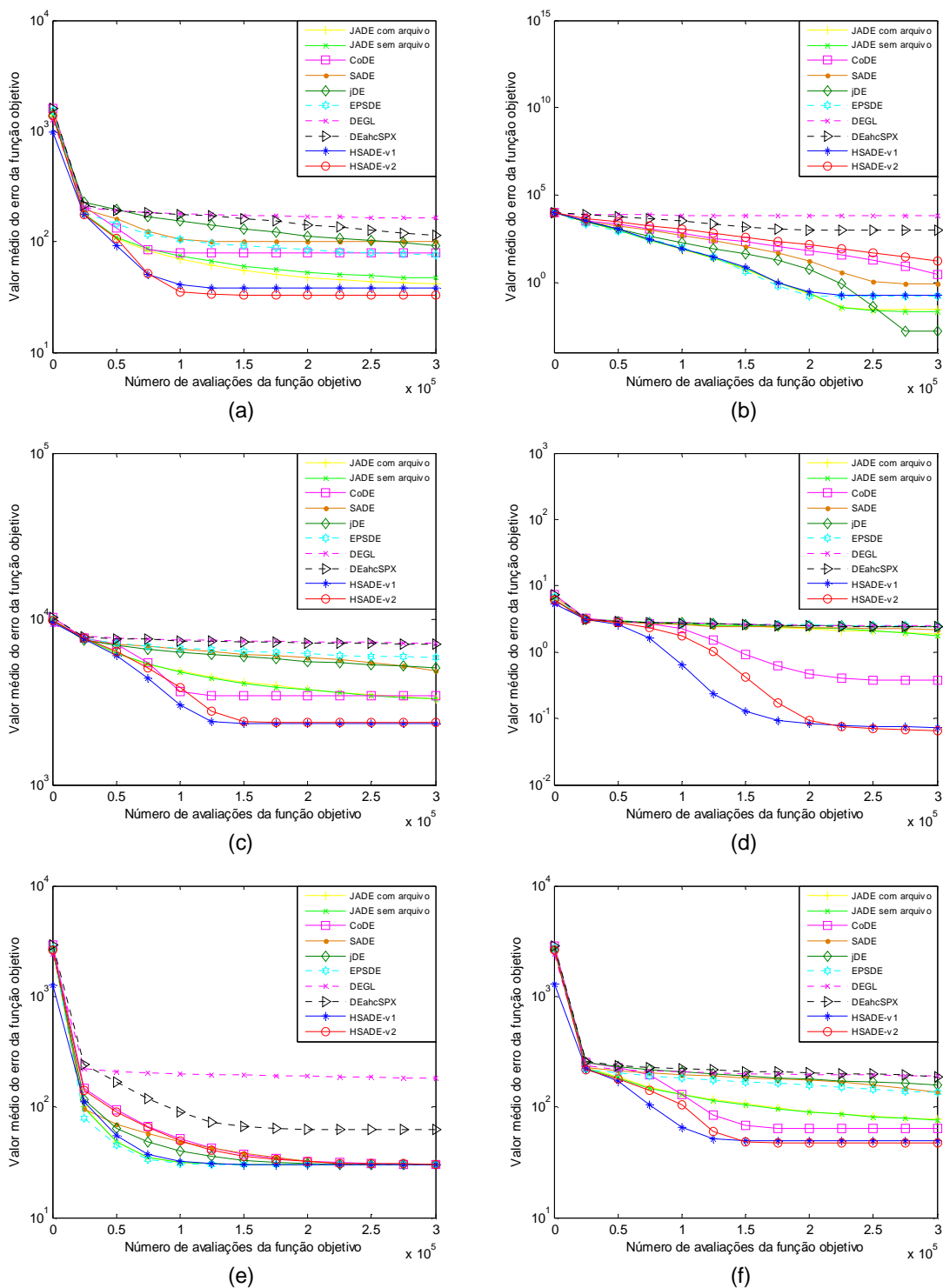


FIGURA 90: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

