


unesp  **UNIVERSIDADE ESTADUAL PAULISTA**
“JÚLIO DE MESQUITA FILHO”
CAMPUS DE GUARATINGUETÁ

EDUARDO BATISTA DE MORAES BARBOSA

**Uma Heurística para Otimização de Meta-Heurísticas
por meio de Métodos Estatísticos**

Guaratinguetá

2016

Eduardo Batista de Moraes Barbosa

**Uma Heurística para Otimização de Meta-Heurísticas
por meio de Métodos Estatísticos**

Tese apresentada à Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, para obtenção do título de Doutor em Engenharia Mecânica na área de Gestão e Otimização.

Orientador: Prof. Dr. Edson Luiz França Senne
Coorientador: Prof. Dr. Messias Borges Silva

Guaratinguetá - SP

2016

B238h Barbosa, Eduardo Batista de Moraes
Uma heurística para otimização de meta-heurísticas por meio de métodos estatísticos / Eduardo Batista de Moraes Barbosa – Guaratinguetá, 2016
194 f. : il.
Bibliografia: f. 144-152

Tese (doutorado) – Universidade Estadual Paulista, Faculdade de Engenharia de Guaratinguetá, 2016.
Orientador: Prof Dr. Edson Luiz França Senne
Coorientador: Prof Dr. Messias Borges Silva

1. Heurística. 2. Otimização combinatória. 3. Inteligência artificial. 4. Estatística. Título

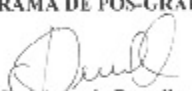
CDU 519.863(043)

EDUARDO BATISTA DE MORAES BARBOS

ESTA TESE FOI JULGADA ADEQUADA PARA A OBTENÇÃO DO TÍTULO DE
"DOUTOR EM CIÊNCIAS"

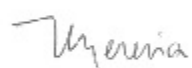
PROGRAMA: ENGENHARIA MECÂNICA
ÁREA: GESTÃO E OTIMIZAÇÃO

APROVADA EM SUA FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO


Prof. Dr. José Antonio Perrella Balestieri
Coordenador

BANCA EXAMINADORA:


Prof. Dr. EDSON LUIZ FRANÇA SENNE
Orientador/UNESP


Prof. Dr. MARCOS ANTONIO PEREIRA
UNESP-FEG


Prof. Dr. FERNANDO AUGUSTO SILVA MARINS
UNESP-FEG


Prof. Dr. ANTONIO AUGUSTO CHAVES
UNIFESP


Prof. Dr. FABRÍCIO MACHIEL GOMES
USP-EEL

Julho de 2016

DADOS CURRICULARES

EDUARDO BATISTA DE MORAES BARBOSA

NASCIMENTO	04.04.1977 - Cachoeira Paulista/SP
FILIAÇÃO	Geraldo Barbosa Filho Neuza Batista de Moraes Barbosa
1996/2000	Curso de Graduação Bacharelado em Computação Universidade de Taubaté, Taubaté/SP
2001/2002	Curso de Pós-Graduação Especialização em Informática Empresarial Faculdade de Engenharia de Guaratinguetá, Guaratinguetá/SP
2003/2006	Curso de Pós-Graduação Mestrado em Computação Aplicada Instituto Nacional de Pesquisas Espaciais, São José dos Campos/SP

Aos meus amados pais

AGRADECIMENTOS

Gostaria de expressar meu agradecimento primeiramente a DEUS por me guiar sempre, dando-me a orientação necessária para superar todas as dificuldades que aparecem em meu caminho. "DEUS dirige todos os acontecimentos, porque está em tudo".

Expresso meu carinho e gratidão às seguintes pessoas:

Meus pais, minha irmã e sobrinhos e minha esposa pelo incentivo, apoio, compreensão e paciência desde o início até a conclusão desta jornada.

Meu orientador Prof. Senne por compartilhar comigo sua valiosa experiência e pela oportunidade da orientação. Por confiar em mim, me guiar na condução desta pesquisa, pelo incentivo e pelas críticas, que continuamente me ajudam a amadurecer como pessoa e pesquisador. Pelas conversas sempre interessantes e ricas de conhecimento, que despertam a minha curiosidade e ampliam meu horizonte.

Meu coorientador Prof. Messias por compartilhar seu conhecimento e por me apresentar DOE. Pela confiança, oportunidade da orientação, incentivo e apoio para desenvolvimento desta pesquisa.

Prof. Antonio Augusto Chaves pela atenção na cessão do *benchmark* PCTSP e disponibilidade nos esclarecimentos das dúvidas a respeito do algoritmo CS.

Prof. Fabrício Maciel Gomes pela atenção nas discussões iniciais sobre DOE e meta-heurísticas.

Professores da Pós-Graduação pelo comprometimento na qualidade do ensino.

Amigos pela ajuda durante o curso.

Banca examinadora pelas sugestões e comentários para melhorar a qualidade desta pesquisa.

"... et ceux qui ne marchent que fort lentement peuvent avancer beaucoup davantage, s'ils suivent toujours le droit chemin, que ne font ceux qui courent et qui s'en éloignent".

René Descartes

BARBOSA, E. B. M. **Uma heurística para otimização de meta-heurísticas por meio de métodos estatísticos**. 2016. 194 f. Tese (Doutorado em Engenharia Mecânica) - Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2016.

RESUMO

A configuração de parâmetros de algoritmos, em especial, das meta-heurísticas, nem sempre é trivial e, frequentemente, é realizada *ad hoc* de acordo com o problema sob análise. A fim de resolver o problema de sintonização de meta-heurísticas, a presente pesquisa propõe uma metodologia que combina o uso de técnicas estatísticas robustas (ex.: Planejamento de Experimentos) e métodos eficientes de Inteligência Artificial (ex.: Algoritmos de Corrida). A ideia central desta metodologia é um método heurístico, denominado Algoritmo de Corrida Orientada por Heurística (HORA), capaz de explorar o espaço de busca para perseguir diferentes alternativas na vizinhança de uma configuração de parâmetros promissora e encontrar sistematicamente boas configurações candidatas para diferentes algoritmos. Em síntese, o método HORA concentra as buscas sobre configurações candidatas promissoras, criadas dinamicamente em um processo iterativo, e utiliza uma técnica estatística robusta para avaliar as diferentes alternativas e descartar aquelas de qualidade inferior, assim que reunir evidências estatísticas suficientes contra elas. A partir dos resultados de diversos estudos computacionais, em que diferentes meta-heurísticas foram aplicadas sobre dois problemas clássicos de otimização combinatória, apresentam-se evidências estatísticas que as sintonizações obtidas pelo HORA são competitivas em relação ao método de Corrida e seu tempo no processo de sintonização é amplamente vantajoso. Em um estudo complementar, um algoritmo já bem configurado da literatura foi sintonizado por meio da metodologia proposta e os resultados da nova sintonização foram comparados com a literatura. Os resultados demonstram que a sintonização obtida pelo HORA pode encontrar soluções de melhor qualidade em relação à sintonização original. Portanto, a partir dos resultados apresentados nesta pesquisa conclui-se que a metodologia para sintonização de meta-heurísticas por meio do método HORA é uma abordagem promissora que pode ser aplicada sobre diferentes meta-heurísticas para resolução de uma diversidade de problemas de otimização.

PALAVRAS-CHAVE: Meta-heurísticas. Otimização Combinatória. Método de Corrida. Planejamento de Experimentos.

BARBOSA, E. B. M. **A heuristic for optimization of metaheuristics by means of statistical methods.** 2016. 194 p. Thesis (PhD in Mechanical Engineering) - Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2016.

ABSTRACT

The fine-tuning of the algorithms parameters, specially, of the meta-heuristics, it is not always trivial and often is performed by *ad hoc* methods according to the problem under analysis. In order to solve the problem of tuning metaheuristics, this research proposes a methodology combining statistical robust techniques (e.g.: Design of Experiments) and efficient methods from Artificial Intelligence (e.g.: Racing Algorithms). The key idea of this methodology is a heuristic method, called Heuristic Oriented Racing Algorithm (HORA), which explores the search space looking for alternatives near of a promising candidate and consistently finds good candidates configuration for different algorithms. Briefly, HORA focuses its searches over the promising candidates configuration, dynamically created in an iterative process, and employs a robust statistical method to evaluate and discarding them, as soon as gather enough statistical evidence against them. The results of several studies, where different metaheuristics were applied to solve two classical combinatorial optimization problems, present statistical evidences that the settings obtained by HORA are competitive to the Racing Algorithms and its time in the fine-tuning process is widely advantageous. In a complementary study, an already well setting algorithm from the literature was tuned by means of the proposed methodology and the new settings were compared with the literature. The results show that the fine-tuning from HORA can find better quality solutions than the original ones. Therefore, from the results presented in this study it is concluded that the methodology for fine-tuning of metaheuristics by means of HORA is a promising approach, which can be applied on different metaheuristics to solve a diversity of optimization problems.

KEYWORDS: Metaheuristics. Combinatorial Optimization. Racing Algorithms. Design of Experiments.

LISTA DE ILUSTRAÇÕES

Figura 1 - Modelo de processo	25
Figura 2 - Representação geométrica de um fatorial completo 2^3	28
Figura 3 - Procedimento sequencial da RSM	32
Figura 4 - Perfis de superfícies de respostas. (a) Máximo; (b) Mínimo; (c) Ponto de sela.....	32
Figura 5 - Representação de um Planejamento Central Composto 2^3	33
Figura 6 - (a) Planejamento Fatorial 2^2 ; (b) CCD para $\alpha = 1$; (c) CCD para $\alpha > 1$	34
Figura 7 - Ilustração de um espaço de soluções	41
Figura 8 - Critérios de diversificação e intensificação.....	43
Figura 9 - Pseudocódigo de um algoritmo de LS	45
Figura 10 - Pseudocódigo da meta-heurística SA.....	48
Figura 11 - Pseudocódigo da meta-heurística GA.....	50
Figura 12 - Representação da população no GA	51
Figura 13 - Processo de cruzamento com corte em 1-ponto	52
Figura 14 - Processo de mutação	53
Figura 15 - Pseudocódigo da meta-heurística VNS.....	54
Figura 16 - Pseudocódigo de um algoritmo de força bruta.....	63
Figura 17 - Representação gráfica da quantidade de computação envolvida no método <i>Hoeffding race</i>	65
Figura 18 - Pseudocódigo de um algoritmo de Corrida.....	68
Figura 19 - Representação gráfica da quantidade de computação envolvida pelas abordagens de Corrida (área sombreada) e de força bruta (área tracejada).....	69
Figura 20 - Processo dinâmico de inclusão (em preto) e exclusão (em cinza) de alternativas no conjunto S de configurações candidatas.....	77
Figura 21 - Esquema lógico da abordagem heurística elaborada nesta pesquisa	78
Figura 22 - Pseudocódigo do algoritmo HORA	79
Figura 23 - Pseudocódigo do algoritmo para criação de novas configurações candidatas	80
Figura 24 - Pseudocódigo do algoritmo para avaliação de configurações candidatas.....	81
Figura 25 - Pseudocódigo do algoritmo para seleção de configurações candidatas sobreviventes	82
Figura 26 - Processamento das atividades para a sequência 1-2-4-3.....	96
Figura 27 - Desempenho das meta-heurísticas SA_H e SA_C	105

Figura 28 - Qualidade das soluções das meta-heurísticas SA _H e SA _C	105
Figura 29 - Desempenho das meta-heurísticas GA _H e GA _C	112
Figura 30 - Qualidade das soluções das meta-heurísticas GA _H e GA _C	113
Figura 31 - Desempenho das meta-heurísticas VNS _H e VNS _C	119
Figura 32 - Qualidade das soluções das meta-heurísticas VNS _H e VNS _C	120
Figura 33 - Fluxograma do método CS.....	126
Quadro 1 - Analogia entre sistema físico e o algoritmo SA.....	47

LISTA DE TABELAS

Tabela 1 - Fatorial fracionado $2^{(3-1)}$ de resolução III	30
Tabela 2 - Esquema de um Planejamento Plackett-Burman com 7 fatores	31
Tabela 3 - Esquema da Análise de Variância (ANOVA)	37
Tabela 4 - Desempenho de quatro algoritmos (M) sobre dez exemplares diferentes de um problema	71
Tabela 5 - Conjunto de configurações candidatas. ($k = 1$; $ S = 4$)	83
Tabela 6 - Desempenho da meta-heurística GA coletados sob diferentes configurações	84
Tabela 7 - Conjunto de configurações candidatas sobreviventes. ($k = 2$; $ S = 7$)	84
Tabela 8 - Desempenho das configurações candidatas sobreviventes.....	84
Tabela 9 - Conjunto de configurações candidatas sobreviventes. ($k = 3$; $ S = 6$)	85
Tabela 10 - Desempenho das configurações candidatas sobreviventes.....	85
Tabela 11 - Problema ilustrativo de sequenciamento	95
Tabela 12 - Atraso total ponderado para a sequência 1-2-3-4.....	96
Tabela 13 - Parâmetros da meta-heurística SA selecionados para sintonização	101
Tabela 14 - TSP: Sintonização da meta-heurística SA sob diferentes métodos.....	102
Tabela 15 - TSP: Resultados computacionais da meta-heurística SA	104
Tabela 16 - wt40: Sintonização da meta-heurística SA sob diferentes métodos.....	106
Tabela 17 - wt50: Sintonização da meta-heurística SA sob diferentes métodos.....	106
Tabela 18 - wt100: Sintonização da meta-heurística SA sob diferentes métodos.....	106
Tabela 19 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística SA (<i>benchmarks</i> wt40, wt50 e wt100)	108
Tabela 20 - Parâmetros da meta-heurística GA selecionados para sintonização	109
Tabela 21 - TSP: Sintonização da meta-heurística GA sob diferentes métodos.....	109
Tabela 22 - TSP: Resultados computacionais da meta-heurística GA	111
Tabela 23 - wt40: Sintonização da meta-heurística GA sob diferentes métodos.....	113
Tabela 24 - wt50: Sintonização da meta-heurística GA sob diferentes métodos.....	114
Tabela 25 - wt100: Sintonização da meta-heurística GA sob diferentes métodos.....	114
Tabela 26 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística GA (<i>benchmarks</i> wt40, wt50 e wt100).....	115
Tabela 27 - Parâmetros da meta-heurística VNS selecionados para sintonização	116
Tabela 28 - TSP: Sintonização da meta-heurística VNS sob diferentes métodos.....	117

Tabela 29 - TSP: Resultados computacionais da meta-heurística VNS	118
Tabela 30 - wt40: Sintonização da meta-heurística VNS sob diferentes métodos.....	121
Tabela 31 - wt50: Sintonização da meta-heurística VNS sob diferentes métodos.....	121
Tabela 32 - wt100: Sintonização da meta-heurística VNS sob diferentes métodos.....	121
Tabela 33 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística VNS (<i>benchmarks</i> wt40, wt50 e wt100)	122
Tabela 34 - Qualidade das soluções produzidas por diferentes meta-heurísticas sintonizadas pelos métodos HORA e Corrida	124
Tabela 35 - Parâmetros da meta-heurística híbrida CS-SA adotados na literatura.....	131
Tabela 36 - Parâmetros da meta-heurística híbrida CS-GA adotados na literatura.....	131
Tabela 37 - Parâmetros da meta-heurística híbrida CS-SA selecionados para sintonização.....	132
Tabela 38 - PCTSP: Sintonização da meta-heurística híbrida CS-SA pelo método HORA	133
Tabela 39 - PCTSP: Resultados computacionais resumidos do CS-SA	133
Tabela 40 - Parâmetros da meta-heurística híbrida CS-GA selecionados para sintonização	134
Tabela 41 - PCTSP: Sintonização da meta-heurística híbrida CS-GA pelo método HORA	135
Tabela 42 - PCTSP: Resultados computacionais resumidos do CS-GA	136
Tabela 43 - Parâmetros da meta-heurística GA selecionados para sintonização	153
Tabela 44 - Planejamento fatorial 2^4 para sintonização da meta-heurística GA	154
Tabela 45 - ANOVA de um planejamento fatorial 2^4 (Exemplar 42)	154
Tabela 46 - Região promissora identificada durante o processo de sintonização da meta- heurística GA.....	155
Tabela 47 - Planejamento CCD para sintonização da meta-heurística GA	156
Tabela 48 - ANOVA de um planejamento CCD (Exemplar 42).....	157
Tabela 49 - Configurações da meta-heurística GA para os exemplares do conjunto de treinamento.....	157
Tabela 50 - Espaço de busca de parâmetro da meta-heurística GA	158
Tabela 51 - wt40: Sintonização da meta-heurística GA sob diferentes métodos.....	159
Tabela 52 - wt40: Resultados computacionais da meta-heurística SA	160

Tabela 53 - wt50: Resultados computacionais da meta-heurística SA	163
Tabela 54 - wt100: Resultados computacionais da meta-heurística SA	166
Tabela 55 - wt40: Resultados computacionais da meta-heurística GA	169
Tabela 56 - wt50: Resultados computacionais da meta-heurística GA	172
Tabela 57 - wt100: Resultados computacionais da meta-heurística GA	175
Tabela 58 - wt40: Resultados computacionais da meta-heurística VNS	178
Tabela 59 - wt50: Resultados computacionais da meta-heurística VNS	181
Tabela 60 - wt100: Resultados computacionais da meta-heurística VNS	184
Tabela 61 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-SA.....	187
Tabela 62 - PCTSP: $\gamma_i \in [1, 1000]$ - Resultados computacionais do CS-SA.....	189
Tabela 63 - PCTSP: $\gamma_i \in [1, 10000]$ - Resultados computacionais do CS-SA.....	190
Tabela 64 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-GA.....	191
Tabela 65 - PCTSP: $\gamma_i \in [1, 1000]$ - Resultados computacionais do CS-GA.....	193
Tabela 66 - PCTSP: $\gamma_i \in [1, 10000]$ - Resultados computacionais do CS-GA.....	194

LISTA DE ABREVIATURAS E SIGLAS

ANOVA	Análise de Variância
CCD	Planejamento Central Composto
DOE	Planejamento de Experimentos
GA	Algoritmo Genético
HORA	Heurística Orientada por Algoritmo de Corrida
ILS	Pesquisa Local Iterativa
LS	Pesquisa Local
PPB	Planejamento Plackett-Burman
PSO	Otimização por Enxame de Partículas
RSM	Metodologia da Superfície de Respostas
SA	Recozimento Simulado
TSP	Problema do Caixeiro Viajante
TTP	Problema do Atraso Total
TWTP	Problema de Minimização do Atraso Total Ponderado em Máquina Única
VNS	Pesquisa em Vizinhança Variável

SUMÁRIO

1. INTRODUÇÃO.....	17
1.1 MOTIVAÇÃO	17
1.2 DELIMITAÇÃO DO PROBLEMA	17
1.3 HIPÓTESE DA PESQUISA.....	20
1.4 OBJETIVOS	21
1.5 PRINCIPAIS CONTRIBUIÇÕES.....	22
1.6 ESTRUTURA E ORGANIZAÇÃO	23
2. EXPERIMENTAÇÃO E PLANEJAMENTO DE EXPERIMENTOS.....	25
2.1 INTRODUÇÃO	25
2.2 PLANEJAMENTO DE EXPERIMENTOS.....	26
2.2.1 Planejamentos Fatoriais	28
2.2.2 Planejamentos Plackett-Burman	30
2.3 METODOLOGIA DE SUPERFÍCIE DE RESPOSTAS	31
2.4 AJUSTE DE MODELOS DE REGRESSÃO.....	34
2.4.1 Análise de Variância	36
2.5 OUTRAS TÉCNICAS DE PLANEJAMENTOS EXPERIMENTAIS	38
3. META-HEURÍSTICAS E O PROBLEMA DE SINTONIZAÇÃO DE PARÂMETROS	41
3.1 INTRODUÇÃO	41
3.2 META-HEURÍSTICAS	42
3.2.1 Busca Local	45
3.2.2 Recozimento Simulado	46
3.2.3 Algoritmo Genético	49
3.2.4 Busca em Vizinhança Variável	54
3.3 O PROBLEMA DA SINTONIZAÇÃO DE PARÂMETROS.....	56
3.4 DECLARAÇÃO DO PROBLEMA DE PESQUISA	59
4. ESTRATÉGIAS PARA OTIMIZAÇÃO DE META-HEURÍSTICAS E O MÉTODO DE CORRIDA.....	61
4.1 INTRODUÇÃO	61
4.2 ESTRATÉGIA DE FORÇA BRUTA.....	62
4.3 ESTRATÉGIA DE CORRIDA	63

4.3.1 Algoritmo de Corrida.....	66
4.3.2 Teste Estatístico de Friedman.....	70
4.4 ALGUMAS ABORDAGENS COM MÉTODOS DE CORRIDA.....	72
5. METODOLOGIA PROPOSTA PARA OTIMIZAÇÃO DE META-HEURÍSTICAS.....	75
5.1 INTRODUÇÃO.....	75
5.2 DESCRIÇÃO GERAL.....	76
5.2.1 Algoritmo de Corrida Orientada por Heurística.....	79
5.2.2 Exemplo Ilustrativo do Algoritmo HORA.....	83
5.3 A ESTRUTURA HEURÍSTICA DO ALGORITMO HORA.....	85
6. PROBLEMAS DE OTIMIZAÇÃO COMBINATÓRIA E ESTUDOS COMPUTACIONAIS.....	88
6.1 INTRODUÇÃO.....	88
6.1.1 Propriedades dos Problemas de Otimização.....	89
6.2 PROBLEMA DO CAIXEIRO VIAJANTE.....	91
6.2.1 Definição e Formulação Matemática.....	92
6.3 PROBLEMA DE SEQUENCIAMENTO DE ATIVIDADES.....	93
6.3.1 Definição e Formulação Matemática.....	95
6.4 ESTUDOS COMPUTACIONAIS.....	97
6.4.1 Sintonização da Meta-heurística SA.....	101
6.4.1.1 Problema do Caixeiro Viajante.....	101
6.4.1.2 Problema de Sequenciamento de Atividades.....	106
6.4.2 Sintonização da Meta-heurística GA.....	109
6.4.2.1 Problema do Caixeiro Viajante.....	109
6.4.2.2 Problema de Sequenciamento de Atividades.....	113
6.4.3 Sintonização da Meta-heurística VNS.....	116
6.4.3.1 Problema do Caixeiro Viajante.....	117
6.4.3.2 Problema de Sequenciamento de Atividades.....	120
6.4.4 Análise dos Estudos Computacionais.....	123
7. ESTUDO COMPLEMENTAR.....	125
7.1 INTRODUÇÃO.....	125
7.1.1 Busca por Agrupamento.....	125
7.1.2 Problema do Caixeiro Viajante com Lucro.....	127
7.2 ESTUDOS COMPUTACIONAIS.....	130

7.2.1 Sintonização da Meta-heurística CS-SA.....	132
7.2.2 Sintonização da Meta-heurística CS-GA.....	134
8. CONCLUSÃO	137
8.1 INTRODUÇÃO	137
8.2 REVISÃO DAS QUESTÕES DE PESQUISA.....	138
8.3 RESUMO DAS CONTRIBUIÇÕES	140
8.4 TRABALHOS FUTUROS	142
REFERÊNCIAS BIBLIOGRÁFICAS	144
APÊNCICE A - Metodologia para Sintonização de Meta-heurísticas	153
A.1 META-HEURÍSTICA GA	153
APÊNCICE B - Resultados Computacionais - Problema de Sequenciamento de Atividades	160
B.1 META-HEURÍSTICA SA	160
B.2 META-HEURÍSTICA GA	169
B.3 META-HEURÍSTICA VNS	178
APÊNCICE C - Resultados Computacionais - Problema do Caixeiro Viajante com Coleta de Prêmios	187
C.1 META-HEURÍSTICA CS-SA.....	187
C.2 META-HEURÍSTICA CS-GA	191

1. INTRODUÇÃO

1.1 MOTIVAÇÃO

Muitos algoritmos e, em particular, as meta-heurísticas possuem um conjunto de parâmetros que afetam seu desempenho em termos de qualidade da solução e tempo de execução. A definição incorreta desses parâmetros pode provocar, em casos extremos, um comportamento inesperado do algoritmo e fazê-lo convergir rapidamente para uma solução ótima local. Em outras situações, o comportamento pode ser aleatório, de modo que o algoritmo não converge para uma solução no limite de tempo aceitável. Em ambos casos, o esforço computacional é desperdiçado com soluções de baixa qualidade.

As meta-heurísticas têm a capacidade de se adaptar a um vasto conjunto de problemas com pequenas modificações de parâmetros em cada caso específico. Essa adaptação normalmente envolve um esforço relativamente grande para sintonizar os parâmetros. Algumas vezes, mesmo as sintonizações menos precisas podem encontrar soluções de qualidade razoável em tempo aceitável¹.

A alta flexibilidade das meta-heurísticas permite encontrar soluções razoáveis para um vasto conjunto de problemas, mas, ao mesmo tempo, dificulta a obtenção de soluções ótimas para cada um deles. Nessas circunstâncias, a configuração dos parâmetros, ou sintonização, de meta-heurísticas revela-se um dos desafios de pesquisa mais importantes no contexto de concepção e aplicação desses algoritmos², uma vez que a sua efetiva utilização é extremamente dependente da seleção correta dos parâmetros e da atribuição de valores adequados a eles.

1.2 DELIMITAÇÃO DO PROBLEMA

Informalmente, o problema de sintonização de meta-heurísticas consiste em determinar os valores dos parâmetros que permitem um algoritmo atingir o melhor desempenho possível em tempo aceitável. A definição do melhor desempenho pode variar amplamente dependendo das necessidades da aplicação e pode ser, por exemplo, a qualidade da solução obtida no

¹ Deve-se ressaltar, no entanto, que a configuração padrão de uma meta-heurística não supera o desempenho de um algoritmo *ad hoc*, quando este último existe, para resolver determinado problema.

² Os termos meta-heurísticas e algoritmos são utilizados alternadamente ao longo desta pesquisa.

limite de tempo aceitável, o tempo necessário para alcançar a solução ótima, a percentagem de vezes que o algoritmo alcança uma solução alvo, dentre outros.

A sintonização de uma meta-heurística não é uma tarefa trivial. Estes algoritmos são formados por componentes específicos como, por exemplo, a representação do problema, as estruturas de vizinhança, os parâmetros, dentre outros. Tais componentes podem influenciar o desempenho das meta-heurísticas na solução de uma diversidade de problemas de otimização. Em geral, há muitos desafios relacionados aos parâmetros, que exigem a utilização de métodos inovadores.

Um desses desafios diz respeito ao domínio dos parâmetros (YUAN e GALLAGHER, 2007; SMIT e EIBEN, 2009). Por exemplo, eles podem ser categóricos, numéricos (contínuos ou discretos) ou condicionais, isto é, existem na condição de outros existirem. Esta diversidade tem impacto na definição das alternativas de configuração desses parâmetros, pois podem se tornar extensas e inviáveis devido às combinações possíveis entre parâmetros e seus domínios. A presente pesquisa se concentra em estudos sobre parâmetros numéricos, tais como a taxa de arrefecimento na meta-heurística Recozimento Simulado, as taxas de cruzamento e mutação no Algoritmo Genético, o número de estruturas de vizinhança na Busca em Vizinhança Variável, etc. Portanto, outros componentes específicos, em geral, de domínio categórico, não serão abordados nesta pesquisa.

A avaliação da qualidade dos ajustes em algoritmos estocásticos pode exigir muitas execuções. Nesse contexto, o desafio consiste em determinar o número ideal de execuções necessárias para sintonizar adequadamente os parâmetros.

Outro desafio está relacionado à configuração dos parâmetros. Por exemplo, há situações em que o ajuste é adequado para uma classe de problemas, por exemplo roteamento de veículos, mas inadequado para outras classes, como sequenciamento de atividades. Este mesmo problema se estende entre exemplares de uma mesma classe³.

Além desses, um outro desafio se refere à estratégia de sintonização (COY et al. 2000; BIRATTARI et al., 2002; ADESON-DÍAZ e LAGUNA, 2006; HUTTER et al., 2009). Ou seja, se o interesse é sintonizar a meta-heurística para resolver um problema específico, então a estratégia mais adequada é buscar a configuração que encontra a melhor solução para o problema. Por outro lado, se o interesse é uma classe de problemas, então a melhor estratégia

³ Uma classe de problemas envolve todos os exemplares que compartilham estruturas semelhantes. Por exemplo, no problema de sequenciamento de atividades, o número de atividades pode distinguir os exemplares.

pode ser aquela que encontra a configuração com o melhor desempenho global sobre os diferentes exemplares.

Estes desafios são de interesse de diferentes comunidades de pesquisa e, por isso, é possível encontrar na literatura uma diversidade de trabalhos que descrevem abordagens variadas para solucioná-los. Muitos desses trabalhos empregam metodologias experimentais, que obtêm um bom nível de sucesso em grande parte dos estudos que propõem uma abordagem empírica na sintonização de meta-heurísticas.

A metodologia de Planejamento de Experimentos (em Inglês, *Design of Experiments - DOE*) é uma abordagem sistemática e rigorosa da área de engenharia, voltada para resolução de problemas que requerem a aplicação de princípios e técnicas para assegurar conclusões válidas e objetivas (MONTGOMERY, 2012). Um experimento concebido estatisticamente oferece vantagens sobre outros que não utilizam técnicas estatísticas, uma vez que a atenção é centrada na medição das fontes de variabilidade, o número de experimentos é determinado de maneira confiável, a detecção dos efeitos é mais precisa e as conclusões são exatas.

Em síntese, as etapas para utilização de DOE podem ser sumarizadas em:

- Caracterização: consiste em compreender o processo como um todo, isto é, avaliar a influência dos fatores e determinar a sua ordem de importância;
- Modelagem: corresponde à confecção do modelo para o processo, ou seja, o estabelecimento da relação de causa e efeito entre fatores e resposta na forma de uma função matemática; e
- Otimização: estabelece os ajustes adequados aos fatores que afetam o processo a fim de encontrar a solução ótima.

Estas três etapas estão diretamente relacionadas ao problema de sintonização de meta-heurísticas, tal que os algoritmos podem ser associados ao processo ao qual se aplica DOE, e seus parâmetros são os fatores desse processo.

Além de DOE, na última década foi proposta uma metodologia robusta e bem estabelecida que emprega conceitos de Inteligência Artificial a partir do método de Corrida para auxiliar a sintonização de meta-heurísticas. A abordagem de Corrida é baseada em testes estatísticos que avaliam o desempenho de diferentes configurações candidatas de maneira iterativa em um processo cíclico, denominado Corrida. Sempre que há evidências estatísticas significativas sobre a inferioridade de uma alternativa (configuração candidata), ela é então descartada da corrida (BIRATTARI et al., 2002). Naturalmente, as configurações candidatas

que, eventualmente, sobrevivem são estatisticamente melhores do que aquelas descartadas durante o processo.

A qualidade da sintonização sugerida pode ser avaliada a partir de execuções do algoritmo alvo com a nova configuração, tal que melhorias no desempenho do algoritmo indicam uma sintonização de boa qualidade. No entanto, uma vez que no início do processo todas as configurações candidatas são avaliadas, a abordagem de Corrida torna-se limitada a situações em que o espaço de busca de parâmetros é finito e enumerável (STYLES e HOOS, 2013). Por outro lado, se o espaço de busca é composto por algumas centenas de candidatos, poucas alternativas podem ser eliminadas do processo, devido à similaridade dos resultados coletados durante a corrida.

A presente pesquisa de doutorado define uma metodologia para auxiliar a sintonização de meta-heurísticas a partir da combinação dos conceitos de DOE e Corrida. Nesta pesquisa, entende-se que resolver o problema de sintonização de meta-heurísticas requer um outro algoritmo com capacidade de pesquisar o espaço de busca de parâmetros de maneira eficaz a fim de determinar valores adequados para os parâmetros de um algoritmo alvo. Assim, a presente pesquisa propõe o desenvolvimento de um método heurístico para trabalhar em um nível de abstração mais elevado, com o propósito de determinar automaticamente a melhor sintonização de parâmetros para diferentes algoritmos alvos.

1.3 HIPÓTESE DA PESQUISA

A hipótese central desta pesquisa é:

Boas configurações para diferentes meta-heurísticas podem ser obtidas com base em análises estatísticas das soluções de um vasto conjunto de exemplares de uma classe de problemas.

Algumas outras questões de pesquisa são:

- Qual a influência dos parâmetros no desempenho das meta-heurísticas?
- Qual a configuração ideal para uma meta-heurística obter um desempenho satisfatório?
- Existe uma configuração única da meta-heurística para diferentes problemas?

- A metodologia para sintonização de meta-heurísticas é eficaz para um algoritmo já bem configurado?
- Os resultados obtidos demonstram que a metodologia pode ser uma ferramenta promissora?

O propósito desta pesquisa é apresentar uma metodologia que possa ser utilizada de maneira eficiente na sintonização de diferentes meta-heurísticas aplicadas na solução de qualquer classe de problemas de otimização. Para ilustrar este objetivo, os estudos estarão centrados nas meta-heurísticas Recozimento Simulado, Algoritmo Genético e Busca em Vizinhaça Variável, aplicadas na solução de dois problemas clássicos de otimização combinatória: Caixeiro Viajante e Sequenciamento de Atividades.