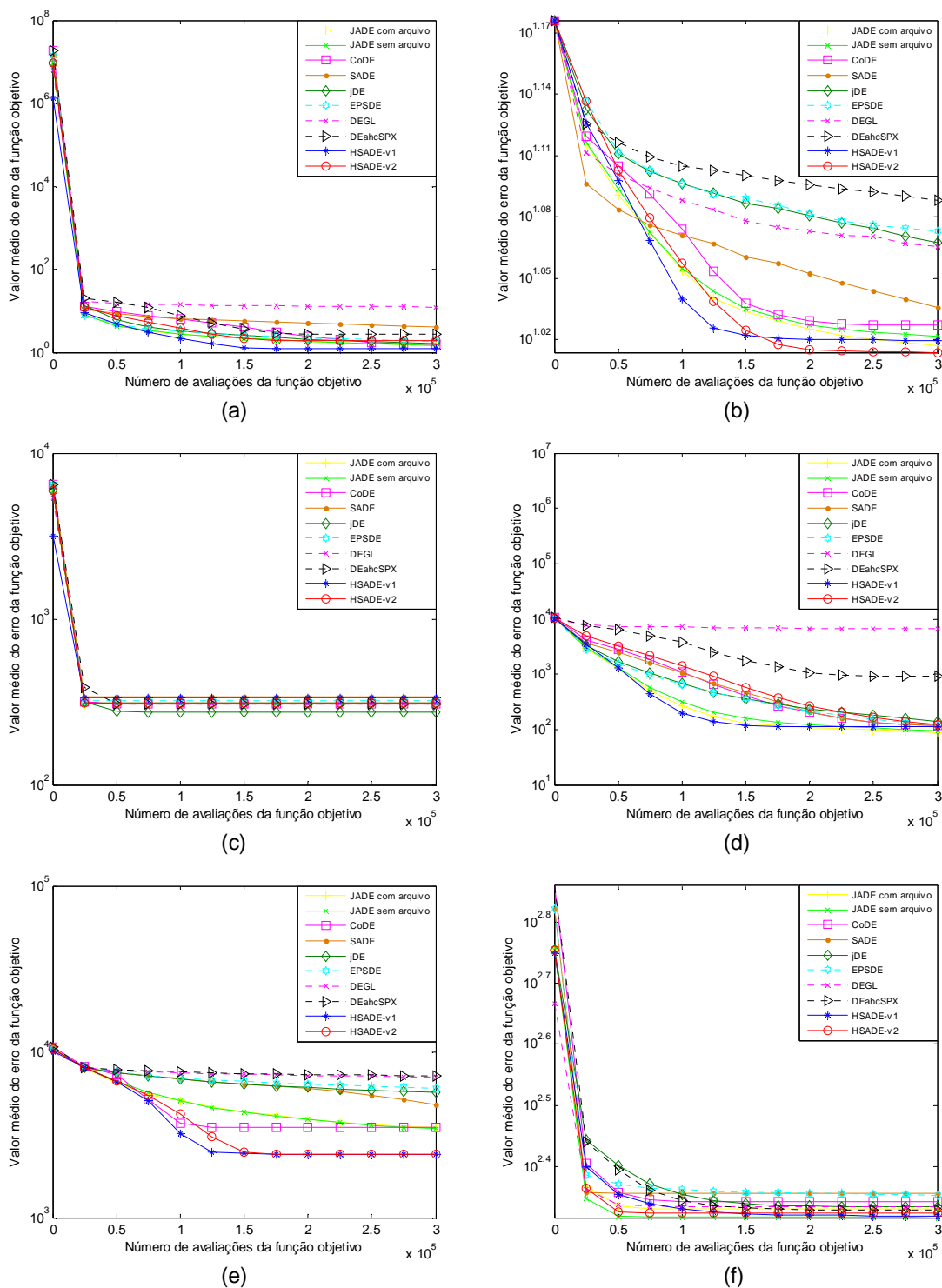


FIGURA 91: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$  para o caso  $D = 30$ .