1.4 OBJETIVOS

A presente pesquisa de doutorado pretende investigar a utilização conjunta de técnicas de Estatística e de Inteligência Artificial a fim de definir uma metodologia para auxiliar a sintonização de meta-heurísticas. O objetivo da pesquisa é desenvolver a partir destes estudos um método heurístico capaz de encontrar sistematicamente boas configurações para diferentes algoritmos. A metodologia é baseada na definição de um espaço de busca de configurações candidatas com o auxílio de DOE e na aplicação do método de Corrida para determinar automaticamente a melhor configuração de uma meta-heurística.

Os objetivos específicos da pesquisa são:

- Pesquisar sobre a utilização combinada de técnicas de DOE e método de Corrida no contexto da sintonização de meta-heurísticas;
- Definir uma metodologia para auxiliar a sintonização de meta-heurísticas a partir da integração de DOE e método de Corrida;
- Desenvolver um método heurístico para sintonização automática de meta-heurísticas;
- Abreviar o esforço e o tempo investidos na sintonização de diferentes meta-heurísticas;
- Garantir confiabilidade na escolha dos valores para os parâmetros das meta-heurísticas;

- Possibilitar a replicação, bem como a reprodução, dos estudos de sintonização de meta-heurísticas; e
- Facilitar o processo de sintonização de meta-heurísticas.

1.5 PRINCIPAIS CONTRIBUIÇÕES

Identifica-se como a principal contribuição desta pesquisa a definição de uma metodologia para sintonização de meta-heurísticas com a utilização combinada de DOE e método de Corrida. A ideia central é o desenvolvimento de um método heurístico, para explorar de maneira eficaz o espaço de busca de parâmetros e perseguir diferentes configurações na vizinhança de uma configuração promissora.

Neste método, as configurações candidatas são criadas dinamicamente em um espaço de busca pré-definido. Esta característica se diferencia do conceito original proposto pelo método de Corrida, cujo melhor candidato é escolhido a partir de um conjunto limitado em que as configurações candidatas são definidas *a priori* no espaço de busca.

No método heurístico desenvolvido, a única limitação está relacionada ao espaço de busca, que é definido a partir de estudos experimentais com DOE. Com essa definição, as configurações candidatas são criadas arbitrariamente e avaliadas em um processo iterativo durante a execução do algoritmo. Por isso, o conjunto de configurações candidatas é dinâmico, ou seja, constantemente modificado. Algumas vezes ele aumenta com a inclusão de novas alternativas e outras ele diminui, devido as exclusões de configurações candidatas de qualidade inferior.

Uma outra contribuição da pesquisa está relacionada ao esforço e tempo investidos na sintonização de meta-heurísticas. A ausência de uma metodologia para sintonização de meta-heurísticas demanda um esforço maior e, conseqüentemente, mais tempo, no desenvolvimento dos algoritmos e, muitas vezes, conhecimento específico do problema sob análise.

Ao longo desta pesquisa foram efetuadas revisões bibliográficas e descrições de:

- Meta-heurísticas utilizadas na pesquisa, com foco nos Recozimento Simulado, Algoritmos Genéticos e Busca em Vizinhança Variável;
- Planejamento de experimentos, com atenção especial aos Planejamentos Fatoriais e Metodologia da Superfície de Respostas;
- Método de Corrida, que são a inspiração do método heurístico desenvolvido;

- Problema de sintonização de meta-heurísticas; e
- Abordagens contemporâneas empregadas na sintonização de meta-heurísticas.

Destacam-se como contribuições científicas a redação de artigos científicos, apresentações de trabalhos em conferências nacionais e internacionais e participações em eventos regionais de Pós-Graduação:

- Uma publicação em revista internacional
 - *International Journal of Engineering Mathematics*;
- Um artigo submetido para avaliação em revista internacional
 - *Computers & Industrial Engineering* (IF 1,783);
- Um artigo submetido para avaliação em revista nacional
 - *Gestão e Produção* (Qualis B3);
- Uma publicação em conferência internacional;
- Duas publicações em conferências nacionais; e
- Duas publicações em jornadas de Pós-Graduação.

1.6 ESTRUTURA E ORGANIZAÇÃO

O documento desta pesquisa está organizado em 8 capítulos. Após o capítulo de introdução, no Capítulo 2 são apresentados os aspectos teóricos sobre DOE com especial atenção para os Planejamentos Fatoriais e Metodologia de Superfície de Respostas. O capítulo apresenta, também, uma introdução a respeito de ajustes de modelos de regressão e Análise de Variância e uma breve introdução sobre outras técnicas de DOE.

O Capítulo 3 descreve o funcionamento básico de algumas meta-heurísticas, com destaque para aquelas utilizadas no desenvolvimento da pesquisa: Recozimento Simulado, Algoritmos Genéticos e Busca em Vizinhança Variável. Esse capítulo é finalizado com a declaração do problema de pesquisa, isto é, o problema de sintonização de meta-heurísticas.

No Capítulo 4 são apresentadas diferentes abordagens para o problema de sintonização de meta-heurísticas, bem como o método de Corrida, cujos conceitos são utilizados no desenvolvimento da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa.

O Capítulo 5 apresenta a contribuição da pesquisa para resolução do problema de sintonização de parâmetros de meta-heurísticas. Nesse capítulo se encontra uma descrição em detalhes da metodologia que propõe o uso combinado de DOE e método de Corrida.

No Capítulo 6 é realizada uma breve introdução sobre problemas de otimização combinatória e dois problemas clássicos da literatura: o Problema do Caixeiro Viajante e o Problema de Sequenciamento de Atividades. O capítulo apresenta, também, os resultados dos estudos computacionais realizados durante o desenvolvimento desta pesquisa.

O Capítulo 7 apresenta um estudo computacional complementar a partir da aplicação da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa. Nesse capítulo considera-se que a meta-heurística híbrida Busca por Agrupamento (em Inglês, *Clustering Search*) é um algoritmo já bem configurado e avalia-se o desempenho dos resultados obtidos antes e após a aplicação da metodologia proposta.

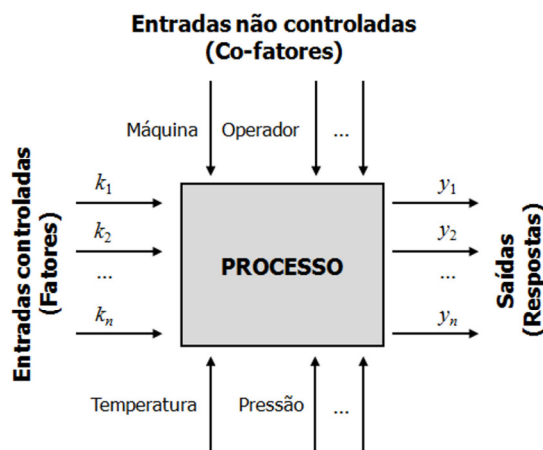
As conclusões da pesquisa são apresentadas no Capítulo 8. Ao final desse capítulo são apresentados, também, o resumo das contribuições desta pesquisa e algumas sugestões para trabalhos futuros voltados para a pesquisa desenvolvida.

2. EXPERIMENTAÇÃO E PLANEJAMENTO DE EXPERIMENTOS

2.1 INTRODUÇÃO

Muito do conhecimento sobre processos e sistemas da área científica é derivado da experimentação. Formalmente, um experimento é uma série de testes (ou execuções) conduzidos de maneira sistemática, com a finalidade de aumentar o conhecimento a respeito de um processo existente ou de explorar um novo processo ou sistema (MONTGOMERY, 2012). No âmbito científico, processos e sistemas podem ser representados a partir de modelos formados por parâmetros de entrada, também chamados *fatores*, tais como máquinas, métodos, pessoas, variáveis discretas ou contínuas e outros recursos, e saídas, isto é, *respostas*, em geral, produzidas por meio de algum processamento (Figura 1).

Figura 1 - Modelo de processo.



Fonte: (MONTGOMERY, 2012), adaptado pelo autor.

Usualmente, experimentos envolvem diversos fatores e um dos objetivos do experimentador é determinar a influência que os fatores têm sobre a resposta. A abordagem para planejar e conduzir um experimento é chamada de estratégia de experimentação. Dentre as diferentes estratégias, pode-se destacar a do melhor palpite (em Inglês, *best-guess approach*), em que o experimentador escolhe arbitrariamente um conjunto de fatores para ser testado. Essa abordagem exige razoável conhecimento teórico e técnico do experimentador. Entretanto, existem pelo menos duas desvantagens associadas à abordagem. A primeira é que resultados inesperados obtidos a partir de um palpite inicial, exigem novas combinações de fatores e novos testes e, por isso, podem requerer um longo período de tempo. Por outro lado,

resultados viáveis obtidos a partir de um palpite inicial podem determinar a parada prematura dos testes.

Uma outra estratégia de experimentação é aquela que considera o estudo isolado de um único fator por vez (em Inglês, *one-factor-at-time*). Essa abordagem procura estabelecer um ponto inicial e os intervalos de valores para cada fator e, então, variar um único fator por vez, mantendo os demais constantes. Sua principal desvantagem é não considerar a existência de interação entre os fatores (MONTGOMERY, 2012).

Uma alternativa a essas limitações consiste no estudo combinado de múltiplos fatores por meio da metodologia de Planejamento de Experimentos (em Inglês, *Design of Experiments - DOE*) (MONTGOMERY, 2012), um arcabouço de técnicas de Matemática e de Estatística que maximiza o aprendizado e consome o mínimo de recursos.

Neste capítulo são apresentados os aspectos teóricos sobre DOE com especial atenção para os Planejamentos Fatoriais e Metodologia de Superfície de Respostas, em geral, empregada na modelagem para uma relação adequada entre fatores e respostas. O capítulo apresenta, também, uma introdução a respeito de ajustes de modelos de regressão e Análise de Variância e uma breve introdução sobre outras técnicas de DOE.

2.2 PLANEJAMENTO DE EXPERIMENTOS

Em geral, experimentos científicos envolvem muitos fatores e, frequentemente, concentram-se em determinar a influência dos fatores sobre a resposta, bem como, os valores (ou configurações) para obter respostas aceitáveis ou com menor variabilidade. As técnicas de DOE permitem estabelecer uma relação de causa e efeito entre fatores e resposta, cuja representação usual é um modelo polinomial de baixa ordem na forma de (1):

$$y = \beta f(x) + \varepsilon, \quad (1)$$

em que o comportamento da resposta y é explicado em parte pelos fatores x por meio de uma função $f(x)$ associados aos coeficientes β , e, em outra parte não captada pela função, representada pelo erro experimental aleatório ε . Em (1), x são variáveis independentes, y a variável dependente e ε é o erro estatístico aleatório, ou resíduos do modelo, que segue uma distribuição normal padrão, com média 0 e desvio padrão σ . No contexto de DOE, grande

parte dos resultados são expressos na forma de modelos de regressão linear (KHURI e MUKHOPADHYAY, 2010).

Um modelo de regressão linear que contém mais de uma variável independente é chamado de modelo de regressão linear múltipla:

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \varepsilon \quad (2)$$

O modelo (2) é linear em relação aos coeficientes β_i ($i = 0, 1, 2, \dots, k$) e descreve um plano no espaço de y e x_i . O coeficiente β_0 é o intercepto do plano, enquanto que β_i são os coeficientes de regressão parciais, tal que β_i representa uma alteração da resposta média correspondente a uma unidade do i -ésimo fator x , quando os demais fatores são constantes. Este modelo (2) é chamado modelo de regressão linear de primeira ordem, pois a potência máxima das variáveis do modelo é 1. Um modelo de segunda ordem, dado por (3), acrescenta termos quadráticos, como $\beta_{ii} x_i^2$, e termos com o produto cruzado de fatores, por exemplo $\beta_{ij} x_i x_j$, em (2).

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \sum_{\substack{j=1 \\ j \neq i}}^k \beta_{ij} x_i x_j + \sum_{i=1}^k \beta_{ii} x_i^2 + \varepsilon \quad (3)$$

Este modelo (3) também é linear em relação aos coeficientes β , mas é referenciado como modelo de regressão polinomial. Os termos quadráticos e produtos de fatores presentes em (3) fazem da resposta uma superfície curvilínea. Este modelo é de segunda ordem, pois a potência máxima das variáveis do modelo é 2.

Em geral, os modelos de regressão são adotados com os propósitos de:

- Estabelecer uma relação, mesmo que aproximada, entre y e x_1, x_2, \dots, x_k , para prever a resposta sob diferentes configurações dos fatores;
- Determinar a significância dos fatores por meio de testes de hipóteses; e
- Determinar as configurações de x_1, x_2, \dots, x_k , que resultam na resposta máxima (ou mínima) sobre uma região de interesse.

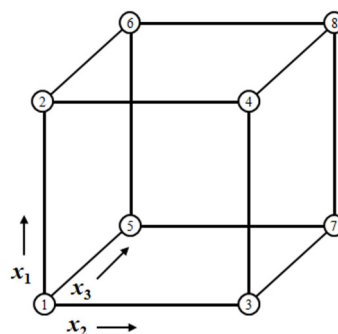
Para alcançar esses propósitos uma série de experimentos deve ser conduzida, a fim de se observar a resposta para diferentes configurações ou valores dos fatores (KHURI e MUKHOPADHYAY, 2010). Em um experimento de DOE, os fatores podem assumir diferentes valores, chamados de *níveis*. As combinações entre fatores e níveis são denominadas tratamentos. Por exemplo, um experimento composto por dois fatores, com n níveis do primeiro fator e m níveis do segundo, correspondendo a $n \times m$ tratamentos, é chamado de planejamento fatorial completo $n \times m$, no qual todas as combinações fator/nível são investigadas.

2.2.1 Planejamentos Fatoriais

Planejamentos fatoriais permitem a investigação simultânea de dois ou mais fatores com um número qualquer de níveis. Seu objetivo é identificar os fatores mais significativos para a resposta e investigar as relações ou interações que existem entre os fatores. Uma interação pode ser definida como a falha de um fator em produzir o mesmo efeito na resposta para cada nível de um outro fator (MONTGOMERY, 2012).

Existem diferentes tipos de planejamentos fatoriais, por exemplo, os fatoriais com dois ou três níveis, que são casos especiais em que os fatores são limitados a um determinado número de níveis. Para o desenvolvimento desta pesquisa os fatoriais com dois níveis receberão uma atenção especial.

Figura 2 - Representação geométrica de um fatorial completo 2^3 .



Quando os fatores estão limitados a um determinado número de níveis, como no caso de planejamentos fatoriais com dois níveis, é possível obter um número reduzido de tratamentos e investigar todos os fatores e suas interações. O esquema padrão dos fatoriais em dois níveis adota a notação "+1" e "-1" para representar os níveis alto e baixo, respectivamente, de cada

fator (BOX et al., 1978; MONTGOMERY, 2012). Quando todas as combinações fator/nível são executadas, o planejamento é denominado fatorial completo 2^k , no qual o 2 representa o número de níveis e k , o número de fatores. Um fatorial completo 2^3 com três fatores (x_1 , x_2 , e x_3) e dois níveis produz 8 tratamentos representados como vértices de um cubo (Figura 2). Na Figura 2 cada vértice representa uma combinação única entre fator e nível, denominada ponto fatorial.

Uma propriedade dos fatoriais completos 2^k é a ortogonalidade. Uma matriz ortogonal indica que o planejamento é balanceado de forma que os níveis dos fatores são ponderados igualmente. Essa propriedade é importante, pois elimina a correlação entre as estimativas dos efeitos principais⁴ e de interações⁵. Assim, a análise estatística de um planejamento fatorial 2^k é direta, uma vez que é possível estimar seus efeitos independentemente. Se o experimento não for ortogonal, seja proposital ou devido a perda acidental dos dados, a interpretação poderá não ser direta (MONTGOMERY, 2012).

Uma desvantagem dos planejamentos fatoriais é que o número de tratamentos cresce geometricamente à medida que o valor de k aumenta. Por isso, mesmo no caso de dois níveis e que o número de fatores seja pequeno, um planejamento fatorial pode tornar-se extenso rapidamente. Nesse contexto, os planejamentos fatoriais fracionados surgem como alternativa para reduzir o número de experimentos.

Um fatorial fracionado é um planejamento no qual somente um subconjunto de tratamentos presentes nos fatoriais completos é utilizado.

No caso de planejamentos fatoriais com dois níveis, uma categoria denominada fracionada $2^{(k-p)}$, em que p é um fator de redução, representa uma alternativa aos fatoriais completos, uma vez que nem todas as combinações de fator/nível são investigadas. Para a confecção de um fatorial fracionado é necessário descrever como os efeitos são confundidos com outros efeitos, ou seja, a resolução do planejamento fracionado. Quando um ou mais efeitos são confundidos, eles não podem ser estimados separadamente. Nos planejamentos fracionados $2^{(k-p)}$ as resoluções mais comuns são:

- III: os efeitos principais são confundidos com interações de 2 fatores;

⁴ Efeito principal: representa uma diferença no comportamento do fator em razão de alterações de seu nível.

⁵ Efeito de interação: Representa uma diferença no comportamento de um fator para os diferentes níveis de outro fator.

- IV: algumas interações de 2 fatores são confundidas com outras interações de 2 fatores, e os efeitos principais são confundidos com interações de 3 fatores; e
- V: as interações de 2 fatores são confundidas com interações de 3 fatores, e os efeitos principais são confundidos com interações de 4 fatores.

Por exemplo, um fatorial completo 2^3 pode ser reescrito como um fatorial fracionado $2^{(3-1)}$ com resolução III, em que todos os fatores ($k = 3$) são estudados, mas os níveis de um deles ($p = 1$) são estimados a partir das interações entre os outros dois fatores. Nessa ilustração, os níveis principais de x_3 são estimados por meio das interações entre x_1 e x_2 , resultando uma matriz experimental com 4 tratamentos (Tabela 1).

Uma desvantagem dos planejamentos fracionados consiste no fato de que, com base nos tratamentos excluídos, os efeitos de interações de altas ordens, que envolvem dois ou mais fatores, são desconsiderados.

Tabela 1 - Fatorial fracionado $2^{(3-1)}$ de resolução III.

Tratamentos	x_1	x_2	x_3
1	-1	-1	+1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	+1

2.2.2 Planejamentos Plackett-Burman

Os planejamentos Plackett-Burman (PLACKETT e BURMAN, 1946) são planejamentos com resolução III, de dois níveis. Esses planejamentos são semelhantes aos fatoriais com dois níveis, mas exigem um número menor de experimentos, principalmente, quando o número de fatores é grande. Nos planejamentos Plackett-Burman (PPB) o número de fatores deve ser sempre menor do que o número de experimentos. Por exemplo, um planejamento com 8 experimentos permite estimar os efeitos principais para até 7 fatores (Tabela 2).

A construção de uma matriz experimental de um PPB se dá a partir da permutação cíclica da primeira linha, que é padrão do PPB. A segunda linha é formada a partir do deslocamento em uma coluna da primeira linha, tal que o sinal da última coluna da primeira linha se desloca para a primeira coluna da segunda linha, o sinal da primeira coluna se desloca para a segunda e assim por diante até que todos os sinais da primeira linha sejam deslocados.

Esse procedimento, classificado como variação helicoidal, é repetido até obter uma matriz $k \times k$. Para finalizar, uma última linha é adicionada à matriz, com todos os fatores configurados no nível baixo (BERES e HAWKINS, 2001). O resultado do procedimento é uma matriz experimental com $(k+1) \times k$ tratamentos.

Tabela 2 - Esquema de um Planejamento Plackett-Burman com 7 fatores.

Tratamentos	x_1	x_2	x_3	x_4	x_5	x_6	x_7
1	+1	+1	+1	-1	+1	-1	-1
2	-1	+1	+1	+1	-1	+1	-1
3	-1	-1	+1	+1	+1	-1	+1
4	+1	-1	-1	+1	+1	+1	-1
5	-1	+1	-1	-1	+1	+1	+1
6	+1	-1	+1	-1	-1	+1	+1
7	+1	+1	-1	+1	-1	-1	+1
8	-1	-1	-1	-1	-1	-1	-1

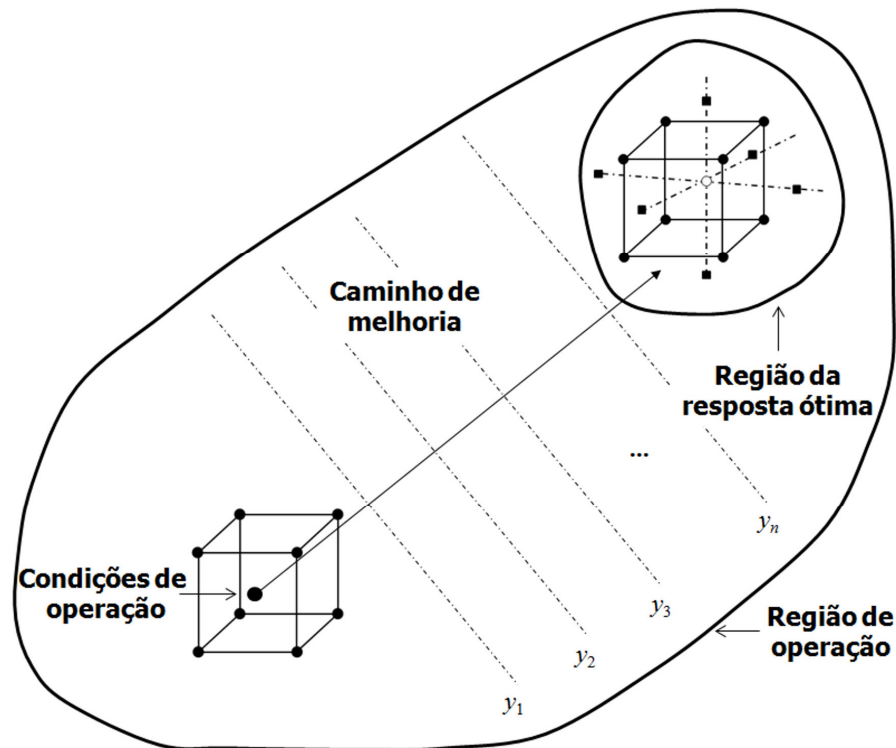
2.3 METODOLOGIA DE SUPERFÍCIE DE RESPOSTAS

A Metodologia de Superfície de Respostas (em Inglês, *Response Surface Methodology - RSM*) é um conjunto de técnicas matemáticas e estatísticas úteis para a modelagem e análise de problemas em que a resposta é influenciada por diversos fatores e o objetivo é otimizar a resposta (BRADLEY, 2007). Os estudos com a RSM, usualmente, visam:

- Compreender a topografia da superfície das respostas de um problema; e
- Encontrar a região onde ocorre a resposta ótima.

Em síntese, a RSM pode ser expressa na forma de um procedimento, cujo início se dá em um ponto remoto ao ótimo na superfície de respostas, por exemplo, as condições de operação na Figura 3, em que há pouca curvatura e um modelo de primeira ordem é apropriado. O objetivo é caminhar rápida e eficientemente até encontrar a região da resposta ótima. Nessa região, um modelo mais elaborado, por exemplo, de segunda ordem, é adequado para localizar o ótimo (MONTGOMERY, 2012).

Figura 3 - Procedimento sequencial da RSM.

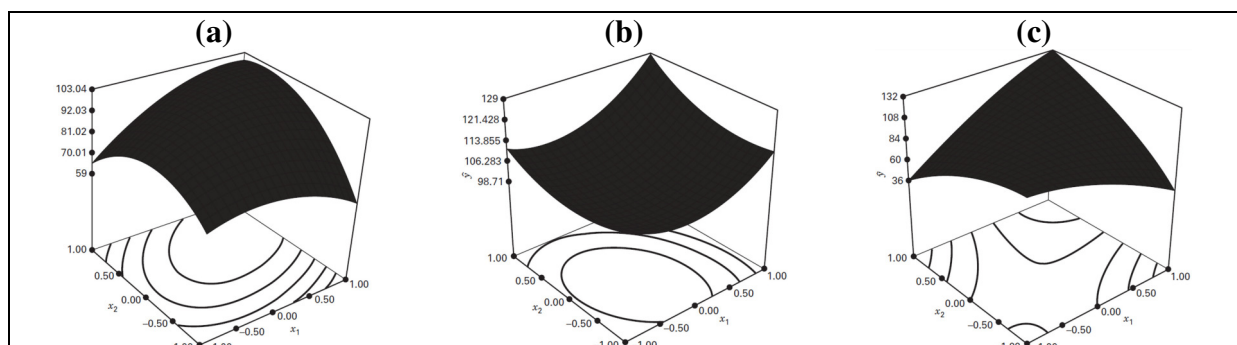


Fonte: (MONTGOMERY, 2012), adaptado pelo autor.

A RSM é muito empregada no refinamento de modelos após a determinação de fatores significativos por meio dos experimentos fatoriais, especialmente, quando há suspeita de curvatura na superfície das respostas.

Freqüentemente, a superfície de respostas é representada no espaço tridimensional como gráficos de contornos que ajudam a identificar os perfis da resposta e localizar regiões de cumes (máximo), vales (mínimo) e sem extremos (ponto de sela) (Figura 4). No entanto, quando existem mais de duas variáveis independentes, os modelos de regressão (2 e 3) são essenciais para analisar o relacionamento entre fatores e resposta.

Figura 4 - Perfis de superfícies de respostas. (a) Máximo; (b) Mínimo; (c) Ponto de sela.



Fonte: (MONTGOMERY, 2012).

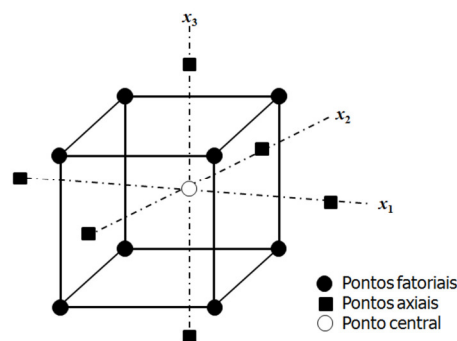
Grande parte dos problemas estudados com a RSM emprega ambos os modelos (2) e (3) a partir de uma série de experimentos, em que a resposta é analisada para cada valor (ou configuração) de fator sugerida. Uma vez que uma região promissora tenha sido encontrada, o modelo de regressão de segunda ordem (3) pode ser adequado para localizar a resposta ótima.

Um modelo de regressão de segunda ordem (3) pode ser construído eficientemente a partir de Planejamentos Centrais Compostos (em Inglês, *Central Composite Design - CCD*) (BOX e WILSON, 1951). Um CCD é constituído por um planejamento fatorial 2^k acrescido de pontos axiais⁶ e pontos centrais⁷. Na forma geral, a representação de um CCD com k fatores é resumida em:

- Pontos fatoriais: $(-1, -1, \dots, -1), (1, -1, \dots, -1), \dots, (1, 1, \dots, 1)$;
- Pontos axiais: $(-\alpha, 0, \dots, 0), (\alpha, 0, \dots, 0), (0, -\alpha, \dots, 0), (0, \alpha, \dots, 0), \dots, (0, 0, \dots, \alpha)$; e
- Pontos centrais: $(0, 0, \dots, 0), (0, 0, \dots, 0), \dots, (0, 0, \dots, 0)$.

Os pontos fatoriais contribuem para estimativas dos termos lineares e das interações entre os fatores. Os pontos centrais contêm informações sobre a curvatura da superfície. Se a curvatura é significativa, os pontos axiais adicionais permitem obter uma estimativa eficiente dos termos quadráticos. O esquema de um CCD com um fatorial completo 2^k , em que $k = 3$, e $2k$ pontos axiais é apresentado na Figura 5.

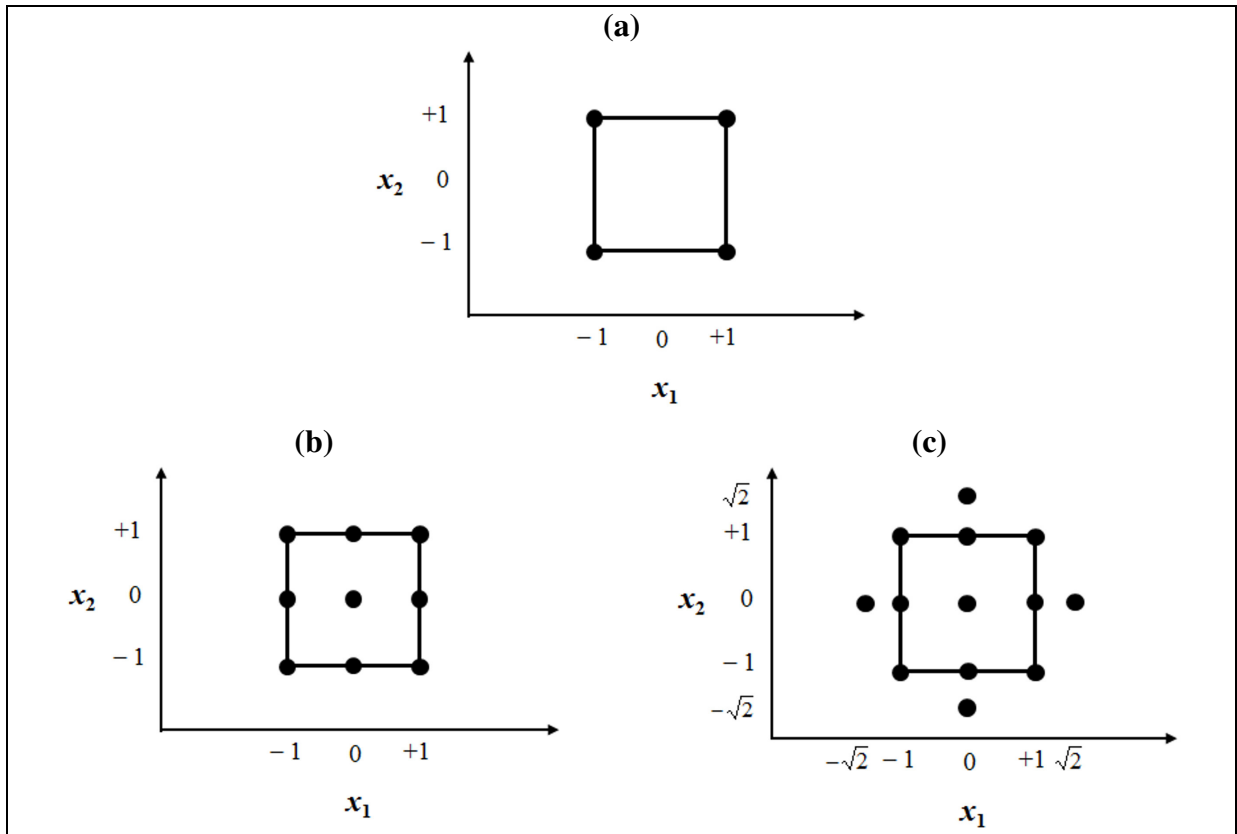
Figura 5 - Representação de um Planejamento Central Composto 2^3 .



⁶ Pontos axiais: pontos de um CCD que extrapolam os valores dos níveis (alto e baixo) dos fatores e atribuem a eles novos limites.

⁷ Pontos centrais: um ou mais pontos de um experimento fatorial, cujos valores representam o centro entre os níveis de um fator.

Figura 6 - (a) Planejamento Fatorial 2^2 ; (b) CCD para $\alpha = 1$; (c) CCD para $\alpha > 1$.



A distância entre pontos centrais e axiais em unidades é representada por α . Geralmente, o valor atribuído aos pontos axiais está no intervalo entre 1 e $k^{1/2}$. Portanto, os pontos podem se distanciar em uma unidade do centro, por exemplo $\alpha = 1$ (Figura 6-b), ou em mais unidades, por exemplo $\alpha = k^{1/2}$ (Figura 6-c). O valor preciso de α depende das propriedades do planejamento (ortogonalidade e rotação) e do número de fatores envolvidos (MONTGOMERY, 2012).

2.4 AJUSTE DE MODELOS DE REGRESSÃO

Um modelo de regressão visa explicar a relação entre variáveis independentes e variável dependente para prever os valores da variável dependente. A relação entre x (variável independente) e y (variável dependente) pode ser expressa com a notação de matriz (4) por meio de um modelo de regressão linear:

$$y = X\beta + \varepsilon, \quad (4)$$

em que y é um vetor ($n \times 1$) de observações (respostas), X é uma matriz ($n \times m+1$) correspondente aos níveis dos fatores x , β é um vetor ($m+1 \times 1$) de coeficientes de regressão e ε é o erro estatístico aleatório.

Para um modelo de primeira ordem, as n equações de duas incógnitas são expressas em termos de y , X e β como:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} \quad X = \begin{bmatrix} 1x_{11} & x_{12} & \dots & x_{1m} \\ 1x_{21} & x_{22} & \dots & x_{2m} \\ \dots & & & \\ 1x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \dots \\ \beta_m \end{bmatrix}$$

Normalmente, o ajuste de modelos de primeira ordem se dá por meio do método de mínimos quadrados (BRADLEY, 2007). A solução de mínimos quadrados é um vetor $\hat{\beta}$ que estima os coeficientes de regressão desconhecidos β . O modelo de regressão múltipla pode ser estimado com:

$$\hat{y} = \hat{\beta}_0 + \sum_{i=1}^k \hat{\beta}_i x_i, \quad (5)$$

em que \hat{y} indica a estimativa ou uma previsão do modelo. Os valores das observações y podem ser diferentes dos valores ajustados \hat{y} com (5). Essa diferença corresponde aos resíduos:

$$e = y - \hat{y} \quad (6)$$

Uma vez que o modelo de regressão linear estimado é obtido, a Análise de Variância pode ser aplicada para determinar a adequação do modelo aos dados (MONTGOMERY, 2012). Modelos de primeira ordem são adequados para superfícies planas, mas são insuficientes para superfícies com a presença de curvatura. Nessa situação, um modelo de regressão polinomial pode ser considerado para auxiliar a localização da região com a resposta ótima.

No contexto de DOE, os modelos de primeira ordem, em geral, usados nos estágios iniciais da investigação por meio de superfícies de respostas são, comumente, associados aos

planejamentos fatoriais 2^k (completos e fracionados) e PPB. Já os modelos de segunda ordem, associados aos CCD, são adequados para aproximação de superfícies curvilíneas. Nos CCD, os pontos centrais contêm informações sobre a curvatura da superfície. Se a curvatura é significativa, os pontos axiais permitem obter estimativas eficientes a respeito dos termos quadráticos. Assim como nos modelos de primeira ordem, a estimativa dos coeficientes de regressão β_i de modelos de regressão polinomial é realizada por meio do método de mínimos quadrados (BRADLEY, 2007).

2.4.1 Análise de Variância

A Análise de Variância (em Inglês, *Analysis of Variance* - ANOVA) é o método estatístico utilizado para identificar fatores significativos em modelos de regressão múltipla, comuns em DOE.

Seja um planejamento fatorial composto por dois fatores quaisquer A e B , com n e m níveis associados a cada fator, respectivamente. A ANOVA de um fatorial completo $n \times m$ é baseada na decomposição da variabilidade total dos fatores, tal que:

$$SQ_T = SQ_A + SQ_B + SQ_{AB} + SQ_E, \quad (7)$$

em que SQ_T é a soma de quadrados total, SQ_A e SQ_B são as somas de quadrados dos dois fatores A e B , SQ_{AB} é a soma de quadrados da interação $A \times B$ e SQ_E é a soma de quadrados dos resíduos. A soma de quadrados total é dada por:

$$SQ_T = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^r (y_{ijk} - \bar{y})^2, \quad (8)$$

em que y_{ijk} é uma observação, \bar{y} representa a média global de todas as observações e r corresponde ao número de vezes que cada tratamento aparece no experimento. A soma de quadrados dos fatores é obtida com (9 e 10),

$$SQ_A = mr \sum_{i=1}^n (\bar{y}_i - \bar{y})^2, \quad (9)$$

e

$$SQ_B = nr \sum_{j=1}^m (\bar{y}_j - \bar{y})^2, \quad (10)$$

onde \bar{y}_i e \bar{y}_j são as médias das observações de cada fator. A soma de quadrados da interação $A \times B$ é calculada em:

$$SQ_{AB} = r \sum_{i=1}^n \sum_{j=1}^m (\bar{y}_{AB} - \bar{y}_i - \bar{y}_j + \bar{y})^2 \quad (11)$$

Em (12) é apresentada a soma de quadrados dos resíduos.

$$SQ_E = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^r (y_{ijk} - \bar{y}_{ij})^2 \quad (12)$$

As somas de quadrados (8-12) são organizadas de maneira conveniente em uma tabela ANOVA (Tabela 3).

Tabela 3 - Esquema da Análise de Variância (ANOVA).

Fontes de Variação	<i>gl</i>	<i>SQ</i>	<i>QM</i>	<i>F</i>
A	(<i>n</i> -1)	<i>SQ_A</i>	<i>SQ_A/(a-1)</i>	<i>QM_A/QM_E</i>
B	(<i>m</i> -2)	<i>SQ_B</i>	<i>SQ_B/(b-1)</i>	<i>QM_B/QM_E</i>
AB	(<i>n</i> -1)(<i>m</i> -1)	<i>SQ_{AB}</i>	<i>SQ_{AB}/(a-1)(b-1)</i>	<i>QM_{AB}/QM_E</i>
Resíduos	<i>nm(r-1)</i>	<i>SQ_E</i>	<i>SQ_E/nm(r-1)</i>	
Total	<i>nmr-1</i>	<i>SQ_T</i>		

Na Tabela 3, as fontes de variação representam as causas de variabilidade do experimento, *gl* são os graus de liberdade⁸ associados a cada fonte de variação, *SQ* corresponde a soma de quadrados, *QM* são os quadrados médios, isto é, as somas de quadrados (dos fatores, da interação e dos resíduos) dividido pelo número de graus de liberdade correspondentes, e *F* é o valor da estatística F obtida pela razão entre *QM* (dos fatores e da interação) e *QM_E*. Em geral, sempre que $F_{obs} > F$, ao nível de significância α escolhido, o teste F é considerado significativo para rejeitar a hipótese de nulidade H_0 e afirmar que há diferenças entre os efeitos dos tratamentos. Caso contrário, H_0 é aceita indicando que há igualdade entre os efeitos dos tratamentos. Outra maneira de avaliar a

⁸ Quantidade de informações independentes em que uma estimativa é baseada.

significância da estatística F é analisar o valor- p (em Inglês, *p-value*) em que H_0 será rejeitada se $p < \alpha$.

Os pressupostos sob os quais a ANOVA é confiável são os mesmos dos testes paramétricos baseados na distribuição normal:

- Normalidade;
- Homoscedasticidade; e
- Multicolinearidade.

Em termos de violação dos pressupostos, o teste F é extremamente robusto contra os desvios da normalidade. Quanto maior o tamanho das amostras, sua aproximação da normalidade se dá automaticamente pelo teorema do limite central. A violação do pressuposto de normalidade introduz erros no nível de significância dos testes estatísticos e perda de eficiência na estimação dos efeitos. A ANOVA assume o pressuposto de homoscedasticidade das variâncias dos resíduos, que significa variância constante dos resíduos para observações diferentes. A violação da homoscedasticidade é relevante quando o tamanho das amostras é diferente. No caso, de amostras grandes com variância maior do que a variância de amostras pequenas, os testes estatísticos devem ser mais conservadores e, provavelmente, menos significativos. Vale ressaltar que a ausência de homogeneidade das variâncias é chamada heteroscedasticidade. O problema mais grave de violação de pressupostos é com relação à independência das observações. A ANOVA pressupõe que todas as observações são mutuamente independentes e que as variáveis independentes, também, são independentes umas das outras. A multicolinearidade ocorre quando as variáveis independentes estão relacionadas entre si e não são independentes umas das outras. Se a correlação é positiva, os testes estatísticos podem indicar excesso de resultados significativos (FIELD, 2009).

2.5 OUTRAS TÉCNICAS DE PLANEJAMENTOS EXPERIMENTAIS

A presente seção introduz uma breve descrição de outras técnicas de DOE muito utilizadas. Conceitos mais aprofundados e detalhes de implementações podem ser obtidos em Montgomery (2012).

Planejamentos Estatísticos de Misturas são muito comuns nas áreas em que um produto é formado por vários componentes ou ingredientes, como na química. Por exemplo, dois tipos

de variáveis podem influenciar a resposta de um sistema químico: as de processo e as de mistura. Os planejamentos fatoriais são associados às variáveis de processos. Quando variáveis de mistura estão envolvidas em uma otimização, o resultado depende da proporção desses componentes e seus níveis não podem sofrer variações sem levar em conta os outros componentes. Ou seja, misturas são sistemas cujas propriedades dependem das proporções relativas dos seus componentes e não de suas concentrações. Nos planejamentos de misturas, o somatório das proporções de seus componentes deve ser a unidade. A modelagem por meio de planejamentos de misturas consiste em ajustar um modelo matemático polinomial a uma superfície de respostas, obtida segundo um planejamento experimental específico, conhecido como planejamento Estatístico de Misturas (COSCIONE et al., 2005).

Um planejamento Quadrado Latino é apropriado para ambientes experimentais heterogêneos em que há dificuldades para efetuar medições ou para controlar simultaneamente as fontes de variabilidade. Por exemplo, em experimentos com seres vivos em que há a suspeita de diferenças sistemáticas (aprendizado, adaptação, etc.) em observações sucessivas. Um quadrado latino é uma matriz $n \times n$, em que o número de fatores é igual ao número de níveis e cada nível ocorre uma única vez em cada linha e cada coluna. No exemplo, os fatores são controlados utilizando-se dois tipos de blocos: linhas e colunas. Portanto, cada bloco terá tantas parcelas quanto forem os tratamentos (DUARTE, 1996).

Os métodos de Taguchi (TAGUCHI e KONISHI, 1987) são muito utilizados, principalmente, no contexto da experimentação industrial. De particular importância são os conceitos de:

- Função de perda de qualidade;
- Sinal e ruído; e
- Arranjos ortogonais.

A função de perda de qualidade é uma função de custo associada ao desvio do ideal. Com base na perda, a maximização do sinal e ruído resulta em minimização da função de perda de qualidade. Taguchi desenvolveu um sistema de arranjos ortogonais que permite que o número máximo de efeitos principais sejam estimados de forma imparcial, com um número mínimo de experimentos.

O objetivo dos métodos de Taguchi é minimizar a variabilidade de um produto, identificando os meios de lhe conferir robustez. Para isso, um conjunto de experimentos deve

ser realizado a fim de simular as condições específicas de produção sob o efeito de diferentes fontes de variação. A partir dos resultados é possível identificar os fatores que mais afetam a variabilidade, bem como os níveis em que deverão ser mantidos durante a produção, de modo que a variabilidade seja mínima.

D-ótimo (em Inglês, *D-optimal*) é uma forma de planejamento implementada a partir de algoritmos computacionais. Eles são particularmente úteis quando os planejamentos clássicos não se aplicam. Diferentes dos fatoriais (completos e fracionários) as matrizes de planejamento D-ótimo, em geral, são não ortogonais e seus efeitos são correlacionados. Esses tipos planejamentos são uma opção independente do tipo de modelo (primeira ordem, quadrático, cúbico, etc.) e objetivo do experimento, por exemplo, triagem, superfície de respostas, dentre outros.

3. META-HEURÍSTICAS E O PROBLEMA DE SINTONIZAÇÃO DE PARÂMETROS

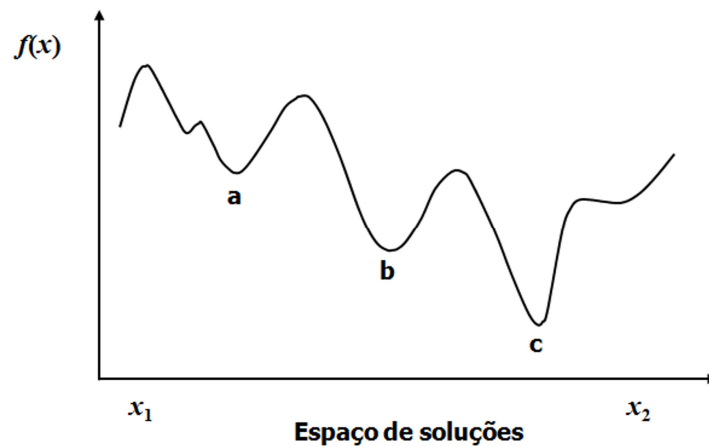
3.1 INTRODUÇÃO

Muitos problemas de otimização são da classe NP-difícil (em Inglês, *NP-hard*), intrinsecamente complexos, e demandam esforço computacional demasiado em sua resolução. Os métodos exatos (por exemplo, enumeração, *branch and bound*, programação linear e inteira, dentre outros) são capazes de encontrar soluções ótimas, mas quando aplicados na resolução de problemas do mundo real (com grandes dimensões, muitas variáveis e restrições) podem ser ineficientes e demorados.

Em contraste, as meta-heurísticas são estratégias de alto nível que direcionam outros algoritmos, por exemplo, pesquisa local, métodos exatos e heurísticas, na exploração do espaço de soluções de problemas de otimização (Figura 7). Apesar da falta de fundamentação teórica (BLUM e ROLI, 2003), sua simplicidade atrai pesquisadores e profissionais, e muitos resultados na literatura sugerem que as meta-heurísticas são técnicas no estado da arte para problemas em que não há um algoritmo específico e eficiente.

Por meio de meta-heurísticas é possível superar soluções ótimas locais de problemas complexos e encontrar soluções de qualidade razoável, com poucas modificações de parâmetros em cada caso específico. Por exemplo, a Figura 7 ilustra o espaço de soluções de um problema, em que os pontos *a* e *b* são mínimos locais e o ponto *c* é o mínimo global entre os pontos x_1 e x_2 .

Figura 7 - Ilustração de um espaço de soluções.



No entanto, quando há interesse em soluções próximas do ótimo, ou mesmo na solução ótima, a sintonização de parâmetros de meta-heurísticas ou, simplesmente, sintonização de meta-heurísticas, se torna uma tarefa relevante no contexto de desenvolvimento, avaliação e aplicação desses algoritmos.

Embora a sintonização de meta-heurísticas tenha sua importância reconhecida na literatura, somente no final da última década foi proposta uma metodologia robusta e bem estabelecida para auxiliar a configuração desses algoritmos (BIRATTARI et al., 2002; BIRATTARI et al., 2009). Entretanto, ainda é possível encontrar o uso de abordagens informais, de tentativa e erro, e a utilização de conhecimentos empíricos para a sintonização de meta-heurísticas. Essas abordagens, além de extremamente entediadas, são pouco confiáveis, de baixa replicação, propensas a erros de configuração e custosas.

O presente capítulo descreve o funcionamento básico de algumas meta-heurísticas, com destaque para aquelas usadas no desenvolvimento desta pesquisa como, por exemplo, Recozimento Simulado, Algoritmos Genéticos e Busca em Vizinhança Variável. O capítulo aborda, também, o problema de sintonização de parâmetros de meta-heurísticas.

3.2 META-HEURÍSTICAS

A palavra heurística vem grego *heuriskein*, que significa "a arte de inventar ou descobrir", enquanto que o prefixo *meta*, significa "em nível superior". O termo meta-heurística foi, primeiramente, introduzido em Glover (1986) como estratégias generalistas que empregam estruturas algorítmicas na concepção de técnicas capazes de encontrar soluções aproximadas para problemas de otimização, de maneira eficiente e com precisão. No entanto, uma rápida pesquisa na literatura revela outras definições (GLOVER et al., 2003; BLUM e ROLI, 2003; ÓLAFSSON, 2006; TALBI, 2009; DOBSLAW, 2010; ARIN et al., 2011, e muitos outros) e algumas propriedades fundamentais para caracterizá-las:

- Meta-heurísticas são estratégias para guiar o processo de pesquisa;
- Elas exploram o espaço de buscas de maneira eficiente a fim de encontrar soluções próximas do ótimo, ou algumas vezes o ótimo;
- As meta-heurísticas são constituídas por ampla variedade de técnicas;
- Seus conceitos básicos permitem descrevê-las de maneira abstrata;
- Elas não são direcionadas a problemas específicos;

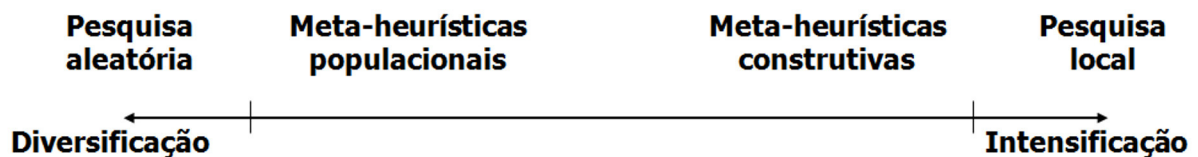
- Os algoritmos meta-heurísticos são aproximativos e, tipicamente, não determinísticos; e
- Eles podem incorporar mecanismos para evitar o confinamento em áreas do espaço de buscas e alguma forma de memória para orientar o processo de pesquisa.

Resumidamente, as meta-heurísticas são técnicas de alto nível que conduzem outros métodos na exploração de problemas complexos.

Ao contrário dos métodos exatos, que garantem a solução ótima, por meio das meta-heurísticas é possível resolver problemas complexos e encontrar soluções de qualidade razoável em tempo aceitável para as necessidades da aplicação. No entanto, não há a garantia de encontrar a solução ótima (TALBI 2009; SÖRENSEN, 2015).

O uso de meta-heurísticas permite escapar de extremos (mínimo ou máximo) locais e seguir a exploração no espaço de soluções a procura de soluções ainda melhores devido a dois critérios fundamentais, porém contraditórios: diversificação (em Inglês, *exploration*) e intensificação (em Inglês, *exploitation*). Diversificação se refere à exploração de todo o espaço de soluções a fim de evitar o confinamento em certas áreas, enquanto que intensificação, se refere à busca focada ou em profundidade em áreas promissoras do espaço de soluções, isto é, aquelas que apresentam boas soluções. Se, por um lado a diversificação é importante para identificar rapidamente as áreas com soluções de alta qualidade, por outro, a intensificação evita perda de tempo em áreas que já foram exploradas ou que não fornecem soluções de qualidade. Grosso modo, a noção de diversificação se refere a estratégias de curto prazo (Figura 8), frequentemente, vinculadas à aleatoriedade, enquanto que intensificação são estratégias de médio e longo prazos com base no uso de algum mecanismo de memória (Figura 8).

Figura 8 - Critérios de diversificação e intensificação.



Fonte: (BLUM e ROLI, 2003), adaptado pelo autor.

Algumas vezes, esses conceitos podem ser facilmente identificados nos algoritmos meta-heurísticos, como no caso da Busca Tabu (GLOVER e LAGUNA, 1997) em que a ideia básica é a escolha de vizinhos a partir de uma ou mais listas Tabu. A restrição no conjunto de vizinhos tem efeito de diversificação sobre a pesquisa, enquanto que, a escolha do melhor vizinho de um conjunto restrito tem efeito de intensificação. O equilíbrio entre diversificação e intensificação varia de acordo com a extensão da lista Tabu, de modo que listas mais curtas resultam em menor influência da diversificação, ao passo que listas mais longas resultam numa influência maior. Um outro exemplo, na meta-heurística Recozimento Simulado (KIRKPATRICK et al., 1983; CERNY, 1985) o critério de aceitação de uma solução, guiado pela taxa de arrefecimento da temperatura e por uma função objetivo, envolve uma quantidade aleatória. A diminuição da temperatura conduz o sistema da diversificação para intensificação levando-o à convergência. Contudo, nos Algoritmos Genéticos (HOLLAND, 1975) esses conceitos nem sempre são tão evidentes. Um exemplo de diversificação é o operador de seleção que, em geral, envolve alguma aleatoriedade, por exemplo, proporcionalidade da elite, torneios, etc. O equilíbrio entre diversificação e intensificação depende das probabilidades de seleção, isto é, se as diferenças são elevadas, a intensificação é mais acentuada, caso contrário, mais suave (BLUM e ROLI, 2003).

Em síntese, toda meta-heurística deve ser concebida com o objetivo de explorar efetiva e eficazmente o espaço de buscas. As buscas devem ser inteligentes o suficiente para localizar áreas com soluções de alta qualidade e, quando necessário, se mover para áreas inexploradas. Portanto, um dos grandes desafios na concepção de meta-heurísticas é o equilíbrio entre intensificação e diversificação (BIONDI NETO et al., 2004).

Um algoritmo meta-heurístico pode ser classificado e descrito de diversas maneiras e, dependendo de suas características, várias classificações são possíveis. Intuitivamente, é possível classificá-los com base na origem do algoritmo, ou seja, aqueles inspirados na natureza, tais como Algoritmos Genéticos e Colônia de Formigas, e outros não inspirados na natureza, por exemplo, Busca Tabu e Busca Local Iterativa (STÜTZLE, 1999). Entretanto, nem sempre essa classificação é significativa, podendo gerar dúvidas do tipo: Busca Tabu é inspirado na natureza? Geralmente, uma meta-heurística é inspirada na natureza, quando seus métodos de pesquisa simulam aspectos do comportamento de seres vivos. Uma outra forma de classificá-los é com base no número de soluções. Um método construtivo ou de trajetória parte de um conjunto solução vazio e segue acrescentando elementos a esse conjunto, até obter uma solução viável para o problema. Já, um método de base populacional parte de um conjunto de soluções, denominado população inicial, e procura uma solução de melhor

qualidade a partir de modificações nos elementos da população (BLUM e ROLI, 2003; BIONDI NETO et al., 2004).

Independente do método (de trajetória ou populacional) a estrutura básica de uma meta-heurística é a mesma. Ou seja, suponha que S' , no método construtivo, é um único ponto no espaço de soluções S , tal que $S' \in S$. No método populacional, considere S' um conjunto de soluções, de modo que $S' \subseteq S$. Seja uma solução candidata ou conjunto de soluções S'' selecionado ao acaso na vizinhança de S' para validações. As validações, em geral, envolvem cálculos, estimativas de desempenho e, em alguns casos, comparações de S'' com S' . Baseado nos resultados, a solução candidata pode ser aceita, de modo que S' passa a ser S'' , ou rejeitada. O enunciado generalista se aplica à meta-heurísticas de trajetória ou populacionais sem revelar os detalhes da pesquisa, que pode ser adaptada a partir de algoritmos de pesquisa local, métodos exatos ou heurísticas, de acordo com as características do problema (ÓLAFSSON, 2006).

3.2.1 Busca Local

A Busca Local (em Inglês, *Local Search* - LS) é um dos mais simples e antigos métodos de otimização (AARTS e LENSTRA, 1997). De fácil implementação, a LS compõe a estratégia de muitas meta-heurísticas com o propósito de intensificar a busca em áreas promissoras do espaço de soluções e obter as melhores soluções dentro dessas áreas.

Figura 9 - Pseudocódigo de um algoritmo de LS.

<p>Entrada: S_0 // <i>Solução inicial</i></p> <p>Saída : S^* // <i>Solução ótima local</i></p> <ol style="list-style-type: none"> 1. $S^* \leftarrow S_0$; 2. Repetir 3. // <i>Gerar vizinhos da solução atual</i> 4. $S' \leftarrow \text{criarVizinho}(S^*)$; 5. // <i>Testar a qualidade da solução vizinha (S')</i> 6. Se $\text{qualidade}(S') < \text{qualidade}(S^*)$ Então 7. $S^* \leftarrow S'$; // <i>atualizar solução ótima</i> 8. Fim Se; 9. Até atingir um critério de parada; 10. Retornar S^*;
--

Tipicamente, uma solução qualquer possui uma vizinhança com muitas soluções candidatas. Em linhas gerais, a LS se move iterativamente de solução em solução no espaço de soluções a partir de pequenas alterações na solução atual.

Normalmente, um algoritmo de LS (Figura 9) começa com uma solução inicial e cada iteração visa substituí-la por uma solução vizinha de melhor qualidade. O algoritmo termina quando todos os vizinhos são piores do que a solução atual, indicando que um extremo local, por exemplo o mínimo, foi encontrado. O desempenho desses algoritmos é fortemente influenciado pelas definições da solução inicial, tamanho da vizinhança e estratégias de pesquisa na vizinhança da solução atual.

As grandes desvantagens dos algoritmos de LS são o fato de convergirem para extremos (mínimo ou máximo) locais e, em geral, o custo computacional. Por isso, esses algoritmos são considerados insatisfatórios na resolução de problemas com muitos extremos locais ou quando a qualidade das soluções locais é muito semelhante, como ocorre no caso dos problemas de otimização combinatória (TALBI, 2009). No entanto, geralmente, a LS é parte integrante de estratégias meta-heurísticas na busca por soluções otimizadas.

3.2.2 Recozimento Simulado

O algoritmo de Recozimento Simulado (em Inglês, *Simulated Annealing* - SA) é um método probabilístico proposto em Kirkpatrick et al. (1983) e Cerny (1985) com a finalidade de encontrar o extremo global de uma função objetivo com inúmeros extremos locais. O SA é um dos métodos meta-heurísticos mais antigos e um dos primeiros algoritmos a adotar uma estratégia explícita para extremos locais.

Largamente aplicado na resolução de problemas de otimização combinatória, o método empregado no SA constitui uma analogia à termodinâmica, especialmente, no modo como são realizados o congelamento e a cristalização de líquidos ou o resfriamento e o recozimento de metais. Isto é, uma vez expostas a altas temperaturas, as moléculas movem-se livremente umas em direção as outras. Se o resfriamento é lento, essa mobilidade termal é perdida, os átomos são alinhados e formam um cristal homogêneo, que representa o estado de energia mínima do sistema. Caso contrário, se o resfriamento ocorre rapidamente, o estado de cristalização não é atingido, mas sim um estado denominado metaestável, no qual a energia é ligeiramente mais elevada (PRESS et al., 1986). O Quadro 1 ilustra a analogia entre os sistemas físicos e o SA.

Quadro 1 - Analogia entre sistema físico e o algoritmo SA.

Sistema Físico	Algoritmo SA
Estado do sistema	Solução
Energia	Função objetivo
Estado fundamental	Solução ótima global
Estado metaestável	Solução ótima local
Temperatura	Parâmetro T

Fonte: (TALBI, 2009), adaptado pelo autor.

A essência do SA é o resfriamento lento de um material, de modo que o produto final se torne uma massa homogênea ao atingir uma configuração mínima de energia (BERTSIMAS e TSITSIKLIS, 1993).

O método é baseado na distribuição de probabilidades de Boltzmann (13),

$$Prob(E) \sim \exp(-E/kT) \quad (13)$$

que expressa a ideia de que um sistema em equilíbrio térmico na temperatura T tem a sua energia E probabilisticamente distribuída entre diferentes estados de E . Ou seja, mesmo sob baixa temperatura, existe uma chance, ainda que pequena, do sistema sair de um extremo de energia local e encontrar um estado melhor, que pode representar o extremo global. A constante de Boltzmann (k) é a quantidade que relaciona temperatura com energia.

Sucintamente, o algoritmo permite a execução de movimentos, que em algumas situações resultam em soluções de pior qualidade, visando escapar dos extremos locais. A probabilidade de executar esses movimentos (de piora) é diminuída durante a evolução do processo de pesquisa.

Em 1953, Metropolis et al. (1953) primeiramente incorporaram esses princípios na forma de cálculo numérico. O sistema termodinâmico simulado foi expresso por meio de mudanças entre os estados de energia com probabilidade $p = \exp[-(E_2 - E_1)/kT]$. Quando $E_2 < E_1$ a alteração de estados é determinada arbitrariamente a partir de um valor atribuído à probabilidade p . Esse esquema de, frequentemente, caminhar no sentido de um extremo mínimo e, algumas vezes, caminhar no sentido contrário, passou a ser conhecido como algoritmo Metropolis (PRESS et al., 1986).

Figura 10 - Pseudocódigo da meta-heurística SA.

```

Entrada:  $S_0, T_{max}$  // Solução inicial e temperatura, inicialmente alta
Saída :  $S^*$  // Melhor solução encontrada
1.  $S \leftarrow S_0$ ; // Solução atual
2.  $S^* \leftarrow S$ ;
3.  $T \leftarrow T_{max}$ ;
4. Repetir
5. // Ciclos em uma mesma temperatura
6. Repetir
7.  $S' \leftarrow \text{criarVizinho}(S)$ ;
8. // Testar a qualidade da solução vizinha ( $S'$ )
9. Se  $\text{qualidade}(S') > \text{qualidade}(S)$  Então
10.  $S \leftarrow S'$ ;
11. // Testar a qualidade da solução atual ( $S$ )
12. Se  $\text{qualidade}(S) > \text{qualidade}(S^*)$  Então
13.  $S^* \leftarrow S$ ; // atualizar a melhor solução
14. Fim Se;
15. Senão
16. // Atualizar a solução atual com certa probabilidade
17.  $N \leftarrow$  um número aleatório entre (0, 1];
18. Se  $N > \exp((\text{qualidade}(S') - \text{qualidade}(S))/T)$  Então
19.  $S \leftarrow S'$ ;
20. Fim Se;
21. Até temperatura atingir a condição de equilíbrio;
22.  $T \leftarrow \text{diminuir}(T)$ ;
23. Até atingir um critério de parada;
24. Retornar  $S^*$ ;

```

A estrutura geral da meta-heurística SA é ilustrada por meio de um pseudocódigo na Figura 10. Inicialmente, T (temperatura) recebe um valor elevado T_{max} , cuja finalidade é aumentar a probabilidade de aceitar uma condição de piora. O procedimento segue em ciclo, e S' (solução candidata qualquer) passa a ser avaliada em relação a S (solução candidata inicial). Se S' apresenta melhor qualidade, então S será sempre substituída por S' . Caso contrário, S pode ser atualizada de acordo com certa probabilidade, geralmente com base na distribuição de Boltzmann (13). Ou seja, se S' é muito pior do que S , a probabilidade é próxima de 0 (zero), ao passo que se eles (S' e S) forem semelhantes, a probabilidade é próxima de 1.

Assim, se S' não for muito pior que S , então S' será aceito com probabilidade razoável. Essa característica permite que o SA, algumas vezes, aceite soluções de pior qualidade.

O critério de parada da meta-heurística SA (Figura 10), frequentemente, considera: S^* igual a melhor solução conhecida, tempo de execução maior do que um limite máximo ou T melhor do que um limite mínimo.

O desempenho da meta-heurística SA é fortemente influenciado por um conjunto de parâmetros. Os altos valores da temperatura nos estágios iniciais aumentam a probabilidade de aceitação de uma solução de baixa qualidade. Além da temperatura inicial, é importante definir o número de iterações realizadas em uma mesma temperatura, bem como o seu arrefecimento ou taxa de decaimento. Normalmente, o arrefecimento da temperatura se dá de maneira constante a uma taxa α pré-definida (14), de modo que quanto mais lento, maior a exploração do espaço de soluções (BLUM e ROLI, 2003; TALBI, 2009):

$$T_{t+1} = \alpha T_t, \quad (14)$$

onde t é um instante de tempo ou uma iteração do algoritmo.

3.2.3 Algoritmo Genético

O Algoritmo Genético (em Inglês, *Genetic Algorithm* - GA) é um algoritmo de otimização numérica inspirado no processo de seleção natural e na genética (COLEY, 1999). Esses algoritmos são de base populacional e caracterizam um modelo computacional do processo evolutivo (BLUM e ROLI, 2003; TALBI, 2009).

Os GA operam sobre uma população (conjunto de soluções de um problema) e fornecem uma maneira intuitiva para explorar o espaço de soluções (MITCHELL, 1998). Originalmente, o GA proposto em Holland (1975) foi inspirado no princípio de sobrevivência enunciado a partir da teoria da evolução das espécies de Darwin de 1858.

Um típico GA utiliza um conjunto de operadores para (i) selecionar indivíduos de uma população, com base em sua aptidão, (ii) recombinar e (iii) modificar esses indivíduos selecionados, para produzir novas gerações com indivíduos mais adaptados. Sucintamente, a ideia dos operadores é transformar uma população através de gerações sucessivas, mantendo as características de adaptação das gerações ascendentes.

Figura 11 - Pseudocódigo da meta-heurística GA.

```

Entrada:  $\mu$ ,  $N$ ,  $p_c$ ,  $p_m$  // Tamanho da população, número de gerações e taxas de cruzamento e mutação
Saída :  $P^*$  // Melhor solução encontrada
1.  $P \leftarrow \{\}$ ; // População inicial
2.  $P^* \leftarrow ()$ ;
3. Para  $\mu$  vezes Faça
4.    $P \leftarrow P \cup \{\text{novο indivíduo escolhido ao acaso}\}$ ;
5. Fim Para;
6.  $g \leftarrow 0$ ;
7. Para  $g < N$  Faça
8.   Para  $p' \in P$  Faça
9.     // Testar a aptidão de cada indivíduo ( $p'$ )
10.    Se  $\text{aptidão}(p') > \text{aptidão}(P^*)$  Então
11.       $P^* \leftarrow p'$ ; // atualizar melhor solução
12.    Fim Se;
13.  Fim Para;
14. // Criar uma nova população de descendentes
15.  $Q \leftarrow \{\}$ ;
16. Para  $\mu/2$  vezes Faça
17. // Aplicar operadores (seleção, cruzamento e mutação)
18.    $P' \leftarrow \text{selecionar}(P)$ ;
19.    $P'' \leftarrow \text{selecionar}(P)$ ;
20.    $(C', C'') \leftarrow \text{cruzamento}(P', P'', p_c)$ ;
21.    $(M', M'') \leftarrow \text{mutação}(C', C'', p_m)$ ;
22.    $Q \leftarrow Q \cup \{M', M''\}$ ;
23. Fim Para;
24.  $P \leftarrow Q$ ; // Nova geração
25.  $g \leftarrow g + 1$ ;
26. Fim Para;
27. Retornar  $P^*$ ;

```

Em um GA, cada iteração corresponde a uma nova geração. Seu princípio básico de funcionamento consiste em aplicar operadores (seleção, cruzamento e mutação) nos indivíduos da população a cada geração. O processo é interrompido quando satisfaz alguma condição de parada (GLOVER et al., 2003). A estrutura geral de um GA é ilustrada no pseudocódigo da Figura 11. Inicialmente, uma população de indivíduos é criada ao acaso.