FONTE: O autor (2014)



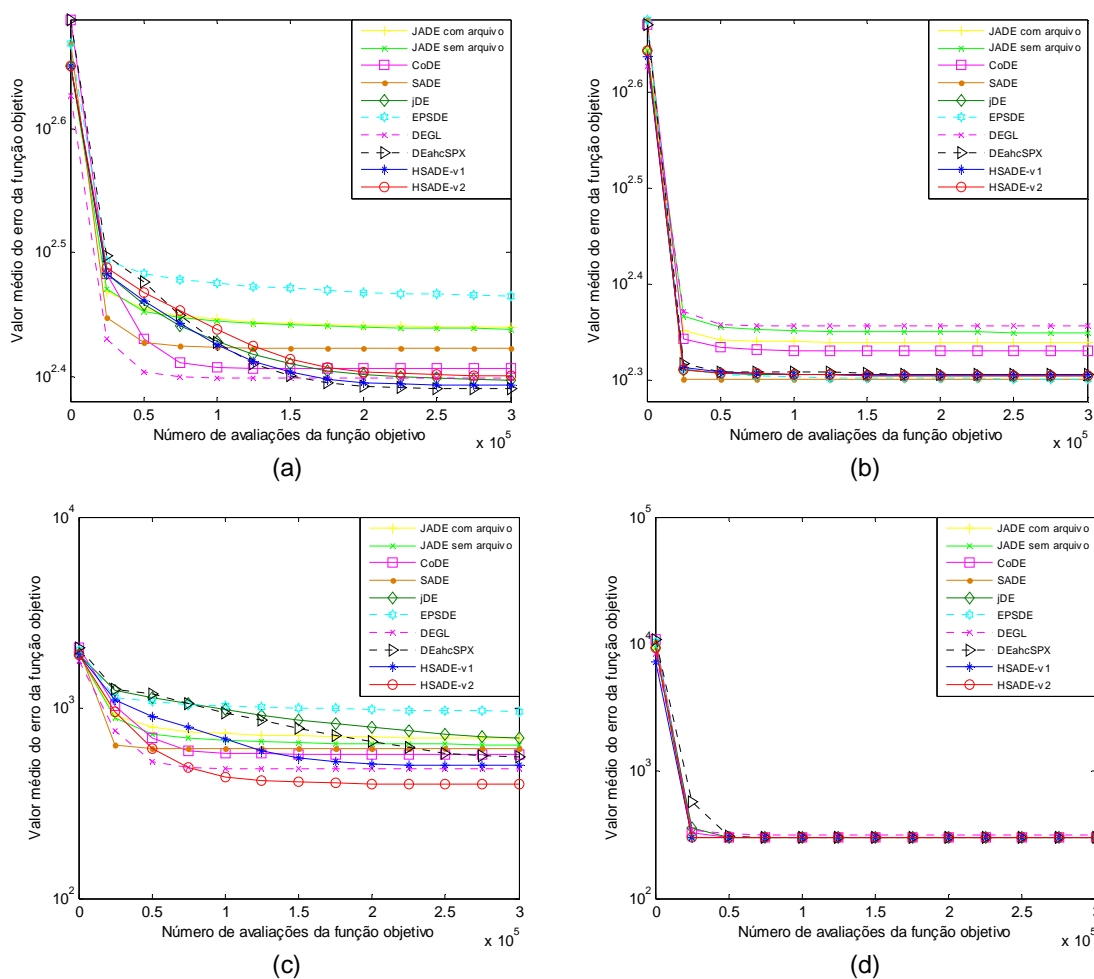


FIGURA 92: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{25}$ , (b)  $F_{26}$ , (c)  $F_{27}$ , (d)  $F_{28}$  para o caso  $D = 30$ .