Durante o processo evolutivo, a aptidão de cada indivíduo da população é avaliada quanto a sua capacidade de se adaptar ao (meio-)ambiente. Alguns membros da população são mantidos pelo operador de seleção, mas suas características devem variar a partir de cruzamentos e mutações a fim de gerar novos indivíduos para as gerações descendentes. Tipicamente, o processo de reprodução é repetido até que o número máximo de gerações seja excedido.

No GA, uma população é uma coleção de soluções candidatas arbitrariamente distribuídas no espaço de soluções. Ao longo das gerações, novos membros são incluídos na população, enquanto outros são excluídos. Os membros da população são indivíduos que, no GA, representam uma solução única. Os indivíduos são codificados na forma de cromossomos, em que a posição de um gene é denominada *locus* e seu valor é chamado de alelo. A Figura 12 apresenta uma representação ilustrativa dos indivíduos de uma população, em que cada linha da matriz corresponde a um cromossomo e cada coluna é um gene. A aptidão de um indivíduo é medida em termos de quanto uma solução é boa em relação à população. Quanto melhor uma solução, maior sua aptidão (THEDE, 2004).

Figura 12 - Representação da população no GA.

	1	2	3	4	5	6	7	8	9	10
a	1	0	1	1	0	1	1	0	1	1
b	1	0	1	0	1	1	0	1	0	1
c	0	1	1	1	0	1	0	1	1	1
d	0	1	0	0	1	0	1	0	1	0
e	0	1	1	1	1	0	1	0	0	0
f	0	0	1	0	0	1	0	1	1	0
g	1	0	1	1	1	0	1	0	1	1
h	1	0	0	1	1	1	0	1	0	1
i	0	0	1	1	0	0	1	0	1	0
j	0	0	1	1	0	1	0	1	1	1

Fonte: (MITCHELL, 1998), adaptado pelo autor.

A escolha dos indivíduos para compor as próximas gerações é realizada pelo operador de seleção. Geralmente, o processo de seleção aplica estratégias intuitivas semelhantes à seleção natural. Uma estratégia denominada roleta, consiste em selecionar n indivíduos de uma lista com m progenitores. Os progenitores são selecionados proporcionalmente à sua aptidão, de modo que indivíduos com maior aptidão podem ser selecionados mais vezes. Por

outro lado, na estratégia elitista, n indivíduos com maior aptidão prevalecem sobre os menos adaptados para compor as próximas gerações. Na seleção por torneio, um subconjunto de indivíduos formados por k de uma população de tamanho n é sorteado ao acaso. Os indivíduos com maior aptidão são selecionados para decidir quais irão se reproduzir. Por exemplo, quando $k = 2$, dois indivíduos competem entre si e o melhor, aquele com maior aptidão, torna-se um dos pais. A seleção por torneio permite a escolha de indivíduos menos tendenciosos, uma vez que não considera a classificação do indivíduo na população (EIBEN e SMITH, 2003).

O operador de recombinação gênica ou cruzamento é responsável pela a troca genética entre dois indivíduos de maneira semelhante ao que ocorre com organismos naturais. O processo de cruzamento consiste em escolher pares de indivíduos selecionados pelo processo de seleção e trocar arbitrariamente as características (*alelos*) entre eles para obtenção de novos indivíduos descendentes (Figura 13). Por exemplo, um processo denominado corte em 1-ponto seleciona ao acaso um ponto de corte nos pares de cromossomos, dividindo-os em duas partições, uma à direita e outra à esquerda do corte. Os indivíduos descendentes são formados pela junção da partição à esquerda (direita) de um progenitor com a partição à direita (esquerda) do outro progenitor. Outras estratégias clássicas de fácil implementação computacional são:

- Cruzamento com corte em n -pontos; e
- Cruzamento com corte uniforme.

Figura 13 - Processo de cruzamento com corte em 1-ponto.

(a) Progenitores.

	1	2	3	4	5	6	7	8	9	10
a	0	0	0	1	0	1	1	0	1	1
b	1	1	1	0	1	1	0	1	0	1

(b) Descendentes.

	1	2	3	4	5	6	7	8	9	10
a	0	0	0	0	1	1	0	1	0	1
b	1	1	1	1	0	1	1	0	1	1

Fonte: (MITCHELL, 1998), adaptado pelo autor.

O processo de cruzamento com corte em n -pontos ($n > 1$) consiste em dividir os pares de cromossomos em n partições. Os indivíduos descendentes são formados pela junção das partições de um progenitor com as partições do outro progenitor. Já no cruzamento uniforme, cada gene é associado aos valores da distribuição uniforme. Os genes do primeiro indivíduo com valores $\alpha < 0,5$, por exemplo, são oriundos de um progenitor. Caso contrário, se $\alpha \geq 0,5$, do outro progenitor. O segundo indivíduo é gerado pelo mapeamento inverso.

Além dos processos apresentados anteriormente, existem também cruzamentos com o uso de múltiplos progenitores, baseadas na sequência dos alelos ou em operações numéricas sobre os alelos (EIBEN E SMITH, 2003).

No mundo real, vários processos podem causar modificações gênicas, sendo o mais simples conhecido como erro de replicação (COLEY, 1999). Nos GA, o operador de modificação gênica ou mutação tem a finalidade de promover a variabilidade ou alterações de estruturas nos indivíduos de uma população. A partir da mutação, os indivíduos descendentes herdam algumas características de seus progenitores, embora não sejam idênticos (Figura 14). O objetivo das mutações é simular o processo de evolução e procurar obter um aumento da aptidão em novas gerações. Deste modo, alguns indivíduos de uma população podem ser transformados para compor uma geração descendente

Figura 14 - Processo de mutação.

(a) Progenitor.

	1	2	3	4	5	6	7	8	9	10
a	0	0	0	1	0	1	1	0	1	1

(b) Descendente.

	1	2	3	4	5	6	7	8	9	10
a	0	1	0	0	1	1	0	1	0	1

Fonte: (MITCHELL, 1998), adaptado pelo autor.

O desempenho do GA é fortemente influenciado por um conjunto de parâmetros. Tipicamente, o número de gerações, bem como, as taxas de cruzamento e mutação influenciam o algoritmo. Em geral, os indivíduos são recombinaados a uma taxa entre 0,4 e 0,9. Por exemplo, se a taxa for fixada em 0,5, então metade da população será formada a partir da seleção e cruzamento. Sem cruzamento, a aptidão média da população deve aumentar até se igualar com a taxa de aptidão do melhor indivíduo. A partir desse ponto, só poderá haver melhoria por meio de mutação. A taxa de mutação é da ordem de 0,001.

Entretanto, esse valor pode variar de acordo com o problema sob análise. Uma maneira de defini-la é usar $1/L$ ou $1/(LN^{1/2})$, em que L é o comprimento de um indivíduo e N o tamanho da população (COLEY, 1999).

3.2.4 Busca em Vizinhaça Variável

O algoritmo de Busca em Vizinhaça Variável (em Inglês, *Variable Neighborhood Search* - VNS) é um método meta-heurístico de trajetória largamente aplicado na resolução de problemas de otimização combinatória e otimização global. Originalmente, proposto em Hansen e Mladenovic (1999 e 2001), o VNS aplica uma estratégia baseada em mudanças dinâmicas na estrutura de vizinhaça de uma solução candidata. Diferente de outros algoritmos de trajetória, o VNS não segue uma trajetória única, mas explora vizinhos cada vez mais distantes de uma solução candidata e usa uma pesquisa local para obter ótimos locais a partir desses vizinhos.

Figura 15 - Pseudocódigo da meta-heurística VNS.

<p>Entrada: S_0, K_{\max} // <i>Solução inicial e número máximo de estruturas de vizinhaça</i></p> <p>Saída : S^* // <i>Melhor solução encontrada</i></p> <ol style="list-style-type: none"> 1. $S^* \leftarrow S_0;$ 2. Repetir 3. $k \leftarrow 1;$ 4. Para $k \leq K_{\max}$ Faça 5. $S' \leftarrow \text{perturbar}(S^*, k);$ 6. $S'' \leftarrow \text{pesquisaLocal}(S');$ 7. // <i>Testar a qualidade da solução candidata (S'')</i> 8. Se $\text{qualidade}(S'') > \text{qualidade}(S^*)$ Então 9. $S^* \leftarrow S'';$ // <i>atualizar melhor solução</i> 10. $k \leftarrow k + 1;$ 11. Senão 12. $k \leftarrow k + 1;$ 13. Fim Se; 14. Fim Para; 15. Até atingir um critério de parada; 16. Retornar $S^*;$

O princípio do VNS assume a existência de K vizinhos (soluções candidatas) em relação a uma solução do espaço de soluções. A ideia básica é explorar sucessivamente um conjunto pré-definido de candidatos para localizar a solução ótima. A exploração se dá tanto por amostragem, como sistematicamente, de modo que pode resultar em diferentes extremos locais (TALBI, 2009).

A estrutura geral de uma meta-heurística VNS básica é ilustrada no pseudocódigo da Figura 15. Cada iteração possui três estágios: perturbação, pesquisa local e movimentação. A cada iteração a solução atual é perturbada até K vezes com a finalidade de gerar uma solução temporária. A solução temporária é o ponto de partida para a pesquisa local, cujo resultado é uma solução candidata. A solução ótima é atualizada, somente se a solução candidata apresentar melhor qualidade. Nesse caso, o procedimento é reiniciado ($K = 1$) e a perturbação se dá na vizinhança da nova solução ótima. Caso contrário, o algoritmo avança e uma nova solução candidata em uma vizinhança um pouco mais distante ($K+1$) da solução atual será considerada e analisada. O algoritmo segue até atingir um critério de parada.

Resumidamente, a estratégia do VNS consiste em perturbar uma solução inicial com a finalidade de proporcionar um bom ponto de partida para a pesquisa local. O resultado deve ser um extremo local próximo de uma solução candidata, não muito distante da melhor solução encontrada.

O processo de perturbação em caso de ausência de melhora da solução inicial corresponde à diversificação. Em particular, se não há melhora na solução inicial e o nível de perturbação é incrementado, a diversificação ocorre progressivamente.

A eficácia do conceito de *mudanças dinâmicas na estrutura de vizinhança* é justificada, pois uma vizinhança de baixa qualidade em uma área do espaço de soluções pode se revelar boa em outra área. Além disso, uma solução extrema local, nem sempre é extrema em relação à outra solução candidata. Este conceito é conhecido como "*One operator, one landscape*" (JONES, 1995a; JONES, 1995b).

O algoritmo VNS é muito geral, por isso é comum encontrar variações desse algoritmo com pequenas modificações, por exemplo, na maneira como o problema é tratado (*Variable Neighborhood Decomposition Search*) e no modo como uma solução é aceita (*Skewed VNS*). Mais detalhes sobre a implementação das variações do VNS podem ser obtidos em Blum e Roli (2003) e Talbi (2009).

3.3 O PROBLEMA DA SINTONIZAÇÃO DE PARÂMETROS

A grande maioria dos algoritmos meta-heurísticos possui um conjunto de parâmetros dos quais é extremamente dependente. Usualmente, esses parâmetros não são genéricos e exigem uma configuração específica de acordo com o problema ou mesmo um exemplar do problema sob análise. Portanto, nem sempre é simples definir *a priori* qual a configuração ideal para uma meta-heurística resolver um determinado problema.

A sintonização de parâmetros de meta-heurísticas pode permitir mais flexibilidade e robustez aos algoritmos, mas requer uma estratégia cuidadosa. Em grande parte das pesquisas que envolvem meta-heurísticas, a sintonização é realizada por meio de tentativa e erro (BIRATTARI, 2009). Essa abordagem apresenta alguns inconvenientes. Do ponto de vista prático, pode implicar em demora para obter boas soluções. Do ponto de vista acadêmico, essa abordagem pode invalidar conclusões sobre experimentos que comparam diferentes algoritmos.

É evidente que, como as meta-heurísticas são sensíveis à configuração de seus parâmetros, uma sintonização cuidadosa, geralmente, melhora o desempenho desses algoritmos de maneira significativa.

A sintonização de meta-heurísticas é, por si só, um problema de otimização, em que o objetivo é otimizar um algoritmo (melhorar o desempenho, aumentar a qualidade da solução, etc.) para resolução de diferentes problemas. Portanto, essa meta-otimização pode ser estendida a qualquer algoritmo, resultando um meta-algoritmo ou uma meta-meta-heurística (BLUM e ROLI, 2003; TALBI, 2009).

No âmbito científico, é possível identificar um crescente interesse a respeito de meta-otimização. Muitos desses estudos adotam métodos estatísticos robustos para coletar, comparar e apresentar seus resultados. Um dos primeiros trabalhos (BARR, et al., 1995), considerado um marco na área de experimentação (BIRATTARI, 2009), discute diversos aspectos sobre o planejamento e análise de experimentos computacionais com métodos heurísticos.

Em Xu e Kelly (1996) pode-se encontrar um estudo sobre a influência dos parâmetros na busca Tabu. Na estratégia adotada pelos autores, um parâmetro é estudado separadamente, enquanto que os demais mantêm-se fixos. Essa abordagem sequencial permite estimar a influência de um único parâmetro sobre os demais. Entretanto, ela negligencia a existência de interações entre eles e não garante encontrar a solução ótima. Por outro lado, Coy et al. (2000) apresentam um estudo experimental sobre a configuração de duas heurísticas de busca local,

em que diversos parâmetros são combinados e estudados simultaneamente com o uso de DOE. O estudo consiste em avaliar diferentes configurações das heurísticas sobre um subconjunto de problemas semelhantes e os resultados (configurações sugeridas) são generalizados para resolver os demais problemas de uma classe de problemas. Os ajustes propostos são eficazes em termos de qualidade da solução e tempo de execução.

Um trabalho de grande destaque (ADESON-DÍAZ e LAGUNA, 2006) propõe um procedimento denominado CALIBRA, que combina o sistema de arranjos ortogonais de Taguchi (L_93^4) e heurísticas de busca local para ajustar qualquer tipo de algoritmo. A proposta é usar os arranjos ortogonais para encontrar regiões promissoras e focar a busca por meio de heurísticas. O procedimento se inicia com um planejamento fatorial para identificar uma região promissora. A região sugerida é explorada com a heurística até encontrar uma solução ótima local. Em seguida, os limites dos parâmetros são reduzidos e novos experimentos são conduzidos. Os resultados são novamente explorados pela heurística de busca local. O procedimento segue até que nenhuma modificação significativa seja identificada nos parâmetros. Os resultados de experimentos com diferentes problemas de otimização revelam diferenças significativas em termos de desempenho dos algoritmos. No entanto, as principais limitações desse procedimento são: o número de parâmetros (no máximo 4 sobre 3 valores críticos) e a ausência de interações entre os parâmetros. O procedimento proposto por Adeson-Díaz e Laguna (2006), portanto, é mais efetivo em situações quando há poucos parâmetros envolvidos e as interações entre eles podem ser negligenciadas.

O uso do DOE na sintonização de meta-heurísticas pode ser encontrado em uma ampla lista de trabalhos da literatura contemporânea (BARTZ-BEIELSTEIN, 2003; BARTZ-BEIELSTEIN et al., 2004; CHAN et al., 2006; KRAMER et al., 2007; RIDGE e KUDENCO, 2007; ARIN et al., 2011; KHOSHJAHAN et al., 2013). Em muitos desses trabalhos é possível identificar o estudo de diferentes técnicas de DOE, mas a maioria opta pelos planejamentos fatoriais e RSM. Também recentemente, alguns pesquisadores (DOBSLAW, 2010; SHAHSAVAR et al., 2011) têm investigado o uso combinado de DOE e técnicas de Inteligência Artificial para compor uma metodologia para auxiliar a sintonização de meta-heurísticas.

Uma outra linha de pesquisa (HUTTER et al., 2007 e 2009) se apoia na aplicação de Busca Local Iterativa (em Inglês, *Iterated Local Search* - ILS) em um conjunto pré-definido de parâmetros. A ILS pode ser definida como uma sequência de buscas locais em que a solução inicial é obtida por meio de uma perturbação do extremo local encontrado

anteriormente. Para que o mecanismo seja eficaz, a perturbação deve se manter próxima do ponto de partida a fim de manter o foco na área explorada.

Os estudos citados anteriormente adotam uma estratégia de sintonização de parâmetros, classificada como *off-line* (TALBI, 2009). Nessa estratégia, os parâmetros das meta-heurísticas (tamanho da lista na Busca Tabu, a taxa de arrefecimento no SA, as taxas de cruzamento e mutação no GA, etc.) são configurados antes da inicialização do algoritmo e mantêm-se constantes até o final de sua execução.

Por outro lado, na sintonização *on-line*, os parâmetros são controlados e atualizados dinamicamente durante a execução do algoritmo (TALBI, 2009). Ambas estratégias (*off-line* e *on-line*) são, algumas vezes, referenciadas na literatura como sintonização de parâmetros e controle de parâmetros, respectivamente (EIBEN e SMIT, 2012).

Uma desvantagem da estratégia *off-line* é o custo computacional elevado, principalmente, quando diferentes configurações são testadas em um número elevado de exemplares de um problema. Entretanto, geralmente, os parâmetros das meta-heurísticas são dependentes do problema ou mesmo dos exemplares de um problema. Uma outra desvantagem é a eficácia na definição *a priori* dos parâmetros, um vez que pode haver necessidade de modificações durante o processo de sintonização (TALBI, 2009).

Uma alternativa são as estratégias de sintonização de parâmetros *on-line*. Tipicamente, a estratégia *on-line* é baseada em técnicas de Inteligência Artificial, por exemplo, aprendizado de máquina⁹ (BIRATTARI, 2009), e em algoritmos inspirados na biologia, com processos que permitem a população se adaptar (herança gênica, sobrevivência do mais apto, etc.). As estratégias de sintonização *on-line* podem ser classificadas em:

- Dinâmica: na atualização dinâmica, as modificações dos parâmetros são executadas sem considerar o progresso da pesquisa. Isto é, podem ocorrer modificações ao acaso ou determinísticas;
- Adaptativa: na atualização adaptativa, as modificações dos parâmetros ocorrem com o progresso da pesquisa, considerando algum mecanismo de memória; e
- Autoadaptativa: na atualização autoadaptativa, as modificações dos parâmetros ocorrem de maneira autônoma com o progresso da pesquisa. Isto é, as

⁹ Área da Inteligência Artificial que se dedica ao desenvolvimento de técnicas computacionais que permitem ao computador tomar decisões automáticas baseado em experiências acumuladas da solução de problemas anteriores.

modificações afetam o estado atual e resultam em aprendizado e experiências para decidir as próximas modificações;

Na estratégia autoadaptativa os parâmetros e operadores do algoritmo são codificados como cromossomos e sujeitos aos mesmos mecanismos evolucionários de um GA. A adaptação ocorre de maneira autônoma a cada geração. Os melhores valores de parâmetros produzem melhores indivíduos que, por sua vez, têm probabilidade maior que a média de sobreviver, reproduzir e propagar os melhores valores (EIBEN e SMITH, 2003).

Existem duas maneiras de implementar esse conceito, uma delas consiste em utilizar uma heurística que considera o *feedback* do estado atual da pesquisa e modifica os valores de parâmetros (BEST et al., 2006). Uma outra maneira é incorporar os parâmetros nos cromossomos, tornando-os sujeitos a evolução (SMITH, 2008). Algumas abordagens autoadaptativas aplicadas na sintonização de meta-heurísticas podem ser encontradas em Hutter et al. (2006) e Kramer (2010).

As estratégias *on-line* são atraentes, mas sua aplicação, em geral, destina-se a um único exemplar grande e complexa (BIRATTARI, 2009). Essa abordagem contrasta com a ideia desta pesquisa, cuja proposta é utilizar uma diversidade de exemplares de uma classe de problemas para sintonizar uma meta-heurística.

3.4 DECLARAÇÃO DO PROBLEMA DE PESQUISA

Seja M uma meta-heurística qualquer com um conjunto de parâmetros $P = \{\alpha, \beta, \dots, \xi\}$ aplicada na resolução de diferentes problemas ou exemplares de um problema $E = \{e_1, e_2, \dots, e_S\}$. Considere que os parâmetros $\alpha, \beta, \dots, \xi$ da meta-heurística M podem assumir valores nos seguintes intervalos $[\alpha_1, \dots, \alpha_I], [\beta_1, \dots, \beta_J], \dots, [\xi_1, \dots, \xi_K]$, respectivamente, onde I, J, \dots, K representam as cardinalidades de cada conjunto de parâmetros envolvido no processo de sintonização. Seja $C_{ij\dots k} = (\alpha_i, \beta_j, \dots, \xi_k)$, com $i \in [1, I], j \in [1, J], \dots, k \in [1, K]$, uma configuração qualquer de parâmetros da meta-heurística M . Assim, no contexto desta pesquisa, o problema de sintonização de meta-heurísticas é formalizado na forma de um espaço de estados:

$$W = (e_s, C_{ij\dots k}), s \in [1, S], i \in [1, I], j \in [1, J], \dots, k \in [1, K] \quad (15)$$

O problema consiste em descobrir qual é a melhor configuração $C_{ij\dots k}$ dos parâmetros P da meta-heurística M para solução de um conjunto de problemas E . Assim, o produto de $S \times I \times J \times \dots \times K$ corresponde ao número esperado de testes experimentais para sintonização dos parâmetros de M sobre E .

Em síntese, a melhor configuração de M para solução de um problema e_s , com $s \in [1, S]$, é um ponto presente em (15) e sua determinação, na pior hipótese, se dá a partir da varredura completa desse espaço de buscas. Por exemplo, considere uma meta-heurística M com o seguinte conjunto de parâmetros $A \in \{\alpha_1, \alpha_2, \alpha_3\}$, $B \in \{\beta_1, \beta_2, \beta_3, \beta_4\}$, $\Gamma \in \{\gamma_1, \gamma_2, \gamma_3\}$ e $\Delta \in \{\delta_1, \delta_2, \delta_3, \delta_4, \delta_5\}$. Cada possível combinação de parâmetros, por exemplo, $\{\alpha_1, \beta_2, \gamma_3, \delta_4\}$, representa a uma configuração diferente de M , tal que $3 \times 4 \times 3 \times 5 = 180$ é o número esperado de testes experimentais para a sintonização dos parâmetros de M para a solução de e_s . Se o conjunto de problemas é tal que $|E| = 50$, ou seja, existem 50 exemplares a serem resolvidos, o número total de testes experimentais será $50 \times 180 = 9000$.

4. ESTRATÉGIAS PARA OTIMIZAÇÃO DE META-HEURÍSTICAS E O MÉTODO DE CORRIDA

4.1 INTRODUÇÃO

Em muitas situações a otimização de meta-heurísticas é realizada de maneira informal com o uso de abordagens entediadas e pouco confiáveis. O problema de otimização de meta-heurísticas é recente, assim como as próprias meta-heurísticas. Uma vez que esses algoritmos são dependentes do problema ou mesmo dos exemplares de um problema, eles precisam ser cuidadosamente sintonizados a fim de obter desempenho adequado na resolução de um problema específico. É nesse processo de sintonização que surgem diferentes estratégias para otimização de meta-heurísticas. Desde a última década a combinação de conceitos de diferentes áreas de conhecimento tem sido proposta para auxiliar a sintonização desses algoritmos.

A otimização semiautomática de meta-heurísticas por meio um *framework* que emprega conceitos de DOE e Inteligência Artificial é proposta em (DOBSLAW, 2010). Nessa abordagem, a etapa de treinamento é dividida em duas fases: sintonização de parâmetros e aprendizagem. Na fase de sintonização, a meta-heurística é aplicada na resolução de diversos exemplares de uma classe de problemas e, em seguida, as técnicas de DOE são utilizadas para definir um conjunto inicial de valores promissores para cada parâmetro. Na fase de aprendizado, uma rede neural artificial é treinada sob a supervisão do conjunto de valores sugeridos, com a finalidade de recomendar valores proeminentes para cada parâmetro. O resultado é um modelo de regressão generalizado para a classe de problemas estudada. O autor ressalta, no entanto, que a principal desvantagem da abordagem proposta é o consumo de tempo excessivo da etapa de treinamento.

Os conceitos da teoria de conjuntos *fuzzy* são aplicados na sintonização da meta-heurística GA em (SHAHSAVAR et al., 2011). A metodologia proposta consiste em utilizar planejamento experimental para encontrar uma combinação ideal para os parâmetros do GA, por exemplo tamanho da população, taxas de cruzamento e mutação. Uma vez que o objetivo é encontrar valores para cada parâmetro considerando, simultaneamente, a qualidade da solução e o tempo de CPU, o problema de otimização é resolvido com a aplicação de programação por metas (em Inglês, *goal programming*) visando transformar um problema biobjetivo em um problema único. Quando comparados com a literatura, os resultados do GA otimizado com técnicas da teoria de conjuntos *fuzzy* apresenta melhor desempenho (qualidade

da solução) e requer menos tempo de CPU para resolver problemas de programação de projetos.

Lessmann et al. (2011) propõem otimização da meta-heurística Otimização por Enxame de Partículas (em Inglês, *Particle Swarm Optimization* - PSO) com um único exemplar de um problema a partir da aplicação de conceitos de mineração de dados. O sistema proposto emprega métodos de regressão para identificar valores dos parâmetros da PSO com base em atualizações ou movimentos anteriores da meta-heurística. Inicialmente, nos primeiros ciclos, os modelos de regressão são estimados a partir de um conjunto de dados de treinamento. Nos ciclos subsequentes, o sistema explora a sensibilidade dos modelos e simula a aprendizagem *online* para atualizá-los. Resultados empíricos sugerem que a eficácia da sintonização de parâmetros pode estar relacionada com informações do exemplar selecionado (assinatura das partículas) e, por isso, o ajuste de regressão parece ser uma abordagem promissora.

Além das estratégias inovadoras propostas na literatura para sintonização de algoritmos, este capítulo também apresenta uma abordagem simples e direta de força bruta, bem como o método de Corrida, cujos conceitos são utilizadas nesta pesquisa no desenvolvimento de uma metodologia para sintonização de meta-heurísticas.

4.2 ESTRATÉGIA DE FORÇA BRUTA

A estratégia mais intuitiva para resolver o problema de sintonização de parâmetros é a força bruta. Em síntese, a estratégia executa o mesmo número de experimentos para cada configuração de um espaço de configurações.

Se o tempo total disponível para a sintonização de uma meta-heurística é T e cada teste ou experimento é executado em t unidades de tempo, então o número total de experimentos para otimizar uma meta-heurística pode ser estimado por $M = T/t$ (BIRATTARI, 2009).

Na estratégia de força bruta, o poder computacional é atribuído uniformemente para diferentes configurações de um espaço de configurações (S), tal que cada configuração candidata $s' \in S$ é testada por $N = M/|S|$ vezes.

Em um algoritmo de força bruta, N exemplares de um problema são selecionados arbitrariamente e as configurações candidatas são testadas uma única vez em cada exemplar. Os resultados são armazenados em memória e a melhor configuração é determinada com base em algum indicativo de desempenho das configurações candidatas sobre as diferentes

exemplares. A estrutura geral de um algoritmo de força bruta é ilustrada no pseudocódigo da Figura 16.

Figura 16 - Pseudocódigo de um algoritmo de força bruta.

```

Entrada:  $S_0, M$  // Configurações candidatas e número total de experimentos
Saída :  $S^*$  // Melhor configuração encontrada
1.  $S \leftarrow S_0$ ;
2.  $N \leftarrow M/S$ ; // total de experimentos para cada configuração candidata
3.  $A \leftarrow \text{vetor}(S)$ ; // desempenho das configurações candidatas
4.  $i \leftarrow 1$ ;
5. Para  $i < N$  Faça
6.    $e \leftarrow \text{criarExemplar}()$ ;
7. // Executar experimentos para cada configuração candidata com o exemplar "e"
8.   Para  $s' \in S$  Faça
9.      $c \leftarrow \text{experimento}(s', e)$ ;
10.     $A(s') \leftarrow \text{média}(A(s'), c, i)$ ; // atualizar o desempenho
11.   Fim Para;
12.    $i \leftarrow i + 1$ ;
13. Fim Para;
14.  $S^* \leftarrow \text{identificarMelhor}(A)$ ;
15. Retornar  $S^*$ ;

```

No algoritmo de força bruta os requisitos de memória são limitados, uma vez que um único fragmento de informação é armazenado. Isto é, a estimativa de desempenho de cada configuração candidata sobre as N diferentes instâncias. Portanto, um único vetor com o tamanho correspondente ao número total de configurações candidatas é suficiente para guardar toda a informação em processamento.

4.3 ESTRATÉGIA DE CORRIDA

Em Maron e Moore (1994), os autores propõem o método *Hoeffding race* para acelerar a seleção de modelos em problemas na Aprendizagem Supervisionada¹⁰. O interesse principal é selecionar os melhores parâmetros estruturais para um método baseado em memória. A

¹⁰ Técnica de Inteligência Artificial em que um sistema deve ser capaz de medir a diferença entre seu comportamento e o comportamento de referência e, se necessário, aplicar correções para reduzir a diferença.

ideia original (MARON, 1994) emprega como medida de desempenho o erro quadrático médio¹¹ computado por meio do método de validação cruzada *leave-one-out*, um método computacionalmente muito dispendioso que consiste em dividir um conjunto de dados em n subconjuntos mutuamente exclusivos de mesmo tamanho para utilizá-los na estimação dos parâmetros de um modelo. Em síntese, a validação *leave-one-out* é um método para avaliar a capacidade de generalização de um modelo, a partir de n exemplares de um conjunto de dados.

Por outro lado, as abordagens baseadas em memória representam uma boa alternativa, uma vez que não envolvem treinamentos e o tempo de computação necessário para realizar n previsões, cada uma com base em um conjunto diferente de exemplares, equivale ao tempo total para a realização de n previsões, com base em um único exemplar. Nesse contexto, a principal ideia introduzida pelo método *Hoeffding race* é que a busca pela melhor estrutura de modelo pode ser acelerada, descartando os candidatos de qualidade inferior assim que evidências suficientes são reunidas contra eles. Esta ideia caracteriza o método de Corrida (BIRATTARI, 2009).

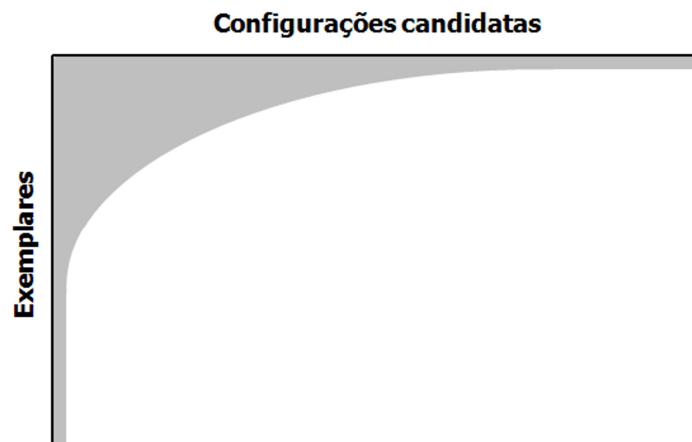
A validação *leave-one-out* relativa ao candidato genérico pode ser realizada incrementalmente. Isto é, a média de n erros quadráticos, sendo cada média relacionada a n exemplares do conjunto de dados. Essa quantidade pode ser aproximada pela média calculada em qualquer subconjunto desses erros (BIRATTARI, 2009). Deve-se observar que, independente do tamanho da amostra, a média calculada nessa amostra é uma estimativa sem tendência do erro quadrático médio e que sua variação diminui com o aumento do tamanho da amostra. À medida que o cálculo prossegue, a estimativa da medida *leave-one-out* para os candidatos fica mais refinada e um teste estatístico de hipótese pode ser adotado para decidir se a diferença observada nas estimativas é significativa. Em caso positivo, os candidatos de qualidade inferior são descartados da corrida e não serão mais avaliados nos ciclos subsequentes.

O método *Hoeffding race* original adota um teste estatístico baseado na fórmula de Hoeffding (1963) relativa à confiança na média empírica de n números positivos, amostrados de forma independente a partir da mesma distribuição, quando o limite superior da variável aleatória é conhecido.

¹¹ Erro quadrático médio (em Inglês, *Mean Squared Error* - MSE) corresponde à média da diferença entre uma estimativa (\hat{y}_i) e seu valor real observado (y_i) elevada ao quadrado, isto é, $MSE = \frac{1}{n} \sum (y_i - \hat{y}_i)^2$, em que n é o número total de observações.

O algoritmo *Hoeffding race* inicia com um conjunto de candidatos e procede considerando iterativamente os exemplares de um dado conjunto de dados de tamanho n . No passo k do processo iterativo, o k -ésimo exemplar do conjunto de dados é considerado e todos os candidatos na corrida são usados para prever a saída. Para um candidato genérico, o erro é usado para atualizar a estimativa *leave-one-out* do erro quadrático médio. Ao completar o passo k , todos os candidatos com um limite inferior abaixo do limite superior do melhor candidato até o momento são eliminados da corrida e deixam de ser considerados em futuras avaliações. A medida que a avaliação dos candidatos sobreviventes prossegue e mais exemplares são considerados no procedimento *leave-one-out*, os intervalos em torno das estimativas empíricas ficam mais precisos e o processo de seleção se torna mais refinado. A quantidade de computação envolvida no método *Hoeffding race* é ilustrada na Figura 17, isto é, à medida que o número de exemplares aumenta, a quantidade de configurações candidatas é reduzida.

Figura 17 - Representação gráfica da quantidade de computação envolvida no método *Hoeffding race*.



Fonte: (BIRATTARI, 2009), adaptado pelo autor.

A ideia subjacente as abordagens de Corrida é muito atraente e, originalmente, foram propostos alguns algoritmos baseados em diferentes testes estatísticos (MOORE e LEE, 1994). Dentre eles, destaca-se o BRACE, um algoritmo baseado em estatística Bayesiana que implementa uma técnica estatística conhecida como blocos (BOX et al., 1978; DEAN e VOSS, 1999; MONTGOMERY, 2012). Um planejamento experimental em blocos, em geral, é adotado quando dois ou mais candidatos são comparados, pois melhora a precisão com a qual as comparações são realizadas. Um bloco é um conjunto de condições experimentais

relativamente homogêneas em que todos os candidatos são testados. No contexto da seleção do melhor candidato a partir da validação *leave-one-out*, a adoção de blocos é simples e natural: cada candidato sobrevivente é testado nos mesmos exemplares e cada exemplar é considerado um bloco.

Durante um ciclo do algoritmo BRACE são estimados incrementalmente o valor médio e a variância do desempenho de cada par de candidatos na corrida. Com base nestas estimativas, um conjunto de testes estatísticos é executado a cada ciclo para verificar se algum candidato é significativamente pior do que qualquer outro. Em caso positivo, o candidato de qualidade inferior é descartado da corrida. O teste estatístico adotado no BRACE é equivalente ao teste de hipóteses *t*-Student em pares (SHESKIN, 2000), realizado em cada par de candidatos sobreviventes. Diferente do teste adotado no método *Hoeffding race*, o *t*-Student é um procedimento paramétrico, que se baseia em algumas premissas em relação às variáveis estocásticas envolvidas. Por exemplo, as estimativas de desempenho computadas são normalmente distribuídas. O BRACE provou ser eficaz e capaz de atingir resultados melhores do que o *Hoeffding race* (MARON e MOORE, 1997; MOORE e LEE, 1994).

Deve-se observar que o método *Hoeffding race* adota um teste estatístico não paramétrico, que exige conhecimento *a priori* de um limite sobre o erro observado e, por isso, reduz significativamente a aplicabilidade do método. Por outro lado, o BRACE prova que a adoção de planejamento em blocos é particularmente eficaz em ambientes de corrida. No entanto, para incluir esse recurso, os autores adotam um teste estatístico paramétrico, cuja aplicabilidade é limitada pela hipótese de que as estimativas envolvidas devem ser normalmente distribuídas.

4.3.1 Algoritmo de Corrida

Os Algoritmos de Corrida (em Inglês, *Racing Algorithms*) são inspirados no método *Hoeffding race* introduzido por Maron e Moore (1994) para resolver o problema de seleção de modelos em Aprendizado de Máquina¹².

A ideia principal subjacente à abordagem de Corrida é que a avaliação do desempenho de cada configuração candidata pode ser realizada de maneira incremental, considerando-se que os exemplares de um problema surgem sequencialmente em um fluxo de exemplares e

¹² Subcampo da Inteligência Artificial dedicado ao desenvolvimento de algoritmos e técnicas que permitam ao computador aperfeiçoar o desempenho em alguma atividade.

que cada configuração candidata da meta-heurística é executada uma única vez sobre um exemplar. Este procedimento garante a menor variância possível para a estimativa de desempenho (BIRATTARI, 2009).

Em um algoritmo de Corrida a estimativa de desempenho de um exemplar é obtida com base em todos os outros exemplares. A variância dessa sequência de estimativas diminui na ordem de $1/n$ e, conseqüentemente, a estimativa do desempenho de uma configuração candidata se torna mais confiável à medida que n cresce e converge para o verdadeiro valor médio de desempenho esperado.

Dada a possibilidade de estabelecer uma sequência de estimativas para cada configuração candidata, o algoritmo de Corrida pode ser definido como aquele que calcula de maneira incremental e em paralelo tais sequências para todas as configurações candidatas. Assim que são obtidas evidências suficientes de que a estimativa de um determinado candidato é pior do que todas as demais, ele é descartado das avaliações futuras.

Um algoritmo de Corrida, portanto, gera uma sequência de conjuntos aninhados de configurações candidatas:

$$S_0 \supseteq S_1 \supseteq S_2 \supseteq \dots,$$

sendo que o conjunto inicial é igual à totalidade de configurações candidatas, isto é, $S_0 = S$. A cada ciclo é possível descartar algumas configurações que parecem ser subótimas com base na informação disponível de ciclos anteriores.

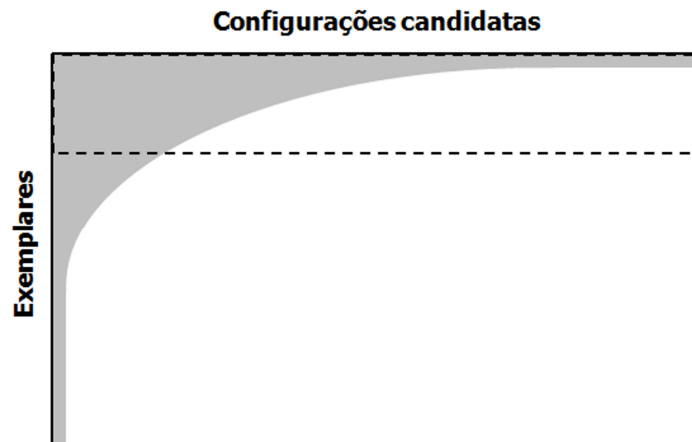
Em uma etapa k do ciclo iterativo, o conjunto de configurações candidatas na corrida é S_{k-1} e um novo exemplar e_k é considerado. Cada candidato $s' \in S_{k-1}$ é testado sobre o exemplar e seu desempenho adicionado ao respectivo vetor de desempenhos para formar um conjunto de vetores, cada um pertencente a uma configuração candidata. Ao final do ciclo k é definido um novo conjunto de configurações candidatas S_k a partir da retirada de uma ou mais configurações subótimas, com base em um teste estatístico que compara todos os desempenhos dos candidatos entre si. O procedimento descrito é iterado até que todas as configurações exceto uma são descartadas ou quando um número máximo de experimentos é alcançado. O pseudocódigo de um algoritmo de Corrida é apresentado na Figura 18.

Figura 18 - Pseudocódigo de um algoritmo de Corrida.

<p>Entrada: S_0 // <i>Configurações candidatas</i></p> <p>Saída : S^* // <i>Melhor configuração encontrada</i></p> <ol style="list-style-type: none"> 1. $S \leftarrow S_0$; // <i>Configurações candidatas sobreviventes</i> 2. $C \leftarrow \{\}$; // <i>Desempenho das configurações candidatas</i> 3. Repetir 4. $e \leftarrow \text{criarExemplar}()$; 5. // <i>Avaliar o desempenho de cada configuração candidata com o exemplar "e"</i> 6. Para $s' \in S$ Faça 7. $C \leftarrow C \cup \text{avaliar}(s', e)$; 8. Fim Para; 9. $F \leftarrow \text{testeEstatístico}(C)$; // <i>coletar estatísticas</i> 10. // <i>Selecionar configurações candidatas sobreviventes</i> 11. $S \leftarrow \text{selecionarElite}(s', C, F)$; 12. Até atingir um critério de parada; 13. $S^* \leftarrow \text{identificarMelhor}(S)$; 14. Retornar S^*;

A aparente vantagem da abordagem de Corrida em relação a de força bruta é que os algoritmos de Corrida permitem alocar melhor os recursos computacionais para as configurações candidatas. Isto é, ao invés de desperdiçar tempo computacional para estimar com precisão o desempenho de candidatos de qualidade inferior, a abordagem de Corrida foca nos candidatos mais promissores e obtém estimativas de variância de desempenho mais baixas para estes últimos. Esta característica garante uma seleção mais precisa do melhor candidato. Uma comparação entre ambas abordagens pode ser observada na Figura 19.

Figura 19 - Representação gráfica da quantidade de computação envolvida pelas abordagens de Corrida (área sombreada) e de força bruta (área tracejada).



Fonte: (BIRATTARI, 2009), adaptado pelo autor.

Em síntese, na abordagem de Corrida, um candidato é descartado assim que há evidências de que ele não é o melhor, ao passo que a abordagem de força-bruta testa todos os candidatos no mesmo número de instâncias.

Durante a corrida são mantidas algumas estatísticas dos resultados obtidos pelos candidatos sobreviventes (MENDENHALL et al., 2008). A quantidade de informação armazenada depende do teste estatístico utilizado para determinar se as diferenças observadas no desempenho dos candidatos é ou não significativa. No pior caso, todos os resultados observados serão armazenados para os candidatos sobreviventes. O requisito de memória para uma abordagem de Corrida é, portanto, limitada por M vezes a quantidade de memória necessária para armazenar o custo da solução encontrada em uma única execução da meta-heurística.

Uma variedade de testes estatísticos pode ser considerada na implementação de um algoritmo de Corrida (BIRATTARI et al., 2009). Por exemplo, o teste estatístico clássico t -Student e a correlação de Bonferroni foram adotados nos algoritmos tNo-Race e tBo-Race, respectivamente. O algoritmo F-Race (BIRATTARI et al., 2002) é baseado no teste estatístico de Friedman, um teste não paramétrico que emprega o conceito de blocos e validações pós testes para detectar a significância dos resultados.

O F-Race é um algoritmo de Corrida inspirado nas principais características dos algoritmos *Hoeffding race* e BRACE, que são a abordagem estatística não paramétrica e a técnica estatística de blocos, respectivamente. O teste de Friedman adotado pelo algoritmo considera a estatística de Conover (1999). Para uma breve descrição do teste, assume-se que o

algoritmo alcançou uma etapa k do ciclo iterativo e que existem $k-1$ configurações candidatas na corrida. No teste de Friedman, os desempenhos das configurações são realizações de variáveis aleatórias mutuamente independentes, denominadas blocos. Cada bloco corresponde aos resultados computacionais obtidos num exemplar por cada configuração na corrida até a etapa k . Os resultados são classificados em ordem crescente dentro de cada bloco e o teste de Friedman é então aplicado. Se o valor do teste excede um nível de significância α , então a hipótese nula é rejeitada e há evidências de que, pelo menos, uma configuração candidata tem melhor desempenho. As análises prosseguem com comparações entre pares de candidatos a fim de descartar aqueles com os piores desempenhos. As configurações descartadas deixam de participar das etapas futuras da corrida e o procedimento segue até restarem dois candidatos na corrida. Nesse caso, os candidatos restantes são analisados com o teste de Wilcoxon (CONOVER, 1999), uma vez que este se revelou mais robusto e eficiente quando restam apenas duas configurações candidatas (BIRATTARI, 2009).

4.3.2 Teste Estatístico de Friedman

Procedimentos estatísticos não paramétricos são indicados nas análises de dados de baixa qualidade, em geral, provenientes de pequenas amostras sobre as quais pouco ou nada se sabe referente à sua distribuição.

Frequentemente, na comparação de duas ou mais amostras, o interesse é detectar diferenças dentre os grupos. Na estatística não paramétrica, o procedimento mais conhecido para analisar essas diferenças é o teste estatístico de Friedman. A estatística de Friedman é um teste não paramétrico análogo à ANOVA (Seção 2.4.1), cujo objetivo é determinar a partir de pequenas amostras se existem diferenças entre os efeitos dos grupos analisados (GARCIA et al., 2010).

O primeiro passo para o cálculo do teste de Friedman é estabelecer uma correspondência entre os resultados originais das amostras e sua classificação. Por exemplo, na Tabela 4 são apresentados resultados ilustrativos dos desempenhos de quatro algoritmos (M) sobre 10 exemplares diferentes de um problema. Os desempenhos de cada algoritmos são classificados separadamente por exemplar, isto é, em blocos, de forma que o melhor desempenho de um bloco tem a classificação 1, o segundo melhor tem a classificação 2, e assim sucessivamente. Em caso de empates, são calculados os valores médios das classificações.

Tabela 4 - Desempenho de quatro algoritmos (M) sobre dez exemplares diferentes de um problema.

Exemplar	$M_1 (r_i^1)$	$M_2 (r_i^2)$	$M_3 (r_i^3)$	$M_4 (r_i^4)$
1	0,000 (1)	0,330 (2)	0,665 (4)	0,628 (3)
2	0,068 (1)	0,686 (2)	0,878 (4)	0,850 (3)
3	0,561 (2)	0,662 (3)	0,726 (4)	0,199 (1)
4	0,216 (1)	0,232 (3)	0,231 (2)	0,544 (4)
5	0,884 (3)	0,308 (1,5)	0,308 (1,5)	0,933 (4)
6	0,215 (1)	0,313 (3)	0,362 (4)	0,292 (2)
7	0,332 (1)	0,483 (2)	0,566 (3)	0,594 (4)
8	0,269 (1)	0,502 (3)	0,437 (2)	0,633 (4)
9	0,044 (1)	0,918 (4)	0,405 (2)	0,482 (3)
10	0,128 (1,5)	0,415 (4)	0,281 (3)	0,128 (1,5)
R_j	1,35	2,75	2,95	2,95

Seja r_i^j a classificação do j -ésimo algoritmo sobre o i -ésimo exemplar de um conjunto com n exemplares de um problema e $R_j = \frac{1}{n} \sum r_i^j$, o valor médio das classificações. Sob a hipótese nula, que estabelece que todos os algoritmos se comportam de maneira semelhante e, portanto suas classificações médias R_j devem ser iguais, o teste estatístico de Friedman é:

$$T = \frac{12n}{k(k+1)} \left[\sum R_j^2 - \frac{k(k+1)^2}{4} \right], \quad (16)$$

distribuído de acordo com χ^2 , isto é, a distribuição de probabilidades chi-quadrado, com $k-1$ graus de liberdade, quando n e k são grandes o suficiente, por exemplo n maior que 10 e k maior que 5.

Os valores médios das classificações (Tabela 4) oferecem informações úteis sobre o desempenho de cada algoritmo, ou seja, na média M_1 (1,35) é melhor, M_2 (2,75) é o segundo melhor, enquanto que M_3 (2,95) e M_4 (2,95) são o terceiro e quarto melhores com a mesma classificação, respectivamente. O teste de Friedman comprova se as medidas de classificações médias são significativamente diferentes de $R_j = 2,5$, esperado sob a hipótese nula. Com a aplicação de (16), $T = 10,74$ e, considerando os quatro algoritmos sobre 10 exemplares, T é distribuído de acordo com a distribuição χ^2 com $4-1 = 3$ graus de liberdade. Uma vez que o valor- p (em Inglês, p -value) computado com a distribuição $\chi^2_3(10,74) \cong 0,01$ e $p < \alpha$, em que usualmente $\alpha = 0,05$, concluí-se que a hipótese nula é rejeitada com alto grau de significância e, portanto, há diferenças entre os resultados analisados. Em testes estatísticos de hipóteses, o

valor- p corresponde à probabilidade de se obter um resultado, pelo menos, tão extremo como o que foi observado, assumindo que a hipótese nula é verdadeira. O valor- p fornece informações sobre a significância de um teste de hipóteses e, também, indica o quanto um resultado é significativo. Em geral, quanto menor o valor- p , mais forte a evidência contra a hipótese nula.

Se a hipótese nula for rejeitada, pode-se prosseguir as análises com um teste *post hoc* complementar. Em geral, o objetivo é comparar uma amostra, denominada controle, com outras amostras consideradas no estudo e decidir quais são significativamente diferentes das outras. A estatística do teste para comparações das amostras em pares depende do procedimento não paramétrico utilizado. Em (GARCIA et al., 2010) a expressão adotada para o teste de Friedman é:

$$z = (R_i - R_j) / \sqrt{\frac{k(k+1)}{6n}}, \quad (17)$$

em que R_i e R_j são as classificações médias obtidas com o teste de Friedman para duas amostras comparadas.

4.4 ALGUMAS ABORDAGENS COM MÉTODOS DE CORRIDA

Na literatura contemporânea é possível encontrar abordagens com o uso do método de Corrida na otimização de diferentes meta-heurísticas. Yuan e Gallagher (2007) propõem uma técnica híbrida para sintonização de algoritmos evolucionários, que combina mecanismos de pesquisa exaustiva e um método de Corrida. Os mecanismos de pesquisas exaustiva são utilizados em situações em que o espaço de buscas é grande e expansível e contém parâmetros simbólicos/nominais, enquanto que os métodos de Corrida são bons quando o espaço de buscas é limitado e restrito à parâmetros numéricos. Resumidamente, a pesquisa exaustiva é responsável por explorar o espaço de buscas e o método de Corrida é aplicado como um método de força bruta, para reduzir significativamente o custo associado em encontrar o melhor candidato de um conjunto de configurações candidatadas de tamanho moderado. A partir de resultados de estudos experimentais, os autores sugerem que a pesquisa exaustiva deve ser usada quando o espaço de buscas é pequeno, por exemplo com dez candidatos ou

menos, enquanto que os métodos de Corrida são mais indicados para espaço de buscas com poucas centenas de candidatos. Em outras situações a técnica híbrida pode ser considerada.

A viabilidade de utilizar o método de Corrida para otimizar algoritmos evolucionários é demonstrada em Smit e Eiben (2009). Nesse estudo a proposta é utilizar os algoritmos evolucionários com módulos independentes para aumentar a eficiência da busca, por exemplo métodos de Corrida e de afiação (em Inglês, *sharpening*). O método de afiação (BARTZ-BEIELSTEIN et al., 2004) consiste em diminuir o número de avaliações para estimar a qualidade dos parâmetros e, como consequência, sumarizar o tempo dos algoritmos de sintonização. A ideia principal é iniciar o algoritmo de sintonização com um conjunto pequeno de configurações candidatas e aumentá-lo progressivamente. A combinação de Corrida e afiação faz com que o maior esforço da sintonização seja dedicado para as configurações promissoras. Resultados experimentais com diferentes combinações de algoritmos e métodos sugerem que a abordagem proposta é vantajosa e não prejudica o desempenho dos algoritmos evolucionários.

Yuan et al. (2010) apresentam um estudo sobre a configuração de algoritmos com o método de busca em malha direta adaptativa (em Inglês, *mesh adaptive direct search - MADS*), um método de pesquisa para otimização global contínua. A proposta é criar um algoritmo híbrido que adiciona os conceitos de MADS ao F-Race. Nessa abordagem, a população de configurações candidatas é amostrada por MADS com a adição de um ponto de controle e avaliada com o F-Race. Se o resultado do F-Race determina que o ponto de controle é melhor, então o processo de configuração falha e um novo ponto de controle é analisado. Caso contrário, o processo é bem sucedido, os parâmetros da MADS são atualizados e uma nova população é amostrada e avaliada. Resultados computacionais demonstram que o método proposto é bem sucedido.

Em Styles e Hoos (2013), os autores demonstram como otimizar algoritmos de alto desempenho a partir de dois protocolos de configuração, que combinam a ideia de aumentar o grau de dificuldade dos exemplares para validar as configurações candidatas. Um dos protocolos utiliza o F-Race, enquanto que o outro é baseado no conceito de corrida com permutação ordenada, que emprega um teste de permutação focado no tempo de execução para eliminação das configurações candidatas. Em síntese, o teste de permutação realiza comparações em pares entre o candidato líder da corrida e os demais candidatos sobreviventes. O procedimento proposto exige a realização de múltiplas análises de correlações entre os candidatos e corridas realizadas sobre configurações obtidas previamente por um processo de configuração anterior. Resultados de um grande estudo empírico

demonstram que os protocolos estudados são eficazes e superam resultados obtidos por outros algoritmos que adotam os mesmos protocolos na literatura.

5. METODOLOGIA PROPOSTA PARA OTIMIZAÇÃO DE META-HEURÍSTICAS

5.1 INTRODUÇÃO

As meta-heurísticas são técnicas promissoras com bom desempenho na resolução de problemas complexos para os quais não há algoritmos específicos e/ou eficientes. Em geral, as meta-heurísticas possuem um conjunto de parâmetros dos quais são extremamente dependentes. A sintonização desses parâmetros requer uma estratégia cuidadosa para garantir o melhor desempenho aos algoritmos. Entretanto, esta atividade nem sempre é trivial e, frequentemente, é realizada *ad hoc* de acordo o problema sob análise, uma vez que a escolha de valores inadequados pode resultar no mau desempenho dos algoritmos e/ou em soluções de baixa qualidade.

A fim de resolver o problema de sintonização de parâmetros de meta-heurísticas (Seção 3.4), a presente pesquisa propõe uma metodologia que combina o uso de técnicas robustas de Estatística e métodos eficientes de Inteligência Artificial. Nesse contexto, se destacam DOE, um arcabouço de técnicas estatísticas que permite estudos simultâneos de múltiplos parâmetros combinados entre si (Capítulo 2), e o método de Corrida, que emprega um método estatístico não paramétrico para simplificar a avaliação de alternativas para a sintonização de algoritmos (Capítulo 4).

Em síntese, a abordagem proposta nesta pesquisa tem a finalidade de definir uma metodologia para auxiliar a sintonização de meta-heurísticas a partir da utilização combinada de DOE e método de Corrida. Os principais benefícios desta metodologia são a redução de esforço e tempo investidos na concepção e otimização de algoritmos eficazes visando a resolução de diferentes problemas de otimização, bem como a confiabilidade na escolha de valores para os parâmetros e a possibilidade de replicação das sintonizações obtidas. A ausência de uma metodologia para sintonização demanda um esforço maior e, conseqüentemente, mais tempo no desenvolvimento dos algoritmos e, muitas vezes, conhecimento específico do problema sob análise. Portanto, a ideia central de uma metodologia para sintonização de meta-heurísticas é desenvolver um método heurístico automático a partir do qual pode-se encontrar sistematicamente boas configurações para diferentes algoritmos com base em avaliações estatísticas de um vasto conjunto de problemas.

Este capítulo apresenta a contribuição dos estudos de doutoramento para resolução do problema de sintonização de parâmetros de meta-heurísticas. A ideia da metodologia proposta é um método heurístico em que as possíveis configurações de parâmetros são imaginadas

como um espaço de busca. Em síntese, o método desenvolvido concentra as buscas em configurações candidatas promissoras, criadas dinamicamente em um processo iterativo, e utiliza uma técnica estatística robusta para avaliar as diferentes alternativas (configurações candidatas) e descartar aquelas de qualidade inferior assim que houver evidências estatísticas para tanto.

5.2 DESCRIÇÃO GERAL

A metodologia para sintonização de parâmetros de meta-heurísticas proposta nesta pesquisa pode ser expressa em um procedimento que se inicia com a seleção arbitrária de n exemplares ($n > 1$) de uma classe de problemas de otimização a serem resolvidos por uma determinada meta-heurística. Os diferentes exemplares são estudados por meio de planejamentos fatoriais completos 2^k , a partir da definição de níveis (alto e baixo) que cada parâmetro da meta-heurística pode assumir.

Planejamentos fatoriais são úteis para identificar os parâmetros (fatores) que têm influência no desempenho dos algoritmos (resposta), mas são insuficientes quando o interesse é determinar ajustes dos fatores que podem otimizar o processo e resultar em soluções de boa qualidade ou mesmo, eventualmente, na solução ótima, para o problema sob análise. Por isso, o próximo estágio da abordagem consiste em empregar a Metodologia da Superfície de Respostas (RSM) como uma ferramenta para explorar as regiões promissoras próximas da solução ótima de cada exemplar e determinar os ajustes ou configurações de qualidade para os fatores estudados.

Os exemplares previamente selecionados são tratados como um conjunto de treinamento, sobre o qual são realizados estudos experimentais para determinar a sintonização de parâmetros de uma meta-heurística. Portanto, ao final dos estudos experimentais existirão n diferentes configurações para cada parâmetro, sendo cada uma relacionada a um exemplar. As configurações identificadas a partir do conjunto de treinamento garantem diversidade aos parâmetros e permitem definir um intervalo de configurações promissoras para cada um deles, denominado espaço de busca de configurações candidatas, cujos limites são os valores máximo e mínimo de cada parâmetro obtido a partir do conjunto de treinamento.

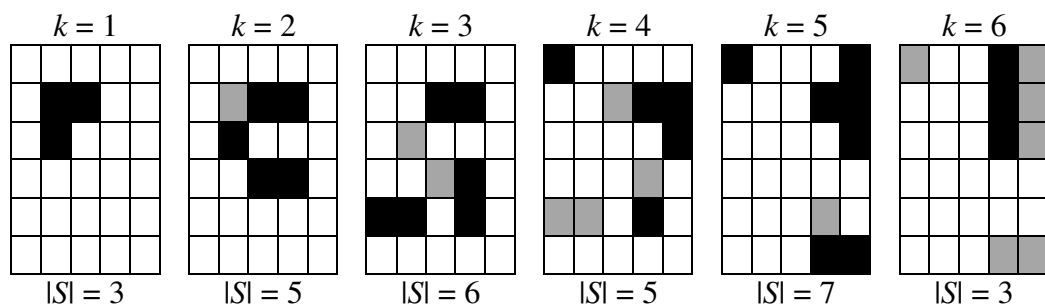
O procedimento segue em um ciclo principal, em que o objetivo é perseguir diferentes alternativas (configurações candidatas) na vizinhança da melhor configuração candidata conhecida, respeitando-se os limites de cada parâmetro no espaço de busca definido

anteriormente. Uma configuração candidata corresponde a uma alternativa diferente obtida a partir de modificações da melhor configuração conhecida.

Durante o processo iterativo as configurações candidatas são classificadas separadamente em blocos, isto é, por exemplar, de acordo com o seu desempenho, tal que a configuração com o melhor desempenho tem a classificação 1, a segunda melhor tem a classificação 2, e assim sucessivamente. Em caso de empates entre diferentes configurações atribui-se o valor médio das classificações a cada um dos candidatos. Após as classificações, as configurações candidatas são avaliadas por meio de um método estatístico não paramétrico e aquelas com qualidade inferior são descartadas do processo assim que houver evidências estatísticas que justifiquem o descarte. A cada iteração, novas alternativas compõem um conjunto de configurações candidatas de tamanho variado, uma vez que algumas configurações são incluídas e outras podem ser excluídas de acordo com as avaliações.

Por exemplo, a Figura 20 ilustra o processo dinâmico de inclusão (em preto) e exclusão (em cinza) de alternativas em um conjunto de configurações candidatas (S). Neste exemplo, a cada iteração k , três novos candidatos do espaço de busca são escolhidos para serem analisados. Inicialmente ($k = 1$), três alternativas compõem o conjunto de configurações candidatas, tal que $|S| = 3$. A partir da segunda iteração ($k = 2$) há inclusão de três novas configurações candidatas em S , mas ao final dessa iteração uma delas, aquela com qualidade inferior, é excluída do conjunto de alternativas, de modo que $|S| = 5$. Na iteração seguinte ($k = 3$) novas alternativas são incluídas ao conjunto S e outras de qualidade inferior são excluídas, resultando em $|S| = 6$. Durante os ciclos é possível observar variações no tamanho de S , devido as inclusões e exclusões de configurações candidatas. Entretanto, não há um padrão definido para determinar o tamanho de S ao final de cada iteração, uma vez que as exclusões são baseadas nas estatísticas coletadas durante os ciclos.

Figura 20 - Processo dinâmico de inclusão (em preto) e exclusão (em cinza) de alternativas no conjunto S de configurações candidatas.



FONTE: Autoria própria.

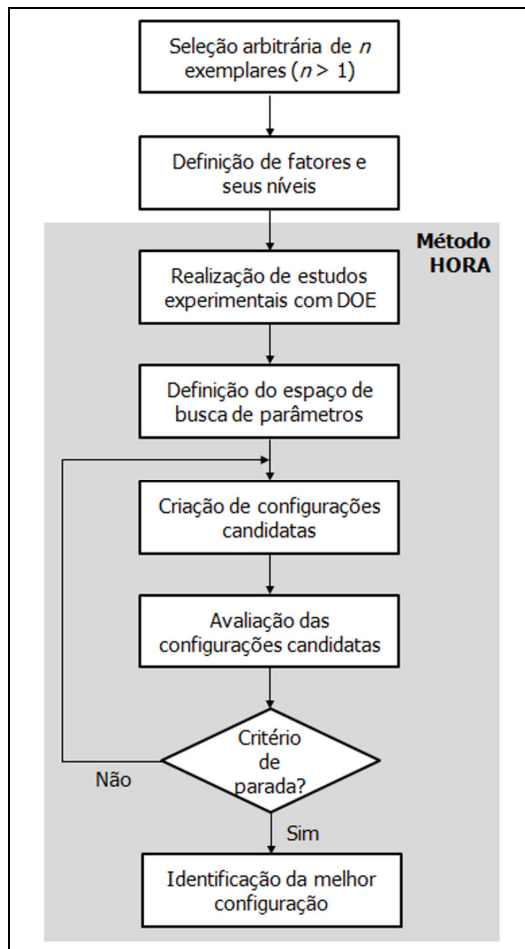
As configurações candidatas criadas durante o processo iterativo são avaliadas com a estatística não paramétrica de Friedman (Seção 4.3.2) e os testes *post hoc* complementares são realizados de acordo com a expressão (17).

O processo iterativo segue até atingir um critério de parada, por exemplo número máximo de alternativas no conjunto de configurações candidatas, tempo de execução maior do que um limite máximo, número de repetições sem alterações no conjunto de configurações candidatas, dentre outros.

Ao final, a melhor configuração, isto é, a sintonização de parâmetros sugerida para a meta-heurística, é identificada dentre as configurações candidatas sobreviventes com base no menor valor médio de suas classificações, obtidas ao longo do processo.

A metodologia descrita nos parágrafos anteriores, nomeada Algoritmo de Corrida Orientada por Heurística (em Inglês, *Heuristic Oriented Racing Algorithm - HORA*), é, por si mesma, uma heurística, que é sumarizada no esquema lógico apresentado na Figura 21.

Figura 21 - Esquema lógico da abordagem heurística elaborada nesta pesquisa.



FONTE: Autoria própria.

5.2.1 Algoritmo de Corrida Orientada por Heurística

A estrutura geral do algoritmo HORA é ilustrada por meio de um pseudocódigo na Figura 22.

Figura 22 - Pseudocódigo do algoritmo HORA.

<p>Entrada: S_0, B_s, K // <i>Configuração candidata inicial, espaço de busca de configurações candidatas e número máximo da vizinhanças</i></p> <p>Saída : S^* // <i>Melhor configuração encontrada</i></p> <ol style="list-style-type: none"> 1. $S \leftarrow S_0$; // <i>Melhor configuração candidata sobrevivente</i> 2. $S' \leftarrow \{\}$; // <i>Configurações candidatas vizinhas</i> 3. $S'' \leftarrow \{\}$; // <i>Conjunto de configurações candidatas sobreviventes</i> 4. $C \leftarrow \{\}$; // <i>Desempenho das configurações candidatas</i> 5. Repetir 6. $e \leftarrow \text{criarExemplar}()$; 7. $S' \leftarrow \text{criarVizinho}(S, B_s, K)$; 8. $S'' \leftarrow S'' \cup S'$; 9. // <i>Avaliar o desempenho de cada configuração candidata com o exemplar "e"</i> 10. Para $s \in S'$ Faça 11. $c \leftarrow C \cup \text{MH}(s, e)$; // <i>Meta-heurística (MH) alvo sob avaliação</i> 12. Fim Para; 13. $F \leftarrow \text{testeEstatístico}(C)$; // <i>coletar estatísticas</i> 14. // <i>Selecionar configurações candidatas sobreviventes</i> 15. $S'' \leftarrow \text{selecionarElite}(S'', C, F)$; 16. $S \leftarrow \text{identificarMelhor}(S'', C, F)$; 17. Até atingir um critério de parada; 18. $S^* \leftarrow S$; 19. Retornar S^*;
--

O algoritmo segue um ciclo principal (linhas 5-17), em que são criadas diferentes alternativas respeitando-se os limites do espaço de busca de configurações candidatas (linha 7). As alternativas compõem um conjunto de configurações candidatas (linha 8), cujas soluções são coletadas por meio de execuções da meta-heurística alvo (MH) sobre diferentes exemplares (linhas 10-12). A solução de cada configuração candidata é avaliada por meio do teste estatístico não paramétrico de Friedman (Seção 4.3.2), e aqueles candidatos de qualidade inferior, isto é, com os piores desempenhos, são descartados do processo (linha 15). O

processo iterativo segue até atingir um critério de parada (linha 17) e a melhor configuração candidata é aquela com o menor valor médio das classificações ao longo do processo.

Dentre as funções que compõem o HORA, o procedimento *criarExemplar()* (linha 6, Figura 22) consiste na criação de sequências de exemplares de um problema ou na escolha de alguns exemplares a partir de um *benchmark* de problemas de otimização.