FONTE: O autor (2014)

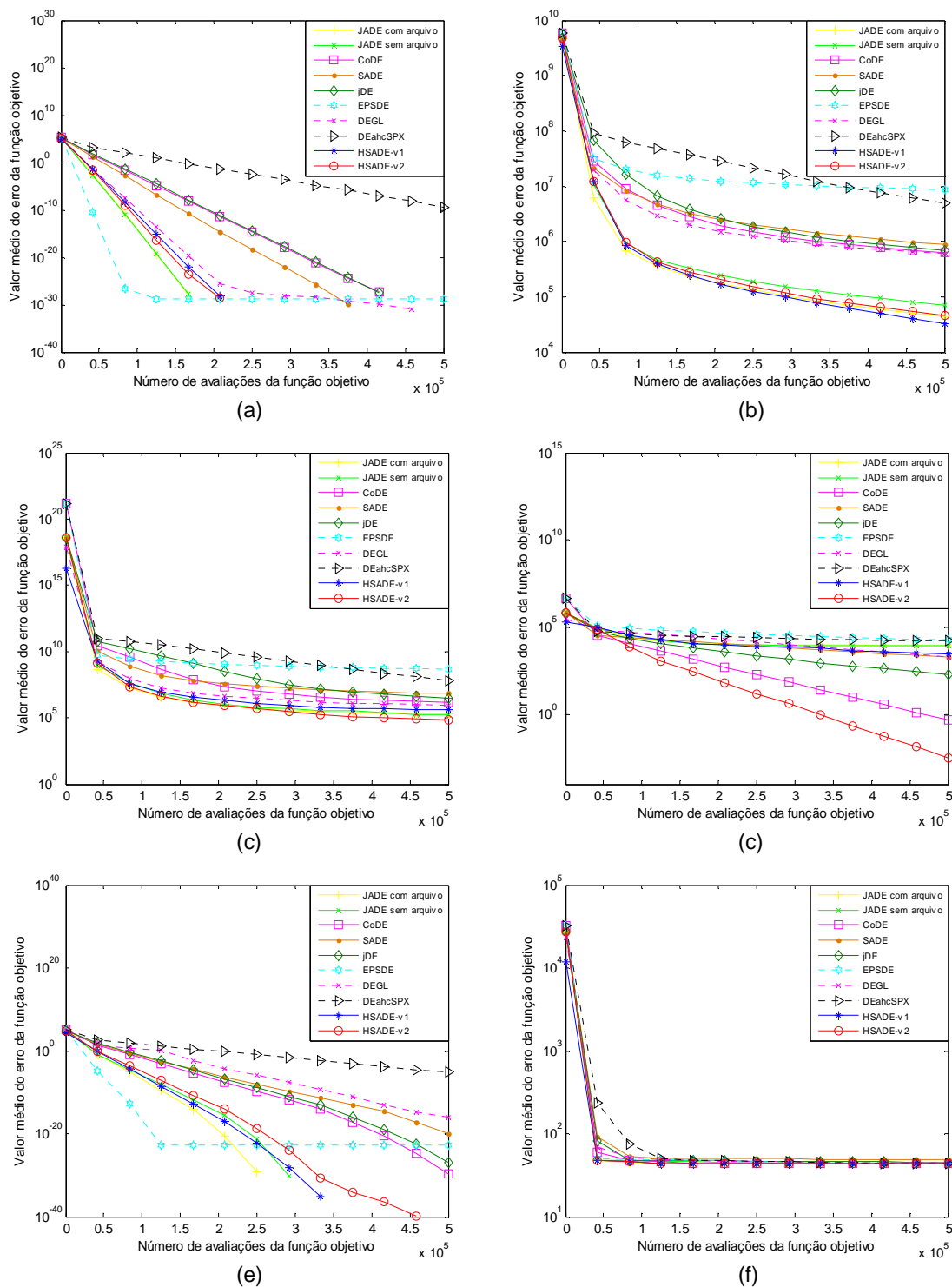


FIGURA 93: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções testes (a)  $F_1$ , (b)  $F_2$ , (c)  $F_3$ , (d)  $F_4$ , (e)  $F_5$ , (f)  $F_6$  para o caso  $D = 50$ .