O procedimento para criação de alternativas (linha 7, Figura 22) é muito relevante para o HORA, pois define os novos candidatos que serão avaliados durante os ciclos. A estrutura geral para criação de configurações candidatas é ilustrada no pseudocódigo da Figura 23.

Figura 23 - Pseudocódigo do algoritmo para criação de novas configurações candidatas.

```

Entrada: S, Bs, K // Melhor configuração candidata sobrevivente, espaço de busca de
configurações candidatas e número máximo de vizinhanças
Saída : S' // Configurações candidatas
1. P ← sortearParametros(|S|); // Sortear os parâmetros que serão alterados
2. S' ← {};
3. Para K vezes Faça
4.   Para p' ∈ P Faça
5.     Pn ← S(p') - aleatório()*(max(Bs) - min(Bs)); // Limite mínimo tolerável de S(p')
6.     Px ← S(p') + aleatório()*(max(Bs) - min(Bs)); // Limite máximo tolerável de S(p')
7.   // Verificar consistências dos parâmetros com relação aos limites originais
8.     Se Pn < min(Bs) Então
9.       Pn ← min(Bs);
10.    Fim Se;
11.    Se Px > max(Bs) Então
12.      Px ← max(Bs);
13.    Fim Se;
14.  // Criar novas configurações candidatas
15.    Se aleatório() < 0.5 Então
16.      P'' ← Px - aleatório()*(Px - Pn);
17.    Senão
18.      P'' ← Pn + aleatório()*(Px - Pn);
19.    Fim Se;
20.    S''(p') ← P'';
21.  Fim Para;
22.  S' ← S' ∪ S'';
23. Fim Para;
24. Retornar S';

```

No início do procedimento, os parâmetros que serão utilizados para criar novas alternativas são sorteados arbitrariamente. Para cada parâmetro sorteado é definido um intervalo de valores toleráveis (linhas 5 e 6, Figura 23), respeitando-se os limites originais do espaço de busca. A partir das definições iniciais são criadas novas alternativas na vizinhança da melhor configuração candidata sobrevivente.

As alternativas criadas durante os ciclos do HORA são avaliadas com o teste estatístico não paramétrico de Friedman (linha 13, Figura 22). O procedimento de avaliação (Figura 24) é iniciado a partir das soluções da meta-heurística alvo coletados sob diferentes configurações candidatas e exemplares. As soluções produzidas por cada configuração candidata são classificadas em blocos de acordo com seu desempenho. Em seguida, são calculados os valores médios das classificações nos blocos e, então, a soma quadrática desses valores. A estatística de Friedman é computada (linha 13, Figura 24) de acordo com a equação (16).

Figura 24 - Pseudocódigo do algoritmo para avaliação de configurações candidatas.

```

Entrada: C // Soluções da meta-heurística alvo sob diferentes configurações candidatas
Saída : F // Resultado das avaliações estatísticas
1.  $n \leftarrow |C|$ ;
2.  $k \leftarrow |C(1,.)|$ ; // Número de colunas em C
3.  $R \leftarrow \text{criarClassificações}(C)$ ;
4.  $M \leftarrow \{\}$ ; // Conjunto de valores médios associados com as classificações
5. Para k vezes Faça
6.    $M \leftarrow M \cup \text{média}(R(.,k))$ ;
7. Fim Para;
8.  $T \leftarrow 0$ ;
9. Para  $m' \in M$  Faça
10.   $T \leftarrow T + \text{soma}(m')^2$ ;
11. Fim Para;
12. // Teste estatístico de Friedman
13.  $F = ((12 * n)/(k * (k + 1))) * (T - (k * (k + 1)^2)/4)$ ;
14. Retornar F;

```

A partir do resultado do teste estatístico, as alternativas são comparadas em pares e aquelas de qualidade inferior são excluídas do conjunto de configurações candidatas (linha 15, Figura 22). Inicialmente, a hipótese nula é verificada ao nível de significância de 90% para confirmar se as classificações dos candidatos nos blocos são iguais. Se a hipótese nula é

rejeitada, então, pelo menos, uma configuração candidata deve ter o desempenho melhor do que os demais. Duas configurações candidatas são consideradas diferentes se o resultado das comparações efetuadas por meio da equação (17) exceder a estatística *t*-Student. Neste caso, o candidato de qualidade inferior é excluído do conjunto de configurações candidatas. Por outro lado, se a hipótese nula é aceita, nenhuma alternativa é excluída do conjunto de configurações candidatas. A estrutura geral para comparações de configurações candidatas é ilustrada no pseudocódigo da Figura 25.

Figura 25 - Pseudocódigo do algoritmo para seleção de configurações candidatas sobreviventes.

<p>Entrada: <i>S, C, F</i> // <i>Configurações candidatas, soluções da meta-heurística alvo sob diferentes configurações candidatas e valor do teste estatístico de Friedman</i></p> <p>Saída : <i>S'</i> // <i>Configurações candidatas sobreviventes</i></p> <ol style="list-style-type: none"> 1. $\alpha \leftarrow 0.90$; // <i>Nível de significância</i> 2. $R \leftarrow \text{criarClassificações}(C)$; 3. $M \leftarrow \text{computarMédia}(R)$; 4. $S' \leftarrow \text{min}(M)$; 5. $p \leftarrow \chi^2_{(k-1)}(F)$; 6. $q \leftarrow 1 - p$; 7. // <i>Comparações de configurações candidatas em pares</i> 8. Se $q < (1 - \alpha)$ Então 9. Para <i>k</i> vezes Faça 10. $\text{test} \leftarrow \text{comparar}(S', M(k))$; 11. Se $\text{test} > t_{1-\alpha, v}$ Então 12. $S \leftarrow S \setminus S(k)$; 13. Fim Se; 14. Fim Para; 15. $S' \leftarrow S' \cup S$; 16. Senão 17. $S' \leftarrow S$; 18. Fim Se; 19. Retornar <i>S'</i>;
--

O melhor candidato presente no conjunto de configurações candidatas é escolhido com base no menor valor médio das classificações obtidas ao longo dos ciclos (linha 16, Figura 22).

A seção a seguir apresenta um exemplo de uso do método HORA proposto a fim de esclarecer e ilustrar os algoritmos apresentados nas Figuras 22 a 25.

5.2.2 Exemplo Ilustrativo do Algoritmo HORA

A presente seção apresenta um exemplo da execução do algoritmo HORA para auxiliar a sintonização de meta-heurísticas. Neste exemplo, considera-se a meta-heurística GA e um problema de otimização meramente ilustrativo. Os parâmetros do GA estudados para sintonização são: probabilidade de cruzamento (p_c), probabilidade de mutação (p_m), tamanho da população (μ) e número de gerações (n).

Foram definidos os seguintes limites para o espaço de busca de configurações candidatas:

- $p_c = (0,65; 0,80)$;
- $p_m = (0,01; 0,04)$;
- $\mu = (70; 90)$; e
- $n = (800; 1000)$.

A partir de uma configuração inicial S_0 são criadas arbitrariamente três configurações candidatas (Tabela 5), respeitando-se os limites originais do espaço de busca.

Tabela 5 - Conjunto de configurações candidatas. ($k = 1$; $|S| = 4$)

Config.	p_c	p_m	μ	n
1	0,761	0,025	80	900
2	0,750	0,034	77	845
3	0,780	0,035	84	852
4	0,758	0,026	81	842

A meta-heurística GA é executada com cada uma das configurações candidatas sobre diferentes exemplares do problema de otimização. As soluções produzidas por cada configuração (Tabela 5), apresentadas em termos do valor da função objetivo, são classificadas (r) dentro do bloco (Tabela 6). Considere que ao final do primeiro ciclo ($k = 1$) não há exclusão de alternativas. Deve-se ressaltar que todas as configurações candidatas são testadas sobre os dois primeiros exemplares, pois para a realização do teste de Friedman são necessários, pelo menos, dois blocos de resultados.

Tabela 6 - Desempenho da meta-heurística GA coletados sob diferentes configurações.

Ciclo	Configurações Candidatas			
	1 (r)	2 (r)	3 (r)	4 (r)
$k = 1$	567 (3)	483 (2)	333 (1)	594 (4)
R	3	2	1	4

No próximo ciclo ($k = 2$) três novos candidatos são adicionados ao conjunto de configurações candidatas (Tabela 7).

Tabela 7 - Conjunto de configurações candidatas sobreviventes. ($k = 2$; $|S| = 7$)

Config.	p_c	p_m	μ	n
1	0,761	0,025	80	900
2	0,750	0,034	77	845
3	0,780	0,035	84	852
4	0,758	0,026	81	842
5	0,779	0,039	79	871
6	0,784	0,039	74	880
7	0,794	0,030	79	831

Considere que a partir de avaliações estatísticas, quatro candidatos de qualidade inferior (1, 2, 4 e 5) são excluídos do conjunto de configurações candidatas. Ao final do segundo ciclo ($k = 2$), a configuração candidata 6 tem a melhor qualidade (valor médio das classificações, $R = 1$), sendo as duas alternativas restantes (3 e 7) iguais, isto é, ambas têm o mesmo valor médio das classificações, $R = 3$ (Tabela 8).

Tabela 8 - Desempenho das configurações candidatas sobreviventes.

Ciclo	Configurações Candidatas						
	1 (r)	2 (r)	3 (r)	4 (r)	5 (r)	6 (r)	7 (r)
$k = 1$	567 (5)	483 (4)	333 (2)	594 (6)	673 (7)	203 (1)	392 (3)
$k = 2$	830 (6)	588 (5)	483 (4)	224 (2)	840 (7)	121 (1)	286 (3)
R	5,5	4,5	3	4	7	1	3

O processo segue ($k = 3$) com a criação de três novas alternativas (Tabela 9). Todas as configurações candidatas criadas durante o processo de corrida são avaliadas sobre os mesmos exemplares considerados nas avaliações das configurações que permanecem na corrida.

Tabela 9 - Conjunto de configurações candidatas sobreviventes. ($k = 3$; $|S| = 6$)

Config.	p_c	p_m	μ	n
3	0,780	0,035	84	852
6	0,784	0,039	74	880
7	0,794	0,030	79	831
8	0,741	0,037	79	961
9	0,776	0,037	78	908
10	0,758	0,037	77	960

Considere que as avaliações estatísticas dos candidatos sobreviventes (3, 6, 7, 8, 9 e 10) excluem do conjunto de configurações candidatas as alternativas de qualidade inferior (3, 7, 8 e 9), restando somente dois candidatos (6 e 10). Assim, o candidato de melhor qualidade é, então, definido como aquele com o menor valor médio das classificações, ou seja, o candidato 6 ($R = 1,33$) (Tabela 10).

Tabela 10 - Desempenho das configurações candidatas sobreviventes.

Ciclo	Configurações Candidatas					
	3 (r)	6 (r)	7 (r)	8 (r)	9 (r)	10 (r)
$k = 1$	333 (3)	203 (1)	392 (4)	707 (6)	522 (5)	288 (2)
$k = 2$	483 (5)	121 (2)	286 (3)	651 (6)	089 (1)	450 (4)
$k = 3$	723 (4)	244 (1)	898 (5)	434 (2)	968 (6)	507 (3)
R	4	1,33	4	4,67	4	3

De acordo com o algoritmo HORA, a sintonização ideal dos parâmetros da meta-heurística GA para resolução do problema de otimização ilustrativo é a seguinte:

- $p_c = 0,784$;
- $p_m = 0,039$;
- $\mu = 74$; e
- $n = 880$.

5.3 A ESTRUTURA HEURÍSTICA DO ALGORITMO HORA

A estrutura heurística adotada nesta versão do algoritmo HORA (Seção 5.2) foi inspirada em um método semelhante ao método empregado pela meta-heurística ILS para criação das soluções candidatas. Em síntese, a ILS explora o espaço de soluções de um problema por meio de perturbações em uma solução ótima local.

No HORA, as configurações candidatas também são criadas a partir de modificações da alternativa com o melhor desempenho no conjunto de treinamento. O processo de criação é dinâmico, pois os parâmetros que serão modificados para definir as novas alternativas são sorteados. Os parâmetros sorteados são modificados ao acaso por meio de adições ou subtrações arbitrárias de valores aleatórios criados durante a execução do algoritmo (Seção 5.2.1). Estas modificações definem as novas configurações candidatas obtidas a partir da configuração com o melhor desempenho no conjunto de treinamento.

Assim como ocorre com os algoritmos meta-heurísticos, a estrutura heurística do HORA pode ser adaptada, de modo que outros métodos de busca podem ser empregados nas futuras versões. Por exemplo, é possível adaptar o método de buscas da VNS ao HORA visando criar novas configurações candidatas a partir de modificações sistemáticas na estrutura de vizinhança de uma alternativa conhecida e concentrar as buscas sobre as configurações que melhoram a qualidade das soluções. De maneira semelhante, outros métodos de diferentes meta-heurísticas podem ser adaptados ao método HORA, por exemplo GA, SA, Busca por Agrupamento (em Inglês, *Clustering Search* - CS), dentre outros.

Pode-se inclusive imaginar que, sendo a meta-heurística alvo do HORA uma meta-heurística *A* qualquer, o próprio método de busca de *A* poderia ser a estrutura heurística do HORA. Por exemplo, se a SA é a meta-heurística alvo a ser sintonizada pelo algoritmo HORA, a estrutura heurística do HORA poderia ser adaptada de modo a aceitar configurações candidatas de qualidade inferior com certa probabilidade. Esta característica permite considerar o algoritmo HORA como uma meta-meta-heurística.

Na literatura de meta-heurísticas é possível encontrar diversos métodos de sintonização, dentre os quais destacam-se CALIBRA (ADESON-DÍAZ e LAGUNA, 2006), F-Race (BIRATTARI et al., 2002) e sua versão iterativa I/F-Race (BALAPRAKASH et al., 2007) e ParamILS (HUTTER et al., 2009).

O CALIBRA combina o sistema de arranjos ortogonais de Taguchi e heurísticas de busca local para sintonização de algoritmos. Em síntese, seu procedimento consiste em identificar uma região promissora por meio dos arranjos ortogonais e explora-la com uma heurística até encontrar uma solução ótima local. Os limites dos parâmetros são ajustados para a nova região. Este procedimento se repete até que não se observem alterações significativas nos parâmetros. A sintonização de algoritmos por meio do F-Race é realizada a partir da definição *a priori* de um conjunto de configurações candidatas. As alternativas definidas neste conjunto são avaliadas durante as corridas com a estatística não paramétrica de Friedman. A corrida segue até atingir um critério de parada ou até que restem duas alternativas no conjunto

de configurações candidatas. O I/F-Race é uma versão iterativa do F-Race, cujo processo de sintonização se resume na aplicação iterativa do F-Race sobre configurações candidatas geradas de acordo com um modelo probabilístico. No ParamILS, o processo de sintonização de algoritmos inicia-se com a avaliação de uma configuração inicial. Em seguida, esta configuração é perturbada em um ou mais parâmetros, avaliada e usada como ponto inicial de um método de intensificação para encontrar o ótimo local. O procedimento é repetido até atingir um critério de parada, em geral, o tempo de execução. Durante este processo as configurações candidatas são avaliadas sobre diversos problemas e seu desempenho armazenado.

As características dos diferentes métodos de sintonização encontrados na literatura podem ser identificadas no método HORA. Isto é, a utilização de DOE na definição do espaço de busca de parâmetros, o conceito de corrida na avaliação das configurações candidatas e a criação de alternativas a partir de perturbações em uma configuração promissora. Entretanto, o diferencial do HORA em relação aos métodos supracitados consiste em combinar estas características em um método heurístico, cuja estrutura pode ser adaptada a diferentes métodos de busca e a exploração do espaço de busca de parâmetros é realizada de maneira dinâmica a partir da criação de configurações candidatas na vizinhança de uma configuração promissora.

6. PROBLEMAS DE OTIMIZAÇÃO COMBINATÓRIA E ESTUDOS COMPUTACIONAIS

6.1 INTRODUÇÃO

Os problemas de otimização são comuns em muitas áreas, por exemplo ciência, engenharia, administração e negócios, e em diferentes domínios. Essencialmente, esses problemas envolvem encontrar um valor extremo (mínimo ou máximo), chamado ótimo, de uma função com inúmeros extremos locais, denominada função objetivo.

A solução de um problema de otimização pode ser representada por um vetor $s = (s_1, s_2, \dots, s_n)$, em que cada uma das n variáveis s_i é denominada variável de decisão e o conjunto de todas as soluções corresponde ao espaço de buscas S , em que $s \in S$.

As variáveis de decisão se dividem em duas categorias: aquelas que assumem valores reais ($s_i \in \mathbb{R}$), denominadas contínuas, e aquelas que assumem valores discretos ($s_i \in \mathbb{Z}$). Consequentemente, um problema de otimização pode ser contínuo, se as variáveis de decisão são contínuas, combinatório, se as variáveis de decisão são um conjunto discreto, ou mistos, se as variáveis de decisão contêm valores contínuos e discretos.

Um problema de otimização combinatória pode ser definido pelo par (S, f) , em que S representa o conjunto de soluções viáveis e $f: S \rightarrow \mathbb{R}$ é uma função objetivo, que atribui um valor real para cada solução $s \in S$. O problema consiste em encontrar uma solução ótima (s^*), tal que:

$$f(s^*) \leq f(s), \text{ para } \forall s \in S, \quad (18)$$

no caso de problemas de minimização (18) ou

$$f(s^*) \geq f(s), \text{ para } \forall s \in S, \quad (19)$$

no caso de problemas de maximização (19).

Em ambos os casos, s^* é um valor extremo de f quando comparado com as demais soluções do espaço de buscas S .

A presente pesquisa está focada em problemas de otimização combinatória. Nesse contexto, este capítulo apresenta uma breve introdução sobre problemas de otimização combinatória e dois problemas clássicos da literatura que serão utilizados nos estudos

computacionais desta pesquisa: o Problema do Caixeiro Viajante (Seção 6.2) e o Problema de Sequenciamento de Atividades (Seção 6.3). Neste capítulo são apresentados, também, os resultados computacionais obtidos a partir da aplicação da metodologia de sintonização de meta-heurísticas proposta nesta pesquisa para ajustes das meta-heurísticas SA, GA e VNS, considerando-se diferentes exemplares dos problemas clássicos de otimização combinatória abordados.

6.1.1 Propriedades dos Problemas de Otimização

O desafio na resolução eficiente de problemas de otimização combinatória é encontrar soluções de boa qualidade em tempo aceitável, independente do método utilizado (por exemplo, pesquisa local, métodos exatos ou heurísticas). A dificuldade surge quando o número de alternativas no espaço de buscas é elevado. Por exemplo, no Problema do Caixeiro Viajante (em Inglês, *Traveling Salesman Problem* - TSP) há um conjunto com n cidades e uma matriz $n \times n$ de distâncias entre as cidades. O objetivo é visitar todas as cidades uma única vez e voltar à cidade de partida, de forma a minimizar a distância total percorrida. Uma vez escolhida a cidade de partida, o tamanho do espaço de buscas desse problema é $(n - 1)!/2$ (MATAI et al., 2010). Assim o número de percursos para um conjunto com 5, 10 e 75 cidades é 12, $1,814 \times 10^5$ e $1,654 \times 10^{107}$, respectivamente. Portanto, muitas vezes a enumeração de todas as soluções é inviável para um problema de otimização combinatória.

A dificuldade de um problema de otimização combinatória está associada à teoria da complexidade de algoritmos (COOK, 1971). A teoria da complexidade computacional é um ramo de estudos da Ciência da Computação e Matemática que trata problemas de decisão, isto é, aqueles problemas nos quais as respostas possíveis são sim ou não. Um problema de otimização pode ser reduzido a um problema de decisão. Por exemplo, no TSP o problema de otimização é encontrar um percurso para minimizar a distância total percorrida, enquanto que o problema de decisão correspondente é descobrir se existe um percurso, cuja distância total percorrida é menor ou igual a uma dada distância L .

A complexidade de um algoritmo é a quantidade de recursos computacionais, por exemplo tempo e memória, exigidos para resolver um problema específico. A dificuldade ou complexidade de um problema é equivalente à complexidade do melhor algoritmo para resolvê-lo, ou seja, aquele com o menor esforço possível (TALBI, 2009). Um importante

aspecto da teoria computacional é categorizar os problemas de otimização em diferentes grupos, denominados classes de complexidade, com base em sua dificuldade.

Um problema é considerado fácil ou tratável se existe um algoritmo para resolvê-lo de maneira eficaz em tempo polinomial. Por outro lado, um problema é difícil ou intratável se nenhum algoritmo pode resolvê-lo em tempo polinomial. As classes de complexidade reúnem um conjunto de problemas com características semelhantes, que podem ser resolvidos com os mesmos recursos computacionais. Existem duas classes importantes de problemas: P e NP (TALBI, 2009; ROTHLAUF, 2011).

A classe de complexidade P, em que P significa polinomial, representa o conjunto de todos os problemas de decisão que podem ser resolvidos por um algoritmo determinístico em tempo polinomial. Um algoritmo determinístico é polinomial para um problema de decisão, se seu consumo de tempo no pior caso é assintoticamente limitado por uma função polinomial. Portanto, todos os problemas em P podem ser resolvidos de maneira eficiente, no pior caso. Alguns problemas clássicos dessa classe são: árvore geradora mínima, caminho mínimo, fluxo máximo em rede, dentre outros.

A classe de complexidade NP, em que NP significa tempo polinomial não determinístico, representa o conjunto de todos os problemas de decisão que podem ser resolvidos por um algoritmo não determinístico em tempo polinomial. Um algoritmo não determinístico contém pontos de decisão, em que diferentes continuações são possíveis.

Os problemas de decisão da classe P são tratáveis e podem ser resolvidos com um algoritmo correto. Todos os problemas em P também estão na classe NP, por isso $P \subseteq NP$. Uma questão em teoria da complexidade computacional é se P é um subconjunto próprio de NP, isto é $P \subset NP$, ou se $P = NP$. Atualmente, assume-se que $P \neq NP$ e por isso existem problemas difíceis em NP, que não estão em P, uma vez que não há algoritmos determinísticos de tempo polinomial para resolvê-los.

Os problemas de decisão classificados como NP-completos (em Inglês, *NP-complete*) são um subconjunto da classe NP. Um problema é NP-completo se este problema pertence tanto à classe NP quanto à classe NP-difícil (em Inglês, *NP-hard*). Uma característica importante dos problemas NP-completos é que se for encontrado um algoritmo de tempo polinomial para sua resolução, então existirá um algoritmo de tempo polinomial para qualquer problema da classe NP.

Os problemas NP-difíceis são problemas de otimização, cujos problemas de decisão associados são da classe NP-completos. Um problema da classe NP-difícil é, no mínimo, tão

difícil quanto o mais difícil problema da classe NP. Mais precisamente, um problema p é NP-difícil se todo problema $q \in \text{NP}$ pode ser reduzido em tempo polinomial a p .

O problema de satisfação (em Inglês, *satisfiability problem*) é NP-completo (COOK, 1971). Muitos problemas clássicos de otimização combinatória são NP-difíceis, por exemplo: sequenciamento de tarefas em máquinas, localização de facilidades, p -medianas, agrupamento de dados, coloração de grafos, rotas de veículos, cobertura de conjuntos, corte e empacotamento, entre outros. Para estes problemas, não se conhece algoritmos polinomiais eficientes para resolução.

6.2 PROBLEMA DO CAIXEIRO VIAJANTE

A área de roteamento (em Inglês, *routing*) é uma das mais estudadas em Matemática Computacional. Em síntese, um problema de rotas inclui o planejamento de itinerários para percorrer diferentes localizações, tais como a programação de entregas de combustíveis e o recolhimento de resíduos em vias, bem como a otimização de problemas de fabricação, por exemplo a determinação de sequências de perfuração em placas de circuitos impressos.

Um dos problemas de rotas mais célebres é o Problema do Caixeiro Viajante (TSP). A simplicidade da concepção e a aparente intratabilidade fazem o TSP desempenhar um papel de referência na literatura. As principais técnicas algorítmicas em otimização combinatória têm o TSP como um dos problemas de *benchmark* motivadores. Exceto para alguns casos especiais, a resolução do TSP é difícil, tal qual o problema de determinar um ciclo Hamiltoniano¹³ em um grafo.

A literatura relacionada ao TSP é extensa devido à sua relevância prática (APPLEGATE, 2007; LAPORTE, 2007; DERIGS, 2009; DAVENDRA, 2010), bem como por ele ocorrer como subproblema de diversos problemas de rotas. Em Angel et al. (1972) o TSP é tratado como um problema de rotas para programação de linhas de ônibus, onde o objetivo é minimizar a distância total percorrida, evitar a lotação dos carros e não exceder a política de tempo para percorrer os itinerários. Uma variação do TSP, denominada TSP com restrição de recursos, tem aplicação no problema de sequenciamento de tarefas com prazo de entrega agregado (MILLER e PEKONY, 1991). O TSP com coleta de prêmio (BALAS, 1989) e o problema de orientação (GOLDEN et al., 1987) são casos especiais do TSP com restrição de

¹³ Um ciclo Hamiltoniano é um percurso que utiliza uma única vez cada vértice de um grafo.

recursos. O problema de encontrar um percurso de comprimento máximo é o objetivo no MAX TSP (BARVINOK et al., 2003). O TSP com máxima dispersão consiste no problema de calcular o percurso de um conjunto de cidades, a fim de maximizar o comprimento mínimo do percurso. Ele é motivado por aplicações nas áreas de manufatura e imagens médicas (ARKIN et al., 1996).

6.2.1 Definição e Formulação Matemática

O enunciado genérico do TSP consiste em um conjunto com n cidades $C = \{1, 2, \dots, n\}$, em que a cada par de cidades (i, j) é associado a uma distância d_{ij} , tal que $i, j \in C$ com $i \neq j$. O objetivo é visitar todas as cidades uma única vez e voltar à cidade de partida, de forma a minimizar a distância total percorrida. A maneira de visitar todas as cidades corresponde à sequência em que elas são visitadas, enquanto que sua ordenação representa o ciclo ou percurso. O aumento no número de cidades torna a determinação dos percursos mais complexa. O problema de decisão correspondente ao TSP pertence à classe NP-completo.

Em geral, o TSP é classificado como simétrico se $d_{ij} = d_{ji}, \forall i, j \in C$, ou assimétrico, se $d_{ij} \neq d_{ji}$ para ao menos um par de cidades. O problema pode ser definido em um grafo não direcionado completo $G = (C, E)$ se for simétrico ou em um grafo direcionado $G = (C, A)$ se for assimétrico. Nessa formulação, o conjunto de cidades C representa os vértices, $E = \{(i, j) \mid i, j \in C, i < j\}$ são as arestas e $A = \{(i, j) \mid i, j \in C, i \neq j\}$, os arcos. A matriz de distâncias D entre as cidades, definida sobre E e A , satisfaz a desigualdade triangular sempre que $d_{ij} \leq d_{ik} + d_{kj}$ para todo i, j e k . Em particular, é o caso de problemas planares em que os vértices são pontos $P = (x_i, y_i)$ no plano e $d_{ij} = [(x_i - x_j)^2 + (y_i - y_j)^2]^{1/2}$ é a distância euclidiana entre duas cidades i e j . A desigualdade triangular também é satisfeita se d_{ij} é o comprimento do caminho mais curto de i para j em G (MATAI et al., 2011).

A formulação do modelo de otimização do TSP simétrico (20-23) proposta em (DANTZIG et al., 1954) é uma das mais referenciadas na literatura:

$$\text{minimizar } \sum_{i < j} d_{ij} c_{ij} , \quad (20)$$

Subjeito à:

$$\sum_{i < k} c_{ik} + \sum_{j > k} c_{jk} = 2, \forall k \in C, \quad (21)$$

$$\sum_{(i,j) \in S} c_{ij} \leq |S| - 1 \quad (S \subset C, 3 \leq |S| \leq n - 3), \quad (22)$$

$$c_{ij} \in \{0, 1\}, \forall i, j \in E, \quad (23)$$

em que d_{ij} corresponde à distância ou custo do percurso entre as cidades i e j e c_{ij} é uma variável de decisão que assume valor 1 se a aresta (i, j) aparece no percurso ótimo, isto é, se existe um caminho entre as cidades i e j , e valor 0 (zero) caso contrário. A restrição (21), denominada restrição de grau, garante a existência de uma ligação única entre as cidades, (22) é a restrição de eliminação de subpercurso, que em conjunto com (21) exclui os subpercursos de C e (23) é uma restrição de integralidade. Essa formulação pode ser estendida para o TSP assimétrico (24-28):

$$\text{minimizar } \sum_{i \neq j} d_{ij} c_{ij}, \quad (24)$$

Subjeito à:

$$\sum_{j=1}^n c_{ij} = 1, \forall i \in C, \quad (25)$$

$$\sum_{i=1}^n c_{ij} = 1, \forall j \in C, \quad (26)$$

$$\sum_{(i,j) \in S} c_{ij} \leq |S| - 1 \quad (S \subset C, 2 \leq |S| \leq n - 2), \quad (27)$$

$$c_{ij} \in \{0, 1\}, \forall i, j \in A. \quad (28)$$

Nessa formulação, (25-26), (27) e (28) são denominadas restrições de grau, de eliminação subpercurso e de integralidade, respectivamente.

6.3 PROBLEMA DE SEQUENCIAMENTO DE ATIVIDADES

Problemas de Sequenciamento (em Inglês, *Scheduling Problems*) estão relacionados com a distribuição de recursos limitados para execução de atividades visando a realização eficiente de trabalhos. Esses problemas envolvem atividades que devem ser organizadas em máquinas, geralmente, com algumas limitações, para otimizar um critério de desempenho. A ideia-chave é encontrar uma ordem para execução das atividades e determinar quando e em qual máquina cada atividade deve ser executada. Essa concepção simples cria um problema

que permanece em grande parte sem solução eficiente. Estudos conduzidos nas últimas décadas indicam que a complexidade de alguns problemas de sequenciamento é NP-difícil (SCHMIDT, 2000).

A teoria de sequenciamento assume que as máquinas são continuamente disponíveis. Essa hipótese pode ser justificada em alguns casos, mas não se aplica em situações com determinados requisitos de manutenção, avarias e outras limitações, que implicam na indisponibilidade das máquinas para processamento e forçam a interrupção das atividades. Um sequenciamento de atividades é preferencial (em Inglês, *preemptive*) se, a qualquer instante, as atividades podem ser interrompidas e reiniciadas mais tarde sem custos, algumas vezes, em outra máquina. Se a interrupção de atividades não é permitida, o sequenciamento é denominado não preferencial.

Em geral, os problemas de sequenciamento são descritos por meio de uma notação de três campos: $\alpha|\beta|\gamma$ (GRAHAM et al., 1979; SCHMIDT, 2000), em que α se refere a características da máquina, β descreve as atividades e características de processamento e γ corresponde à medida de desempenho que será otimizada.

Na literatura contemporânea um dos objetivos mais amplamente estudados no contexto de sequenciamento de atividades é a minimização do *makespan*. Esse problema é expresso como $F|pmtn|C_{\max}$ pela notação de três campos, em que F indica o uso de máquinas dedicadas, $pmtn$ descreve atividades preferenciais, isto é, aquelas que podem ser arbitrariamente interrompidas, e C_{\max} é a medida de desempenho referente ao *makespan* (PINEDO, 2002). Em síntese, um problema de *makespan* consiste em encontrar uma sequência de execução das atividades, tal que o tempo de conclusão da última atividade executada seja o menor possível.

Outras medidas de desempenho também são muito estudadas em sequenciamento de atividades, especialmente, aquelas relacionadas com datas para início e finalização das atividades, muito significativas para a indústria devido à necessidade de cumprimento de prazos. Dentre os problemas de sequenciamento com restrições de data, destaca-se o Problema do Atraso Total (em Inglês, *Total Tardiness Problem* - TTP), cujo objetivo é determinar uma sequência para execução das atividades, tal que o atraso total T_{\max} seja minimizado. O atraso de uma atividade corresponde à diferença entre a data programada para sua finalização e a finalização efetiva da atividade (KOULAMAS, 2010).

6.3.1 Definição e Formulação Matemática

O problema TTP, expresso na forma de $\|1, d\|T_{\max}$, é definido como um conjunto com n atividades $J = \{1, 2, \dots, n\}$ para serem processadas sem interrupção por uma máquina única continuamente disponível, com capacidade para processar no máximo uma atividade de cada vez. Cada atividade ($j \in J$) está disponível no instante inicial, consome um tempo de processamento p_j (unidades de tempo) positivo e ininterrupto e possui restrições em relação à data de finalização d_j . O atraso T_j é computado como $\max(0, C_j - d_j)$, em que C_j é tempo de completude da atividade j .

Uma extensão do TTP pode ser obtida com a atribuição de diferentes prioridades w_j para a execução das atividades. Essa modificação resulta no TTP Ponderado (em Inglês, *Total Weighted Tardiness Problem* - TWTP), cujo objetivo é organizar as atividades de maneira a encontrar uma sequência ideal para minimizar o atraso total ponderado $\sum w_j T_j$. Expresso sob forma de $\|1, d_j\|w_j T_j$, no TWTP as atividades $j \in J$ possuem prioridades w_j umas sobre as outras. Por exemplo, na Tabela 11 são apresentados dados ilustrativos de 4 atividades para um problema de sequenciamento. Cada atividade consome $p = [6, 8, 3, 9]$ unidades de tempo de processamento e possui restrições em relação à data de finalização $d = [10, 16, 8, 18]$ e prioridade $w = [1, 5, 4, 3]$ sobre as demais atividades.

Tabela 11 - Problema ilustrativo de sequenciamento.

Atividades	1	2	3	4
p	6	8	3	9
d	10	16	8	18
w	1	5	4	3

Para ilustrar o cálculo do atraso total ponderado admita-se, inicialmente, a sequência de processamento: 1-2-3-4. O processamento das atividades é iniciado com a atividade 1, que consome 6 unidades de tempo sem atraso. Isto é, a atividade 1 se encerra no instante 6 e sua data de finalização é o instante 10, por isso $T_1 = \max(0, 6 - 10) = 0$. O processamento segue com a atividade 2 imediatamente após 1. O processamento total das primeiras atividades consome 14 unidades de tempo ($p_1 + p_2 = 6 + 8 = 14$) sem atraso, pois a finalização da atividade 2 está programada para o instante 16. Ao seguir o processamento, a atividade 3 é finalizada no instante 17 e, como seu término está programado para o instante 8, ocorre atraso de 9 unidades de tempo, isto é, $T_3 = \max(0, 17 - 8) = 9$. A sequência de processamento é finalizada com a atividade 4 no instante 26, com atraso de 8 unidades de tempo. Ao final do

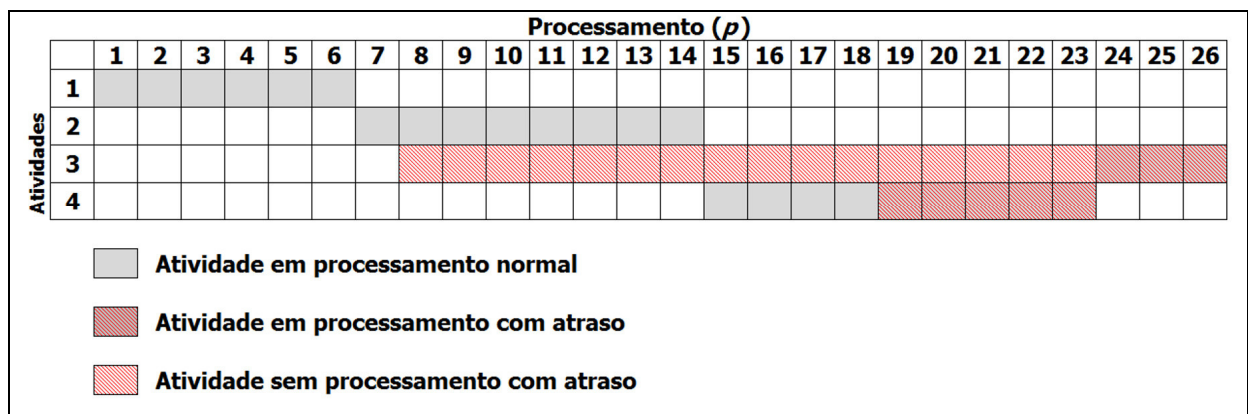
processamento, a sequência 1-2-3-4 produz um atraso total ponderado de 60 unidades de tempo, isto é, $\sum w_j T_j = 1 \times 0 + 5 \times 0 + 4 \times 9 + 3 \times 8 = 60$ (Tabela 12).

Tabela 12 - Atraso total ponderado para a sequência 1-2-3-4.

Atividades	1	2	3	4
p	6	8	3	9
d	10	16	8	18
w	1	5	4	3
C	6	14	17	26
T	0	0	9	8
wT	0	0	36	24

Uma pequena alteração na sequência de processamento das atividades pode modificar completamente o resultado. Por exemplo, a sequência de processamento 1-2-4-3 (Figura 26) produz um atraso total ponderado de 87 unidades de tempo.

Figura 26 - Processamento das atividades para a sequência 1-2-4-3.



Em geral, em problemas de sequenciamento com n atividades e m máquinas é possível obter $(n!)^m$ diferentes combinações de sequenciamento. Portanto, no TWTP ilustrativo (Tabela 12), em que $n = 4$ atividades e $m = 1$ máquina, há $4! (= 24)$ maneiras diferentes de organizar a sequência das atividades. O TWTP em máquina única é um caso especial de problemas de sequenciamento classificado como NP-difícil (LENSTRA et al., 1977).

O TWTP é relevante em situações reais, tais como sequenciamento da produção e de pousos e decolagens de aeronaves, designação da sequência de etapas em projetos de desenvolvimento, entregas de mercadorias para clientes com prioridades na cadeia de suprimentos, dentre outros. Ao longo das últimas décadas, muitos estudos propõem diferentes abordagens para resolver o TWTP. O método exato *branch-and-bound* (JOUQUET et al.,

2002; BABU et al., 2004) permite encontrar a solução ótima do problema, mas em geral demanda o consumo excessivo dos recursos computacionais, por exemplo tempo, especialmente quando o tamanho do problema ultrapassa 50 atividades (CRAUWELS et al., 1998). Outros estudos empregam métodos heurísticos e meta-heurísticos e incluem pesquisa local (CRAUWELS et al., 1998), busca tabu (BOZEJKO et al., 2006), recozimento simulado (NEARCHOU, 2004), algoritmo genético (LIU et al., 2005), dentre outros (ARROIO et al., 2008; WANG e TANG, 2009).

O modelo de otimização é formulado da seguinte maneira (29-34):

$$\text{minimizar } \sum_{j=1}^n w_j T_j, \quad (29)$$

Subjeito à:

$$T_j \geq C_j - d_j, \forall j \in J, \quad (30)$$

$$C_j \geq d_j, \forall j \in J, \quad (31)$$

$$C_j + p_k \leq C_k, \forall j, k \in J, \quad (32)$$

$$C_k + p_j \leq C_j, \forall j, k \in J, \quad (33)$$

$$T_j, C_j \geq 0, \forall j \in J, \quad (34)$$

em que as restrições (30) restringem que o atraso seja maior ou igual a diferença entre o tempo programado para finalização da atividade e sua finalização efetiva, (31) garantem que o tempo de completude da atividade seja maior ou igual ao seu tempo de processamento, (32) e (33) são restrições que forcem que uma atividade j seja processada antes de outra atividade k e vice-versa para qualquer par de atividades, e (34) estabelecem que o atraso e o tempo de completude da atividade devem ser positivos.

6.4 ESTUDOS COMPUTACIONAIS

Os estudos computacionais da presente pesquisa foram realizados em um computador pessoal com processador Intel Core i7 3.6GHz, memória RAM de 8GB e sistema operacional Windows 7 64 bits. Os programas referentes aos métodos HORA e Corrida foram desenvolvidos em ambiente de computação científica gratuito Scilab (URL: <http://www.scilab.org>).

A metodologia para sintonização de meta-heurísticas proposta nesta pesquisa foi aplicada no ajuste fino de três meta-heurísticas distintas (SA, GA e VNS) a partir de dois estudos computacionais. Em um dos estudos, aplicou-se o método HORA na sintonização dos algoritmos supracitados, enquanto que no outro, utilizou-se o método de Corrida. Todas as meta-heurísticas foram sintonizadas para resolução de dois problemas clássicos de otimização: TSP e TWTP.

Para realização dos estudos de sintonização utiliza-se um conjunto de treinamento formado por n exemplares, em que $n > 1$. Nesta pesquisa, os exemplares que formam os conjuntos de treinamento foram selecionados arbitrariamente dos *benchmarks* TSP da biblioteca TSPLIB (REINELT, 1991) para o TSP, e wt40, wt50 e wt100 da biblioteca OR-Library (BEASLEY, 1990) para o TWTP com 40, 50 e 100 atividades, respectivamente.

Nos estudos experimentais com DOE, os parâmetros das meta-heurísticas são associados aos fatores e seus valores aos níveis. Os estudos iniciais por meio de planejamentos fatoriais 2^k são exploratórios e visam identificar os fatores que influenciam o desempenho das meta-heurísticas, bem como a existência de interações entre eles.

Em seguida, foram realizados estudos de otimização por meio da Metodologia da Superfície de Respostas a partir de um CCD, com a finalidade de detectar as configurações de parâmetros próximas de extremos (máximo ou mínimo).

Os estudos experimentais com DOE foram repetidos para cada um dos exemplares que formam o conjunto de treinamento, tal que o resultado final são n alternativas distintas de configurações, cada uma relacionada a um exemplar. As configurações identificadas permitem definir um intervalo de valores para cada parâmetro, denominado espaço de busca de parâmetros, cujos limites são os valores máximo e mínimo de cada parâmetro identificados pelos estudos experimentais.

A sequência da metodologia consiste em aplicar o método HORA para ajuste fino dos parâmetros das meta-heurísticas. Nesta pesquisa, a cada ciclo do método heurístico são criadas 4 configurações candidatas arbitrárias no espaço de busca de parâmetros e, para cada uma delas a meta-heurística é executada durante 15 segundos sobre um conjunto de treinamento expandido. Nesta pesquisa, o conjunto de treinamento expandido corresponde a todos os exemplares que compõem os *benchmarks* de cada um dos problemas de otimização selecionados.

Para avaliar a qualidade dos ajustes obtidos com o HORA, aplicou-se o método de Corrida sobre os mesmos algoritmos, conjuntos de treinamento e espaços de busca de parâmetros. No método de Corrida, cada parâmetro é definido em um intervalo discreto com

número de níveis arbitrários no espaço de busca, em geral, resultando em uma quantidade relativamente grande de configurações candidatas. A meta-heurística foi executada durante 15 segundos sobre o mesmo conjunto de treinamento expandido utilizado pelo HORA.

O processo de sintonização com os métodos HORA e Corrida foi repetido 10 vezes. Os resultados das sintonizações são apresentados em termos de média e desvio padrão ($\mu \pm \sigma$) calculados após dez repetições do processo de sintonização.

O Apêndice A apresenta a sequência de passos da metodologia para sintonização de meta-heurísticas aplicada no ajuste fino da meta-heurística GA para o *benchmark* wt40 do TWTP.

As próximas Seções (6.4.1, 6.4.2 e 6.4.3) apresentam as sintonizações das meta-heurísticas SA, GA e VNS pelos métodos HORA e Corrida para os problemas clássicos de otimização combinatória: TSP e TWTP.

As meta-heurísticas, em geral, empregam algum grau de aleatoriedade para diversificar as buscas e evitar o confinamento em certas áreas do espaço de buscas. Por isso, uma única execução desses algoritmos pode resultar em uma solução diferente da próxima execução. Diante deste contexto, os resultados computacionais desta pesquisa foram produzidos após 5 execuções das meta-heurísticas SA, GA e VNS para os dois problemas selecionados. Estes resultados são apresentados em termos dos seguintes parâmetros:

- sol: melhor solução encontrada pela meta-heurística;
- desvio: erro relativo entre a solução média e a melhor solução encontrada pela meta-heurística;
- gap: erro relativo entre a melhor solução encontrada pela meta-heurística e a melhor solução conhecida;
- t: tempo médio em segundos após 5 execuções da meta-heurística; e
- delta: erro relativo entre a melhor solução obtida do método HORA e a melhor solução do método de Corrida.

O erro relativo entre a solução média e a melhor solução encontrada pela meta-heurística é dado como:

$$desvio = \left(\frac{sol_m - sol}{sol} \right) \times 100, \quad (35)$$

em que sol_m é a solução média computada após 5 execuções da meta-heurística e sol é a melhor solução encontrada pela meta-heurística.

O erro relativo entre a melhor solução encontrada pela meta-heurística e a melhor solução conhecida é computado com:

$$gap = \left(\frac{sol - sol^*}{sol^*} \right) \times 100, \quad (36)$$

em que sol é a melhor solução encontrada pela meta-heurística e sol^* é a melhor solução conhecida.

O erro relativo entre a melhor solução obtida do método HORA e a melhor solução do método de Corrida é calculado por meio de:

$$delta = \left(\frac{sol_H - sol_C}{sol_C} \right) \times 100, \quad (37)$$

em que sol_H é a melhor solução obtida com o método HORA e sol_C é a melhor solução obtida com o método de Corrida.

Nas tabelas de resultados completos, os valores em destaque (negrito) correspondem aos exemplares nos quais o método HORA encontra soluções de qualidade superior em relação ao método de Corrida, em termos da função objetivo.

Os resultados computacionais foram avaliados estatisticamente por meio de testes de hipóteses t bilaterais para duas médias pareadas. No contexto desta pesquisa, cada unidade amostral foi associada a duas medições, sendo uma delas correspondente aos resultados do método HORA e outra, aos resultados do método de Corrida. O nível de significância dos testes de hipóteses foi estabelecido em 10% ($\alpha = 0,10$) e as hipóteses nula (H_0) e alternativa (H_a) foram enunciadas em termos do desempenho das meta-heurísticas (parâmetro "sol") sintonizadas pelos métodos HORA e Corrida:

- $H_0: sol_D = sol_H - sol_C = 0$ (Os desempenhos não se alteram); e
- $H_a: sol_D \neq 0$ (Os desempenhos se alteram).

O número de graus de liberdade para consulta da tabela *t*-Student foi determinado de acordo com o número de exemplares que formam os *benchmarks* dos problemas de otimização selecionados. Isto é, o *benchmark* TSP é formado por 40 exemplares simétricos com distância Euclidiana entre as cidades e número de cidades menor que 500. Os *benchmarks* wt40, wt50 e wt100 de problemas TWTP são formados por 125 exemplares com 40, 50 e 100 atividades, respectivamente.

6.4.1 Sintonização da Meta-heurística SA

O presente estudo trata da sintonização de 3 parâmetros da meta-heurística SA por meio da metodologia para sintonização de meta-heurísticas previamente apresentada. Os parâmetros foram selecionados de acordo com recomendações da literatura e suas configurações iniciais para os estudos experimentais correspondem aos valores adotados com mais frequência em estudos empíricos (Tabela 13).

Tabela 13 - Parâmetros da meta-heurística SA selecionados para sintonização.

Parâmetros	Descrições	Valores
T_0	<i>Valor da temperatura inicial</i>	[10000; 150000]
SA_{max}	<i>Número de iterações a cada estágio da temperatura</i>	[500; 1500]
α	<i>Taxa de resfriamento da temperatura</i>	[0,900; 0,980]

Nas próximas seções são apresentados dois estudos de sintonização, sendo um deles com o método HORA e o outro, com o método de Corrida. A qualidade dos resultados produzidos será comparada a partir da execução da meta-heurística SA sob diferentes sintonizações obtidas por ambos métodos para cada problema de otimização selecionado.

6.4.1.1 Problema do Caixeiro Viajante

As sintonizações da meta-heurística SA por meio dos métodos HORA e Corrida para o TSP são apresentadas na Tabela 14.

Tabela 14 - TSP: Sintonização da meta-heurística SA sob diferentes métodos.

Parâmetros	HORA	Corrida
T_0	139935 ± 42263	138988 ± 41976
SA_{\max}	1584 ± 112	1672 ± 118
α	$0,947 \pm 0,000$	$0,947 \pm 0,000$
Tempo total	696	8338

A partir da comparação dos resultados obtidos pelos métodos HORA e Corrida (Tabela 14) pode-se observar que a sintonização da meta-heurística SA para o *benchmark* TSP encontra-se em uma região restrita do espaço de busca de parâmetros e seus valores parecem convergir para um valor comum. Esta característica é observada para os três parâmetros (T_0 , SA_{\max} e α) são identificados como significantes para o desempenho da meta-heurística SA pelos estudos experimentais.

Destaca-se nestes resultados que os valores de sintonização para o parâmetro SA_{\max} situam-se fora dos limites delimitados inicialmente para os estudos experimentais (Tabela 13). Esta extrapolação de limites, frequentemente, associada aos estudos experimentais com a Metodologia da Superfície de Respostas, é justificada pelos pontos axiais, que podem superar os limites previamente definidos para garantir rotatividade aos planejamentos e estimativas adequadas para os termos quadráticos do modelo de regressão polinomial.

Os resultados demonstram que o método HORA é mais eficaz para obter a sintonização ideal da meta-heurística em termos de tempo de execução, pois demanda uma fração de tempo requerido pelo método de Corrida. Note-se que a principal diferença entre os métodos de sintonização é a criação das configurações candidatas no espaço de busca de parâmetros, que no HORA é dinâmica, uma vez que elas são criadas em tempo de execução durante os ciclos do algoritmo, ao passo que o método de Corrida utiliza um conjunto de valores predefinidos. No entanto, salienta-se que os processos de avaliação são idênticos para ambos métodos.

As sintonizações obtidas são utilizadas na configuração de duas versões da meta-heurística SA, sendo uma delas configurada de acordo com o método HORA (SA_H) e a outra, em conformidade com o método de Corrida (SA_C). Cada versão do algoritmo foi executada cinco vezes sobre 40 exemplares do *benchmark* TSP. Os critérios de parada dos algoritmos são: número máximo de iterações sem alterações no valor da função objetivo (200 iterações) e tempo máximo de execução (300 segundos).

Os resultados computacionais completos das duas versões são apresentados na Tabela 15, cuja estrutura é descrita na Seção 6.4. Nesta tabela, o nome de cada exemplar do *benchmark* TSP aparece acompanhado do número de cidades correspondente.

A análise dos resultados revela que a meta-heurística SA_H produz soluções de melhor qualidade em relação à versão SA_C (Tabela 15). A partir dos resultados é possível observar que a versão SA_H encontra soluções melhores (sol) do que a SA_C para 20 exemplares, ao passo que ambas apresentam as mesmas soluções para 2 exemplares. A qualidade das soluções encontradas (gap) pela meta-heurística sintonizada com o método HORA é, em média, 25% melhor do que as soluções encontradas pela versão sintonizada com o método de Corrida.

No entanto, salienta-se que as soluções encontradas por ambas versões da meta-heurística SA são, em geral, de qualidade razoável e distantes das soluções ótimas conhecidas dos exemplares selecionados. A partir dos resultados pode-se observar que ambas as versões (SA_H e SA_C) encontram a solução ótima de apenas um dos exemplares (*berlin52*).

Os resultados revelam também que a meta-heurística SA_H é mais robusta (desvio) e apresenta melhora de 2,18% na qualidade das soluções (delta) em relação à versão SA_C . Em relação ao tempo médio de execução a SA_H é, em geral, mais lenta ($t \cong 272$ segundos) do que a meta-heurística SA_C ($t \cong 264$ segundos). O déficit de desempenho pode estar relacionado ao parâmetro T_0 (temperatura inicial), que é um pouco maior na versão SA_H e, conseqüentemente, pode demandar mais tempo para encontrar uma solução de qualidade razoável.

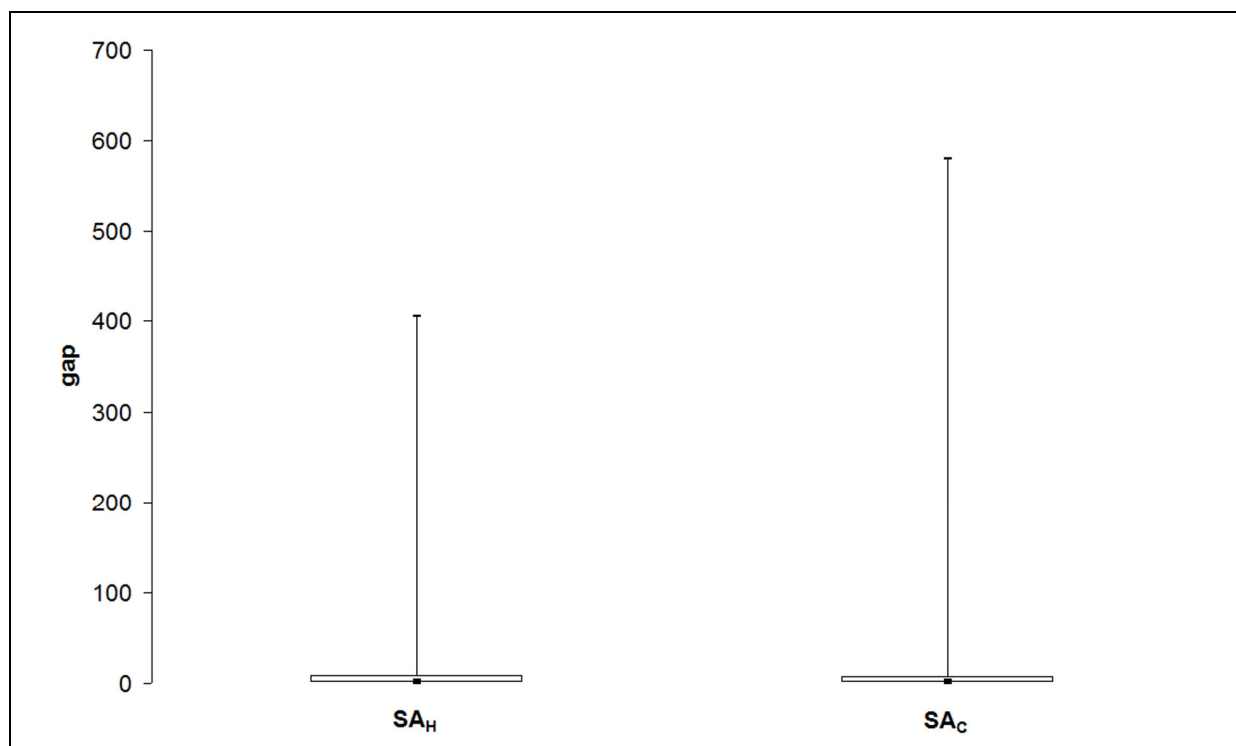
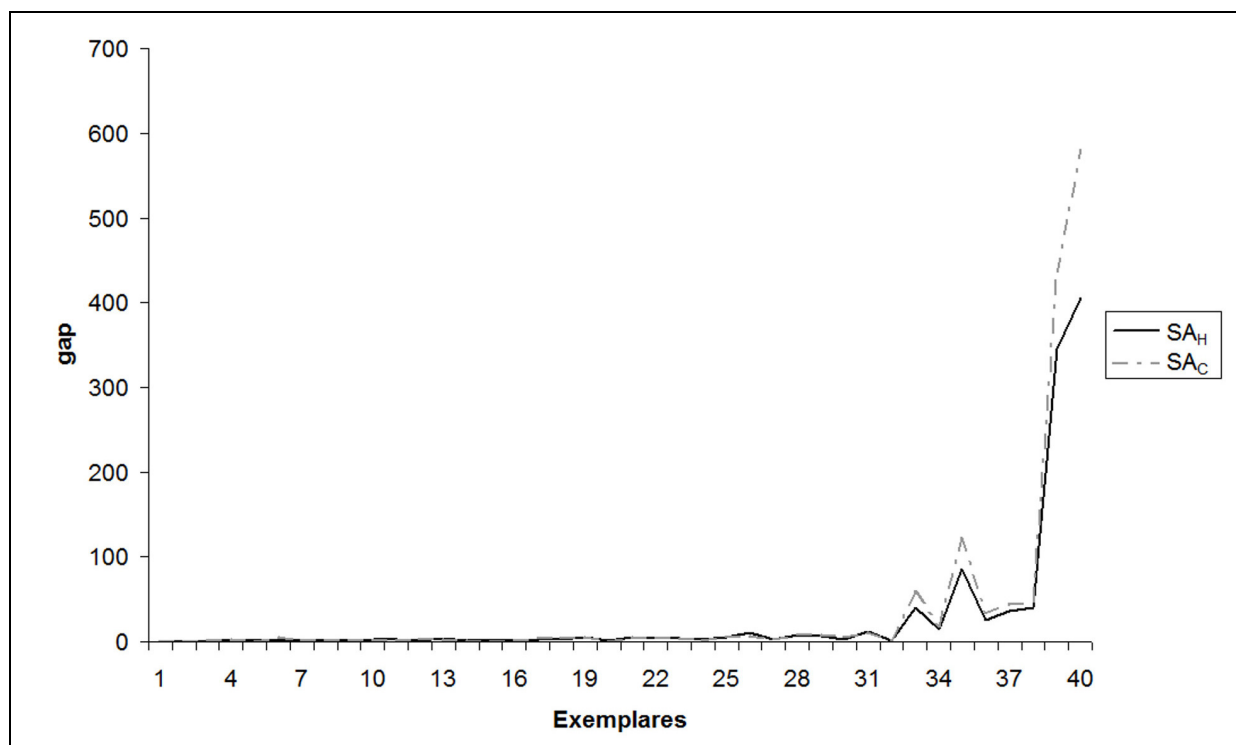
A análise das estatísticas (Figura 27) confirma que as soluções encontradas pela meta-heurística SA_H são melhores do que aquelas encontradas pela SA_C . A partir dos *boxplots* é possível observar que as soluções da SA_H apresentam menor amplitude. Entretanto, ambas versões apresentam assimetrias semelhantes, isto é, positivas e moderadas, sendo a versão SA_H mais assimétrica.

O gráfico da qualidade das soluções encontradas (Figura 28) revela um comportamento semelhante de ambas meta-heurísticas sobre todos os exemplares. Isto é, a qualidade das soluções encontradas é razoável para grande parte dos exemplares do *benchmark* TSP e há perda acentuada de desempenho (qualidade das soluções) das duas meta-heurísticas para os exemplares com número maior de cidades. No entanto, ainda assim, a SA_H apresenta soluções de qualidade superior para a maior parte dos exemplares.

A partir dos resultados computacionais e baseado nos teste de hipóteses, conclui-se com 90% de confiança que a meta-heurística sintonizada pelo método HORA produz resultados melhores do que a versão sintonizada pelo método de Corrida. Neste estudo, observa-se também a ampla vantagem do HORA em relação ao tempo do processo de sintonização.

Tabela 15 - TSP: Resultados computacionais da meta-heurística SA.

Exemplar	melhor	SA _H				SA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
<i>eil51</i>	426	429	0,89	0,70	141	427	1,41	0,23	136	0,47
<i>berlin52</i>	7542	7542	2,33	0,00	98	7542	1,72	0,00	60	0,00
<i>st70</i>	675	685	0,70	1,48	190	686	2,39	1,63	178	-0,15
<i>eil76</i>	538	554	1,41	2,97	214	554	0,87	2,97	192	0,00
<i>pr76</i>	108159	110045	0,56	1,74	185	108589	1,90	0,40	166	1,34
<i>rat99</i>	1211	1244	3,62	2,73	260	1278	1,66	5,53	243	-2,66
<i>kroA100</i>	21282	21510	2,12	1,07	257	21424	2,27	0,67	234	0,40
<i>kroB100</i>	22141	22351	1,94	0,95	261	22363	2,39	1,00	229	-0,05
<i>kroC100</i>	20749	21032	1,53	1,36	257	20983	1,33	1,13	232	0,23
<i>kroD100</i>	21294	21798	2,07	2,37	257	21500	2,83	0,97	239	1,39
<i>kroE100</i>	22068	22582	1,47	2,33	263	22341	2,34	1,24	238	1,08
<i>rd100</i>	7910	8051	2,05	1,78	263	8200	1,01	3,67	245	-1,82
<i>eil101</i>	629	656	1,43	4,29	263	649	2,93	3,18	247	1,08
<i>lin105</i>	14379	14498	2,50	0,83	266	14661	2,09	1,96	256	-1,11
<i>pr107</i>	44303	44702	0,90	0,90	259	44537	1,36	0,53	245	0,37
<i>pr124</i>	59030	60007	1,06	1,66	285	60119	1,46	1,84	271	-0,19
<i>bier127</i>	118282	121974	1,18	3,12	293	122302	1,08	3,40	292	-0,27
<i>ch130</i>	6110	6256	2,53	2,39	297	6395	1,40	4,66	299	-2,17
<i>pr136</i>	96772	102079	1,00	5,48	298	101297	1,37	4,68	296	0,77
<i>pr144</i>	58537	59242	2,89	1,20	299	58850	1,16	0,53	299	0,67
<i>ch150</i>	6528	6839	1,34	4,76	299	6834	2,75	4,69	299	0,07
<i>kroA150</i>	26524	27704	1,47	4,45	300	27490	1,92	3,64	299	0,78
<i>kroB150</i>	26130	27144	1,74	3,88	300	27079	1,43	3,63	299	0,24
<i>pr152</i>	73682	75427	1,70	2,37	299	74695	1,94	1,37	299	0,98
<i>u159</i>	42080	44312	3,91	5,30	299	44227	2,02	5,10	299	0,19
<i>rat195</i>	2323	2565	2,42	10,42	299	2493	14,92	7,32	300	2,89
<i>d198</i>	15780	16288	1,15	3,22	300	16282	1,61	3,18	299	0,04
<i>kroA200</i>	29368	31690	1,97	7,91	300	31861	1,95	8,49	299	-0,54
<i>kroB200</i>	29437	31412	2,97	6,71	299	31727	1,89	7,78	299	-0,99
<i>ts225</i>	126643	130731	4,39	3,23	300	133380	2,54	5,32	299	-1,99
<i>tsp225</i>	3916	4360	8,94	11,34	300	4313	35,74	10,14	299	1,09
<i>pr226</i>	80369	81651	3,89	1,60	300	81830	6,93	1,82	299	-0,22
<i>gil262</i>	2378	3329	56,20	39,99	300	3750	118,26	57,70	299	-11,23
<i>pr264</i>	49135	56626	5,48	15,25	300	58269	11,32	18,59	300	-2,82
<i>a280</i>	2579	4766	65,47	84,80	300	5700	104,12	121,02	299	-16,39
<i>pr299</i>	48191	60113	12,89	24,74	300	64006	29,81	32,82	300	-6,08
<i>lin318</i>	42029	57318	16,27	36,38	300	60511	38,37	43,97	299	-5,28
<i>linhp318</i>	41345	57819	18,48	39,85	300	60568	38,41	46,49	299	-4,54
<i>rd400</i>	15281	67956	39,88	344,71	300	81836	38,31	435,54	299	-16,96
<i>fl417</i>	11861	59952	70,78	405,45	299	80710	78,43	580,47	299	-25,72
Média	--	37380,98	8,89	27,39	272	38556,45	14,19	35,98	264	-2,18

Figura 27 - Desempenho das meta-heurísticas SA_H e SA_C .Figura 28 - Qualidade das soluções das meta-heurísticas SA_H e SA_C .

6.4.1.2 Problema de Sequenciamento de Atividades

A presente seção apresenta os resultados dos estudos de sintonização da meta-heurística SA para resolução do Problema de Minimização do Atraso Total Ponderado em máquina única (TWTP). As sintonizações obtidas para a meta-heurística SA referem-se a diferentes *benchmarks* do problema supracitado (wt40, wt50 e wt100) com número variado de atividades, isto é, 40, 50 e 100 atividades.

As sintonizações da meta-heurística SA para os *benchmarks* do TWTP são apresentados nas Tabelas 16, 17 e 18, respectivamente.

Tabela 16 - wt40: Sintonização da meta-heurística SA sob diferentes métodos.

Parâmetros	HORA	Corrida
T_0	128767 ± 42206	119778 ± 36697
SA_{\max}	1391 ± 87	1316 ± 0
α	$0,946 \pm 0,001$	$0,946 \pm 0,001$
Tempo total	698	3213

Tabela 17 - wt50: Sintonização da meta-heurística SA sob diferentes métodos.

Parâmetros	HORA	Corrida
T_0	131873 ± 42287	131859 ± 40626
SA_{\max}	1220 ± 24	1202 ± 0
α	$0,946 \pm 0,001$	$0,945 \pm 0,000$
Tempo total	881	3099

Tabela 18 - wt100: Sintonização da meta-heurística SA sob diferentes métodos.

Parâmetros	HORA	Corrida
T_0	196923 ± 63446	188380 ± 61452
SA_{\max}	1304 ± 61	1231 ± 40
α	$0,951 \pm 0,002$	$0,948 \pm 0,000$
Tempo total	544	7539

A partir da comparação dos resultados (Tabelas 16, 17 e 18) pode-se observar que as sintonizações obtidas por ambos métodos (HORA e Corrida) são semelhantes para os três parâmetros (T_0 , SA_{\max} e α) identificados como significantes para o desempenho da meta-heurística SA pelos estudos experimentais. Mesmo sem construir *a priori* um conjunto de alternativas para a pesquisa, o método HORA é mais eficaz em todos os estudos realizados. Por exemplo, para o *benchmark* wt40, o processo de sintonização ocorre em aproximadamente 22% do tempo requerido pelo método de Corrida.

As mesmas características são observadas nos estudos com os *benchmarks* wt50 e wt100. Isto é, os resultados são muito semelhantes e parecem situar-se em uma mesma região do espaço de busca. Novamente, o método HORA é mais eficaz para encontrar a sintonização ideal da meta-heurística, uma vez que demanda uma fração do tempo requerido pelo método de Corrida.

Portanto, a partir destes estudos de sintonização pode-se observar que a definição do espaço de busca de parâmetros faz as sintonizações convergirem para uma região comum no espaço de busca. Esta característica também foi identificada no estudo anterior, em que a amplitude de valores no espaço de parâmetros é relativamente grande. Independente do método de sintonização e do tipo de problema escolhido observa-se que os valores encontrados para os parâmetros são semelhantes.