FONTE: O autor (2014)

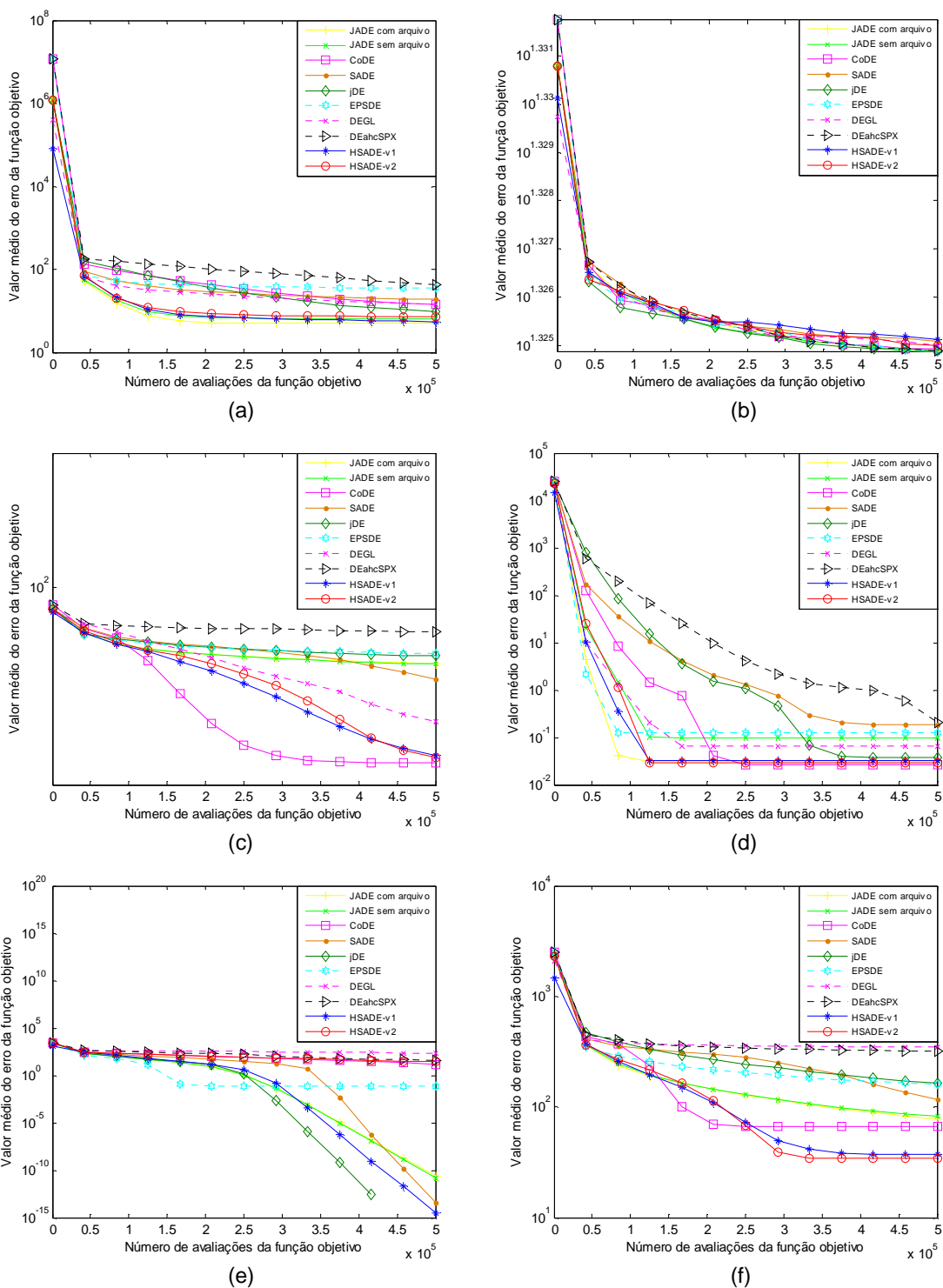


FIGURA 94: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_7$ , (b)  $F_8$ , (c)  $F_9$ , (d)  $F_{10}$ , (e)  $F_{11}$ , (f)  $F_{12}$  para o caso  $D = 50$ .

FONTE: O autor (2014)

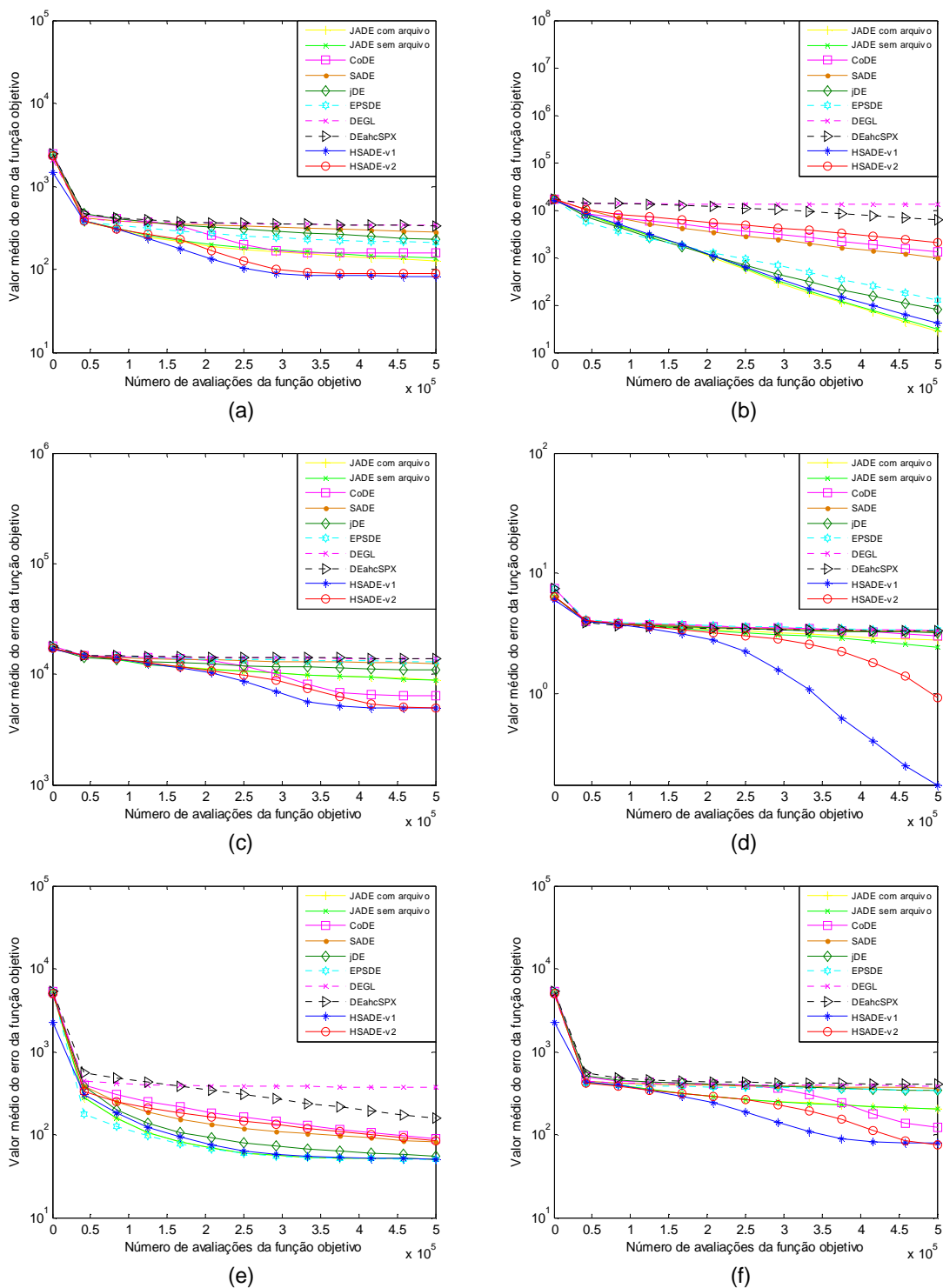


FIGURA 95: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{13}$ , (b)  $F_{14}$ , (c)  $F_{15}$ , (d)  $F_{16}$ , (e)  $F_{17}$ , (f)  $F_{18}$  para o caso  $D = 50$ .

FONTE: O autor (2014)

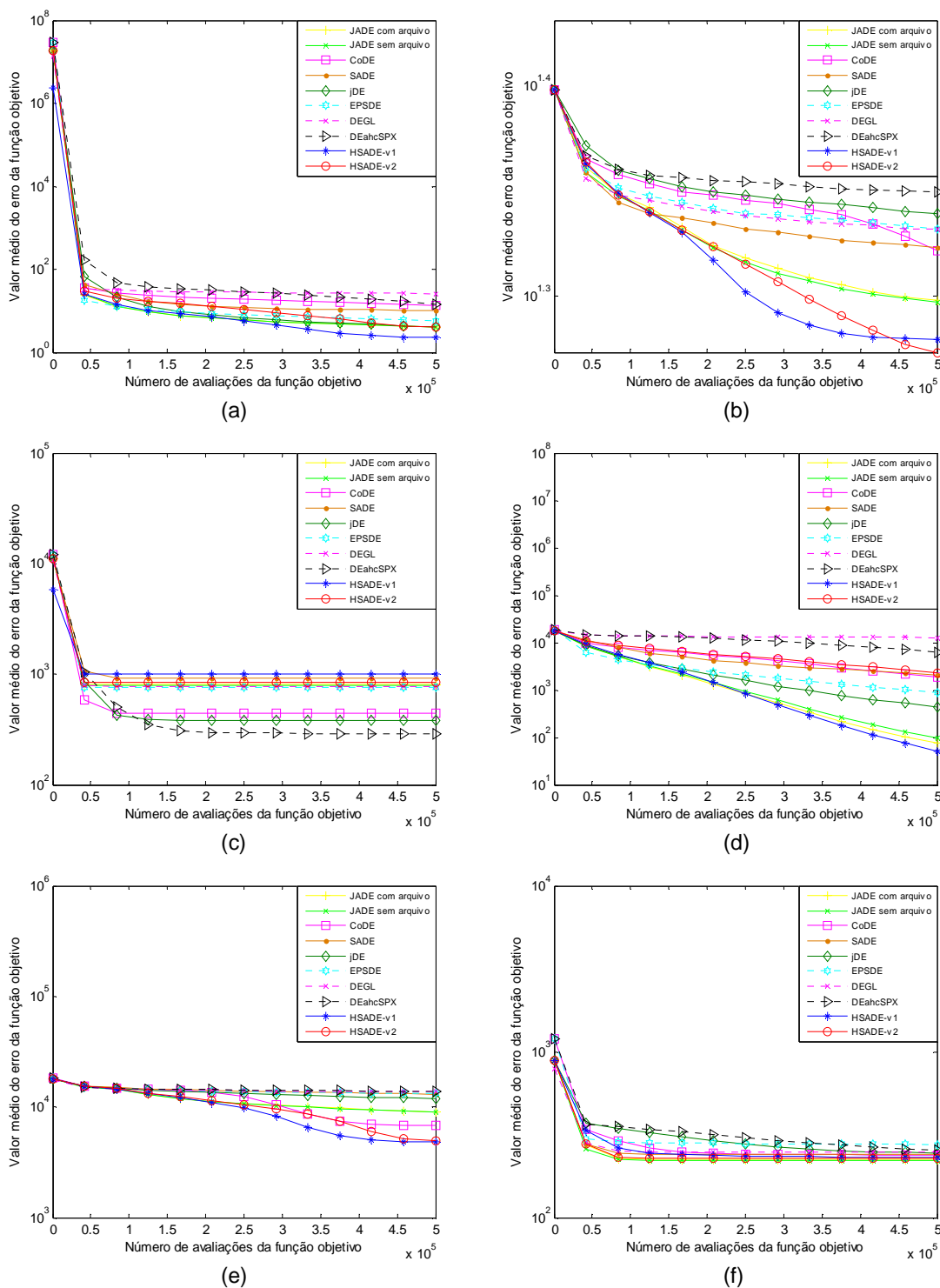


FIGURA 96: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADE-v1 e HSADE-v2 para as funções (a)  $F_{19}$ , (b)  $F_{20}$ , (c)  $F_{21}$ , (d)  $F_{22}$ , (e)  $F_{23}$ , (f)  $F_{24}$  para o caso  $D = 50$ .

FONTE: O autor (2014)

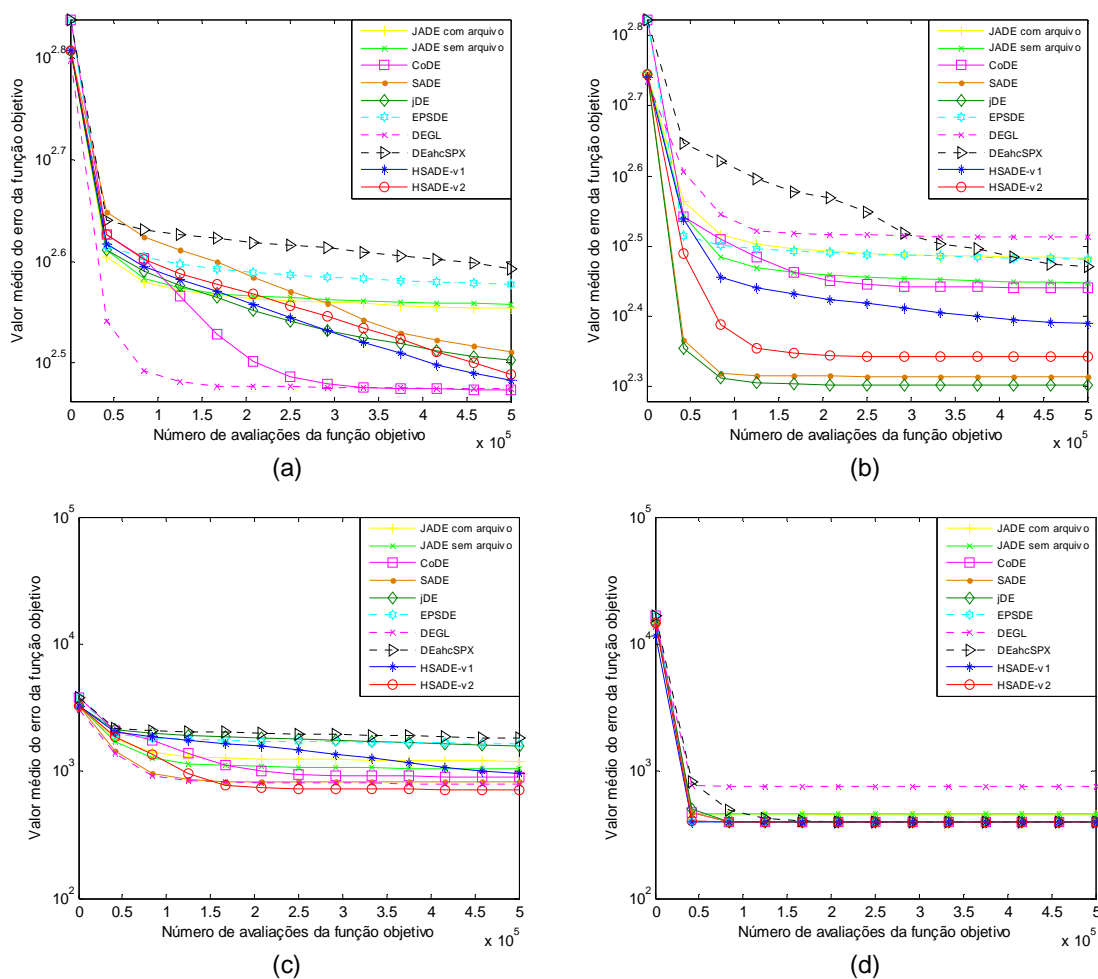


FIGURA 97: Gráfico de convergência de JADE com e sem arquivo, CoDE, jDE, SaDE, EPSDE, DEGL, DEahcSPX, HSADe-v1 e HSADe-v2 para as funções (a)  $F_{25}$ , (b)  $F_{26}$ , (c)  $F_{27}$ , (d)  $F_{28}$  para o caso  $D = 50$ .

FONTE: O autor (2014)