As sintonizações da meta-heurística SA para os diferentes *benchmarks* do TWTP (wt40, wt50 e wt100) são utilizadas na configuração de diversas versões do algoritmo. Com base nos ajustes obtidos (Tabelas 16, 17 e 18) cada versão foi executada cinco vezes sobre 125 exemplares de cada *benchmark* utilizado nos estudos de sintonização com os seguintes critérios de parada: número máximo de iterações sem alterações no valor da função objetivo (200 iterações), e tempo máximo de execução (300 segundos).

Os resultados computacionais completos de todas as versões da meta-heurística SA para os *benchmarks* do TWTP são apresentados no Apêndice B. Devido à extensão dos resultados, apresenta-se um resumo por meio de valores médios dos parâmetros na Tabela 19. As colunas desta tabela correspondem ao número de exemplares em que uma versão encontra soluções de qualidade superior em relação à outra (melhor), qualidade média das soluções (gap), robustez em relação à melhor solução encontrada solução média (desvio), melhora do método HORA em relação ao método de Corrida (delta) e tempo médio de execução em segundos (t). Nos resultados resumidos, as versões SA_{H40} e SA_{C40} são sintonizadas pelos métodos HORA e Corrida, respectivamente, para o wt40, enquanto que, SA_{H50} e SA_{C50} são as versões sintonizadas por ambos métodos para o wt50. Para o wt100, as duas versões são SA_{H100} e SA_{C100} , sintonizadas de acordo com os métodos HORA e Corrida, respectivamente.

Tabela 19 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística SA (*benchmarks* wt40, wt50 e wt100).

Versões	melhor	gap	desvio	delta	t
SA _{H40}	34/125	1,25	1,23	-0,30	68
SA _{C40}	18/125	1,62	1,13	--	40
SA _{H50}	35/125	0,81	1,58	-0,21	78
SA _{C50}	39/125	1,06	1,24	--	57
SA _{H100}	51/125	1,99	10,37	-0,91	197
SA _{C100}	46/125	3,57	9,47	--	155

A análise dos resultados revela que as versões da meta-heurística SA sintonizadas com o método HORA apresentam resultados, frequentemente, melhores em relação à versão SA_C (Tabela 19). Isto é, as versões SA_H encontram soluções melhores (coluna "melhor") para os diferentes *benchmarks* do TWTP. Entretanto, estes resultados apresentam 3% de inferioridade para o wt50.

A qualidade das soluções encontradas (gap) pelas versões da meta-heurística SA sintonizadas pelo método HORA é melhor para todos os problemas analisados. A partir dos resultados pode-se observar que as sintonizações das versões SA_C são mais robustas (desvio), uma vez que os desvios entre a melhor solução encontrada e a solução média são menores. Entretanto, deve-se ressaltar que há diminuição na robustez das soluções de todas as versões da meta-heurística SA com o aumento do número de atividades dos problemas.

A superioridade do método HORA é confirmada com a melhora na qualidade das soluções (delta) de todos os *benchmarks*, isto é, wt40, wt50 e wt100. No entanto, as versões sintonizadas pelo método de Corrida apresentam ampla vantagem de aproximadamente 20 segundos no tempo de execução em relação às versões SA_H.

Em síntese, as sintonizações da meta-heurística SA para os *benchmarks* do TWTP por meio do método HORA são competitivas. Mesmo com desvantagens nos resultados de alguns parâmetros (desvio e *t*), as versões SA_H melhoram a qualidade das soluções dos diferentes problemas. Entretanto, de acordo com o teste de hipóteses não há evidências estatísticas para considerar diferenças nos resultados computacionais observados pelas diferentes versões da meta-heurística SA. Portanto, ao nível de significância de 10% os resultados são semelhantes. Mas, ao considerar-se o tempo consumido pelo processo de sintonização, o método HORA é mais eficaz, uma vez que demanda muito menos tempo do que o método de Corrida.

6.4.2 Sintonização da Meta-heurística GA

No presente estudo serão sintonizados 4 parâmetros da meta-heurística GA por meio da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa. Os parâmetros do algoritmo foram selecionados de acordo com recomendações da literatura e suas configurações iniciais para os estudos experimentais correspondem aos valores adotados com mais frequência por estudos empíricos (Tabela 20).

Tabela 20 - Parâmetros da meta-heurística GA selecionados para sintonização.

Parâmetros	Descrições	Valores
p_c	<i>Probabilidade cruzamento</i>	[0,40; 0,90]
p_m	<i>Probabilidade de mutação</i>	[0,001; 0,025]
μ	<i>Tamanho da população</i>	[10; 100]
n	<i>Número de gerações</i>	[100; 1000]

Os estudos de sintonização apresentados nas próximas seções empregam os métodos HORA e Corrida na sintonização da meta-heurística GA. A qualidade dos resultados produzidos será comparada a partir da execução da meta-heurística GA sob diferentes sintonizações obtidas por ambos métodos para cada problema de otimização selecionado.

6.4.2.1 Problema do Caixeiro Viajante

As sintonizações da meta-heurística GA obtidas por meio dos métodos HORA e Corrida para o TSP são apresentados na Tabela 21.

Tabela 21 - TSP: Sintonização da meta-heurística GA sob diferentes métodos.

Parâmetros	HORA	Corrida
p_c	0,665 ± 0,235	0,549 ± 0,233
p_m	0,020 ± 0,005	0,020 ± 0,007
μ	81 ± 15	84 ± 15
n	760 ± 230	690 ± 228
Tempo total	896	16740

A partir dos resultados obtidos por ambos métodos (Tabela 21) pode-se observar que a sintonização da meta-heurística GA para o *benchmark* TSP encontra-se em uma mesma região do espaço de busca de parâmetros, tal que seus valores são semelhantes. Esta característica é observada, principalmente, para os parâmetros significantes para o

desempenho da meta-heurística (p_c , p_m e μ), de acordo com os estudos experimentais com DOE. Os resultados demonstram que o método HORA é mais eficaz no processo de sintonização, uma vez que demanda 5% do tempo requerido pelo método de Corrida.

A qualidade das sintonizações obtidas pelos métodos HORA e Corrida são comparadas a partir de duas versões da meta-heurística GA. Isto é, a versão GA_H é sintonizada de acordo com o método HORA, enquanto que a versão GA_C é sintonizada pelo método de Corrida (GA_C). Os algoritmos foram executados cinco vezes sobre 40 exemplares do *benchmark* TSP, considerando-se como critérios de parada o número máximo de 200 iterações sem alterações no valor da função objetivo e o tempo máximo de execução de 300 segundos.

Os resultados computacionais completos são apresentados na Tabela 22, cuja estrutura é descrita na Seção 6.4.

A partir dos resultados pode-se observar que a versão do GA sintonizada pelo método HORA encontra soluções melhores (sol) do que a versão GA_C para 20 exemplares e ambas meta-heurísticas (GA_H e GA_C) têm a mesma solução para 1 exemplar (Tabela 22).

O desempenho da meta-heurística GA_H é melhor em termos dos parâmetros: sol, desvio e gap. No entanto, a partir dos resultados pode-se observar que a GA_C apresenta melhora de 0,35% na qualidade das soluções (delta) em relação à versão sintonizada pelo método HORA.

O tempo médio de execução da versão GA_C é em média 20 segundos mais rápido ($t \cong 102$ segundos) do que a GA_H ($t \cong 132$ segundos). A divergência de tempo observada pode ser relacionada às diferenças em relação número de gerações da meta-heurística GA (parâmetro n), que na versão GA_C é menor (Tabela 21).

Tabela 22 - TSP: Resultados computacionais da meta-heurística GA.

Exemplar	melhor	GA _H				GA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
<i>eil51</i>	426	548	10,00	28,64	89	562	10,18	31,92	78	-2,49
<i>berlin52</i>	7542	10906	6,21	44,60	98	10169	11,78	34,83	79	7,25
<i>st70</i>	675	1162	6,25	72,15	109	1179	8,50	74,67	52	-1,44
<i>eil76</i>	538	883	4,24	64,13	112	893	6,83	65,99	76	-1,12
<i>pr76</i>	108159	184088	7,44	70,20	109	186107	8,73	72,07	89	-1,08
<i>rat99</i>	1211	2503	11,14	106,69	103	2503	4,25	106,69	69	0,00
<i>kroA100</i>	21282	51750	6,77	143,16	107	48097	13,14	126,00	101	7,60
<i>kroB100</i>	22141	47150	9,34	112,95	109	54552	4,46	146,38	82	-13,57
<i>kroC100</i>	20749	48653	4,32	134,48	121	45617	19,00	119,85	83	6,66
<i>kroD100</i>	21294	50275	3,92	136,10	110	47604	12,75	123,56	61	5,61
<i>kroE100</i>	22068	47340	9,51	114,52	127	50991	9,30	131,06	103	-7,16
<i>rd100</i>	7910	17160	8,32	116,94	123	16991	7,04	114,80	82	0,99
<i>eil101</i>	629	1109	10,39	76,31	102	1126	7,21	79,01	52	-1,51
<i>lin105</i>	14379	35082	8,06	143,98	114	36074	6,40	150,88	74	-2,75
<i>pr107</i>	44303	155534	8,79	251,07	118	130438	22,51	194,42	88	19,24
<i>pr124</i>	59030	173770	20,62	194,38	95	180076	6,62	205,06	61	-3,50
<i>bier127</i>	118282	235828	3,73	99,38	110	223367	3,31	88,84	60	5,58
<i>ch130</i>	6110	14808	4,04	142,36	100	14096	10,19	130,70	63	5,05
<i>pr136</i>	96772	254075	4,96	162,55	85	252143	4,27	160,55	54	0,77
<i>pr144</i>	58537	235054	5,52	301,55	98	221865	9,47	279,02	72	5,94
<i>ch150</i>	6528	16381	4,78	150,93	86	15757	11,61	141,38	64	3,96
<i>kroA150</i>	26524	72984	11,04	175,16	121	72510	8,20	173,38	67	0,65
<i>kroB150</i>	26130	70713	6,47	170,62	113	72468	3,72	177,34	70	-2,42
<i>pr152</i>	73682	294214	9,46	299,30	106	294578	4,36	299,80	88	-0,12
<i>u159</i>	42080	120137	13,00	185,50	101	123143	3,65	192,64	75	-2,44
<i>rat195</i>	2323	6524	5,42	180,84	108	6377	4,42	174,52	96	2,31
<i>d198</i>	15780	53175	8,82	236,98	133	46061	7,86	191,89	109	15,44
<i>kroA200</i>	29368	96214	2,83	227,62	125	93002	6,30	216,68	103	3,45
<i>kroB200</i>	29437	91452	5,38	210,67	136	91087	10,67	209,43	95	0,40
<i>ts225</i>	126643	437982	6,30	245,84	125	484636	2,21	282,68	103	-9,63
<i>tsp225</i>	3916	11336	10,95	189,48	126	12376	1,73	216,04	103	-8,40
<i>pr226</i>	80369	437458	7,79	444,31	138	470687	4,56	485,66	106	-7,06
<i>gil262</i>	2378	7539	5,48	217,03	159	8207	1,42	245,12	127	-8,14
<i>pr264</i>	49135	288342	9,68	486,84	156	307324	2,87	525,47	139	-6,18
<i>a280</i>	2579	9387	3,88	263,98	172	9102	6,34	252,93	144	3,13
<i>pr299</i>	48191	182468	11,84	278,64	188	196727	3,52	308,22	159	-7,25
<i>lin318</i>	42029	151323	9,37	260,04	214	158278	8,03	276,59	181	-4,39
<i>linhp318</i>	41345	161635	2,47	290,94	215	164263	3,73	297,30	173	-1,60
<i>rd400</i>	15281	58742	2,99	284,41	290	52947	11,89	246,49	259	10,94
<i>fl417</i>	11861	114765	8,35	867,58	317	113325	10,54	855,44	327	1,27
Média	--	106261,23	7,50	204,57	132	107932,63	7,59	205,13	102	0,35

As inspeções visuais das estatísticas (Figura 29) demonstram que as soluções encontradas por ambos métodos são semelhantes. No entanto, a meta-heurística GA_C tem amplitude suavemente menor do que a versão GA_H e sofre menos variabilidade em relação à

qualidade das soluções. Ambas meta-heurísticas (GA_H e GA_C) apresentam assimetrias positivas e moderadas, sendo a versão GA_C mais assimétrica.

Em relação à qualidade das soluções, pode-se identificar visualmente (Figura 30) que ambas versões (GA_H e GA_C) têm comportamentos semelhantes sobre todos os exemplares do *benchmark* TSP. Destaca-se no gráfico a degradação no desempenho (qualidade das soluções) das duas versões da meta-heurística GA a medida que aumenta o número de cidades dos exemplares.

De acordo com os resultados computacionais, a versão da meta-heurística GA sintonizada pelo método HORA apresenta resultados melhores do que a versão GA_C em termos de diferentes parâmetros (sol, gap e desvio), enquanto que a meta-heurística GA_C melhora a qualidade das soluções do TSP.

A partir do teste de hipóteses ao nível de significância de 10% confirma-se as observações destacadas anteriormente em relação à semelhança dos resultados produzidos pela meta-heurística GA sintonizada por ambos métodos (HORA e Corrida). No entanto, salienta-se novamente que o método HORA apresenta ampla vantagem em termos do tempo consumido pelo processo de sintonização.

Figura 29 - Desempenho das meta-heurísticas GA_H e GA_C .

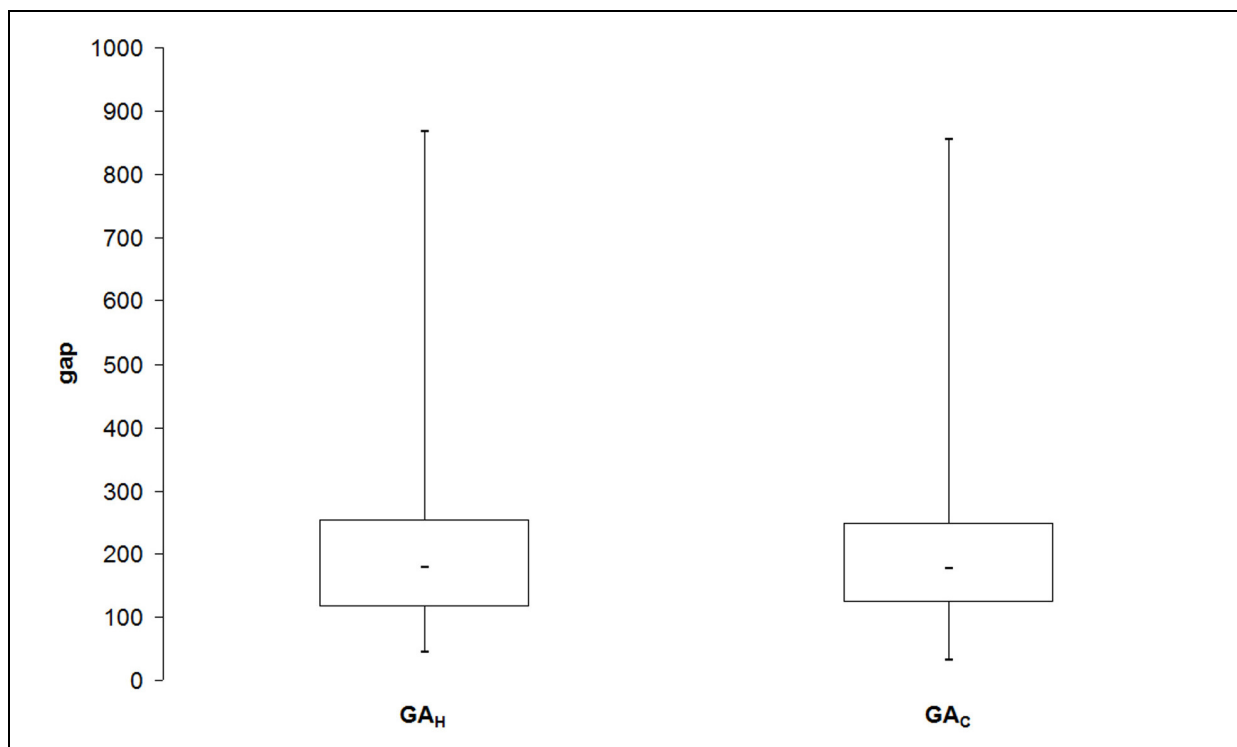
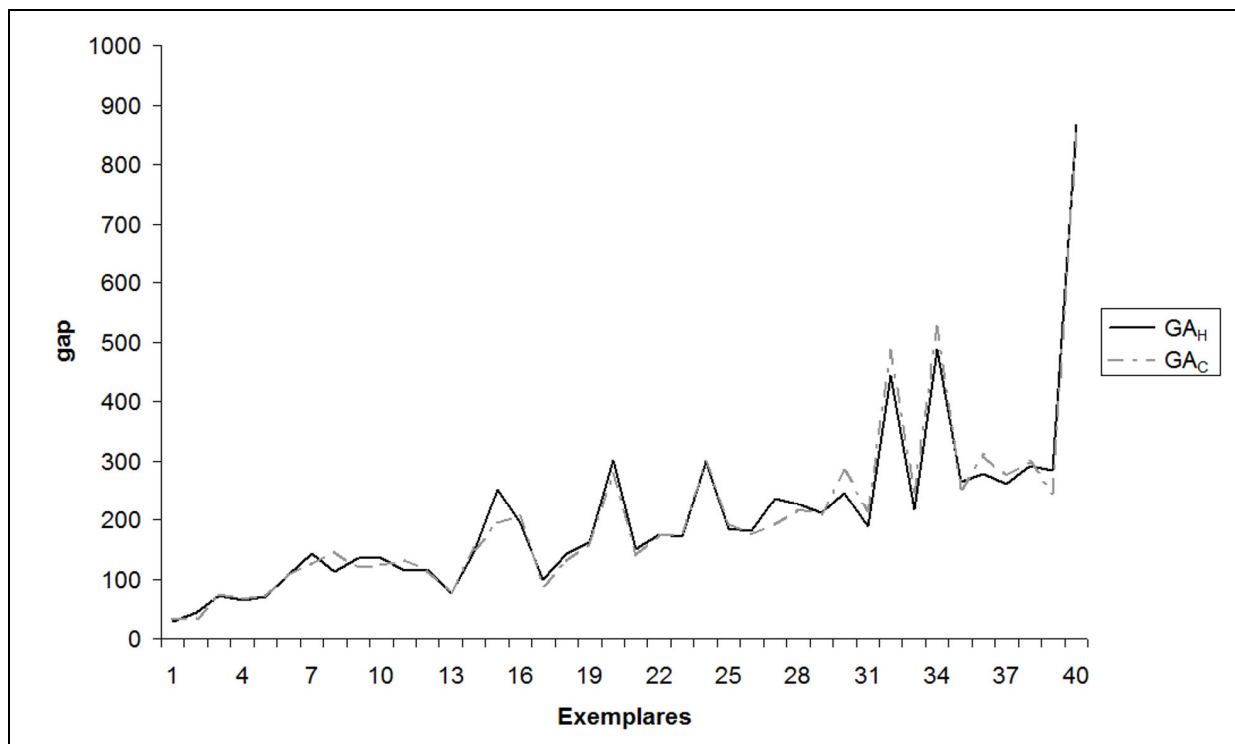


Figura 30 - Qualidade das soluções das meta-heurísticas GA_H e GA_C .

6.4.2.2 Problema de Sequenciamento de Atividades

A presente seção apresenta o estudo de sintonização dos parâmetros da meta-heurística GA (Tabela 20) para resolução do TWTP com número variado de atividades, isto é, 40, 50 e 100 atividades.

As sintonizações da meta-heurística GA para os diferentes *benchmarks* do TWTP (wt40, wt50 e wt100) são apresentados nas Tabelas 23, 24 e 25, respectivamente.

Tabela 23 - wt40: Sintonização da meta-heurística GA sob diferentes métodos.

Parâmetros	HORA	Corrida
p_c	$0,699 \pm 0,211$	$0,695 \pm 0,210$
p_m	$0,040 \pm 0,012$	$0,051 \pm 0,010$
μ	80 ± 11	90 ± 13
n	983 ± 115	1087 ± 144
Tempo total	875	11700

Tabela 24 - wt50: Sintonização da meta-heurística GA sob diferentes métodos.

Parâmetros	HORA	Corrida
p_c	$0,698 \pm 0,216$	$0,702 \pm 0,215$
p_m	$0,033 \pm 0,007$	$0,034 \pm 0,007$
μ	149 ± 18	134 ± 11
n	923 ± 118	952 ± 164
Tempo total	675	20867

Tabela 25 - wt100: Sintonização da meta-heurística GA sob diferentes métodos.

Parâmetros	HORA	Corrida
p_c	$0,687 \pm 0,205$	$0,679 \pm 0,205$
p_m	$0,024 \pm 0,002$	$0,024 \pm 0,002$
μ	127 ± 17	129 ± 23
n	736 ± 67	745 ± 66
Tempo total	829	9388

As características observadas nos resultados destes estudos (Tabelas 23, 24 e 25) são as mesmas já identificadas em estudos anteriores, inclusive considerando-se outros algoritmos e problemas. Isto é, há semelhança nos valores obtidos independente do método de sintonização (HORA e Corrida), principalmente, para os parâmetros identificados como significantes pelos estudos experimentais, isto é, p_m (probabilidade mutação), μ (tamanho da população) e n (número de gerações). De acordo com os resultados, os valores das sintonizações obtidos pelos diferentes métodos (HORA e Corrida) convergem para uma região comum dos espaço de busca de parâmetros.

Novamente, destaca-se o desempenho do método HORA em termos do tempo do processo de sintonização. A sintonização da meta-heurística GA para o wt40 ocorre em aproximadamente 7% do tempo requerido pelo método de Corrida, enquanto que para o wt50 e wt100, o tempo do método HORA é 97% e 91% mais ligeiro.

As sintonizações obtidas pelos métodos HORA e Corrida para os diferentes *benchmarks* do TWTP (Tabelas 23, 24 e 25) são utilizadas na configuração de diversas versões da meta-heurística GA. As versões da meta-heurística foram executadas cinco vezes sobre 125 exemplares de cada *benchmark* (wt40, wt50 e wt100), com os seguintes critérios de parada: número máximo de 200 iterações sem alterações no valor da função objetivo, e tempo máximo de execução de 300 segundos.

Os resultados computacionais completos de todas as versões da meta-heurística GA são apresentados no Apêndice B. Devido à extensão dos resultados, apresenta-se um resumo na Tabela 26. As colunas desta tabela correspondem ao número de exemplares em que uma

versão encontra soluções de qualidade superior em relação à outra (melhor), qualidade média das soluções (gap), robustez em relação à melhor solução encontrada solução média (desvio), melhora do método HORA em relação ao método de Corrida (delta) e tempo médio de execução em segundos (t). Nos resultados resumidos, as versões GA_{H40} e GA_{C40} correspondem as sintonizações obtidas pelos métodos HORA e Corrida, respectivamente, para o wt40, enquanto que, GA_{H50} e GA_{C50} são as versões sintonizadas por ambos métodos para o wt50. Para o wt100, as duas versões são GA_{H100} e GA_{C100} , sintonizadas de acordo com os métodos HORA e Corrida, respectivamente.

Tabela 26 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística GA (*benchmarks* wt40, wt50 e wt100).

Versões	melhor	gap	desvio	delta	t
GA_{H40}	22/125	2,66	7,73	--	89
GA_{C40}	43/125	1,74	6,69	0,69	79
GA_{H50}	38/125	1,99	17,33	-0,37	153
GA_{C50}	50/125	4,92	9,62	--	119
GA_{H100}	48/125	70,56	24,62	--	140
GA_{C100}	56/125	51,24	22,83	2,29	112

A análise dos resultados (Tabela 26) revela que as versões da meta-heurística GA sintonizadas pelo método de Corrida são frequentemente melhores em termos dos parâmetros analisados (melhor, gap, desvio, delta e t).

As versões GA_C encontram soluções melhores (coluna "melhor") para todos os problemas. No entanto, em relação à qualidade das soluções encontradas (gap) pode-se observar que a versão GA_H produz soluções mais próximas das soluções ótimas para o *benchmark* wt50.

As sintonizações obtidas pelo método de Corrida são mais robustas (desvio) e se destacam das versões GA_H para todos os *benchmarks* do TWTP. De acordo com os resultados, os maiores desvios computados por ambas versões da meta-heurística GA referem-se ao wt100.

A superioridade do método de Corrida é confirmada com a melhora na qualidade das soluções (delta) dos *benchmarks* wt40 e wt100. No entanto, a versão sintonizada pelo método de Corrida se destaca para o wt50, em que a melhora na qualidade das soluções é de 0,37%. Em relação ao parâmetro delta, a maior diferença observada se refere ao wt100, cuja piora da versão GA_H é de 2,29%.

O método de Corrida também se destaca em relação ao tempo de execução, isto é, as versões GA_C são, em geral, mais rápidas do que suas versões correspondentes sintonizadas pelo método HORA.

Em síntese, as sintonizações da meta-heurística GA pelo método de Corrida são melhores para os *benchmarks* wt40 e wt100, enquanto que a versão sintonizada pelo método HORA se destaca para o wt50. No entanto, de acordo com o teste de hipóteses, somente os resultados produzidos para o wt40 são estatisticamente significantes, sendo que o método de Corrida apresenta o melhor desempenho. As diferenças observadas para os *benchmarks* wt50 e wt100 são estatisticamente insignificantes, tal que a hipótese de nulidade H_0 , que estabelece que os desempenhos da meta-heurística GA produzidos pelas sintonizações com os métodos HORA e Corrida são semelhantes, é aceita com 90% de confiança. Portanto, a superioridade do método HORA é observada em termos do tempo consumido no processo de sintonização, que é aproximadamente 90% mais rápido em relação ao método de Corrida.

6.4.3 Sintonização da Meta-heurística VNS

Para este estudo foram selecionados 3 parâmetros da meta-heurística VNS para sintonização por meio metodologia para sintonização de meta-heurísticas proposta nesta pesquisa. Os parâmetros foram selecionados de acordo com recomendações da literatura e suas configurações iniciais para os estudos experimentais correspondem aos valores adotados com mais frequência em estudos empíricos (Tabela 27).

Tabela 27 - Parâmetros da meta-heurística VNS selecionados para sintonização.

Parâmetros	Descrições	Valores
n	Número máximo de iterações	[100; 300]
k	Número de estruturas de vizinhança	[2; 6]
δ	Distância entre as estruturas de vizinhança	[1; 5]

Os estudos de sintonização apresentados nas próximas seções empregam os métodos HORA e Corrida na sintonização da meta-heurística VNS. A qualidade dos resultados produzidos será comparada a partir da execução da meta-heurística VNS sob diferentes sintonizações obtidas por ambos métodos para cada problema de otimização estudado.

6.4.3.1 Problema do Caixeiro Viajante

Os resultados da sintonização da meta-heurística VNS por meio dos métodos HORA e Corrida para o TSP são apresentados na Tabela 28.

Tabela 28 - TSP: Sintonização da meta-heurística VNS sob diferentes métodos.

Parâmetros	HORA	Corrida
n	303 ± 129	301 ± 130
k	5 ± 1	4 ± 1
δ	3 ± 0	2 ± 0
Tempo total	64	737

A comparação dos resultados obtidos pelos métodos Corrida (Tabela 28) pode-se observar que a sintonização da meta-heurística VNS para o *benchmark* TSP são semelhantes. No entanto, de acordo com os estudos experimentais, somente o parâmetro k (número de estruturas de vizinhança) é identificado como significativo para o desempenho da meta-heurística VNS.

Destaca-se nestes resultados que os valores de sintonização para o parâmetro n (número máximo de iterações) extrapolam os limites delimitados inicialmente para os estudos experimentais (Tabela 27). Esta extrapolação está associada aos estudos experimentais com a Metodologia da Superfície de Respostas, que podem superar os limites previamente definidos para garantir rotatividade aos planejamentos e estimativas adequadas para os termos quadráticos do modelo de regressão polinomial.

De acordo com os resultados (Tabela 28), o processo de sintonização pelo método HORA é mais eficaz e consome aproximadamente 10% do tempo requerido pelo método de Corrida.

A partir dos resultados do processo de sintonização foram preparadas duas versões da meta-heurística VNS. Isto é, uma delas sintonizada pelo método HORA (VNS_H) e outra, pelo método de Corrida (VNS_C). Cada versão do algoritmo foi executada cinco vezes sobre 40 exemplares do *benchmark* TSP, considerando-se os seguintes critérios de parada: número máximo de iterações sem alterações no valor da função objetivo (200 iterações) e tempo máximo de execução (300 segundos).

Os resultados computacionais completos são apresentados na Tabela 29, cuja estrutura é descrita na Seção 6.4.

Tabela 29 - TSP: Resultados computacionais da meta-heurística VNS.

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
<i>eil51</i>	426	459	9,15	7,75	127	459	7,93	7,75	72	0,00
<i>berlin52</i>	7542	8585	5,54	13,83	144	8127	8,36	7,76	97	5,64
<i>st70</i>	675	841	4,66	24,59	272	861	7,22	27,56	180	-2,32
<i>eil76</i>	538	676	1,48	25,65	280	649	5,79	20,63	299	4,16
<i>pr76</i>	108159	137762	6,15	27,37	292	130383	6,07	20,55	299	5,66
<i>rat99</i>	1211	1653	6,92	36,50	300	1753	3,32	44,76	299	-5,70
<i>kroA100</i>	21282	30148	10,13	41,66	300	32400	8,26	52,24	299	-6,95
<i>kroB100</i>	22141	31178	13,07	40,82	300	32786	9,30	48,08	300	-4,90
<i>kroC100</i>	20749	33723	8,70	62,53	300	35441	6,15	70,81	300	-4,85
<i>kroD100</i>	21294	33732	4,22	58,41	300	30145	10,56	41,57	299	11,90
<i>kroE100</i>	22068	29579	17,46	34,04	300	34507	3,12	56,37	299	-14,28
<i>rd100</i>	7910	10356	13,70	30,92	300	11556	7,97	46,09	299	-10,38
<i>eil101</i>	629	826	4,00	31,32	300	776	11,26	23,37	299	6,44
<i>lin105</i>	14379	20428	12,70	42,07	300	23573	6,51	63,94	300	-13,34
<i>pr107</i>	44303	91302	20,27	106,09	300	100943	8,61	127,85	299	-9,55
<i>pr124</i>	59030	114385	13,77	93,77	301	113994	3,48	93,11	299	0,34
<i>bier127</i>	118282	155723	4,40	31,65	301	157307	4,75	32,99	299	-1,01
<i>ch130</i>	6110	10220	7,48	67,27	301	9674	7,90	58,33	300	5,64
<i>pr136</i>	96772	163172	8,90	68,61	301	161247	7,70	66,63	299	1,19
<i>pr144</i>	58537	142965	12,12	144,23	302	140593	11,37	140,18	299	1,69
<i>ch150</i>	6528	10772	7,07	65,01	302	11135	7,57	70,57	300	-3,26
<i>kroA150</i>	26524	48994	6,33	84,72	302	46776	12,13	76,35	300	4,74
<i>kroB150</i>	26130	51029	6,65	95,29	303	49497	5,96	89,43	303	3,10
<i>pr152</i>	73682	167636	20,92	127,51	301	181816	6,87	146,76	301	-7,80
<i>u159</i>	42080	83283	7,99	97,92	302	79471	7,88	88,86	299	4,80
<i>rat195</i>	2323	4555	4,71	96,08	302	4414	4,80	90,01	302	3,19
<i>d198</i>	15780	30545	20,83	93,57	304	35256	4,91	123,42	299	-13,36
<i>kroA200</i>	29368	61340	11,08	108,87	308	62395	3,94	112,46	301	-1,69
<i>kroB200</i>	29437	56585	14,84	92,22	307	57126	12,24	94,06	303	-0,95
<i>ts225</i>	126643	300014	6,53	136,90	314	256487	15,88	102,53	299	16,97
<i>tsp225</i>	3916	8007	5,58	104,47	313	7601	9,44	94,10	306	5,34
<i>pr226</i>	80369	278183	16,20	246,13	312	317625	8,24	295,21	300	-12,42
<i>gil262</i>	2378	5550	3,81	133,39	316	5145	7,70	116,36	312	7,87
<i>pr264</i>	49135	213082	7,31	333,67	303	190111	24,61	286,92	308	12,08
<i>a280</i>	2579	6374	7,79	147,15	328	6866	5,09	166,23	318	-7,17
<i>pr299</i>	48191	135806	4,54	181,81	334	137572	5,17	185,47	303	-1,28
<i>lin318</i>	42029	112547	5,62	167,78	353	109379	9,93	160,25	303	2,90
<i>linhp318</i>	41345	118495	8,27	186,60	342	117384	6,52	183,91	301	0,95
<i>rd400</i>	15281	45790	5,37	199,65	347	44446	4,88	190,86	351	3,02
<i>fl417</i>	11861	107419	2,90	805,65	402	108342	5,92	813,43	314	-0,85
Média	--	71592,98	8,98	112,34	300	71400,45	7,88	113,44	289	-0,36

A partir dos resultados computacionais pode-se observar que a meta-heurística VNS_C encontra soluções melhores (sol) do que a versão sintonizada pelo método HORA para 20

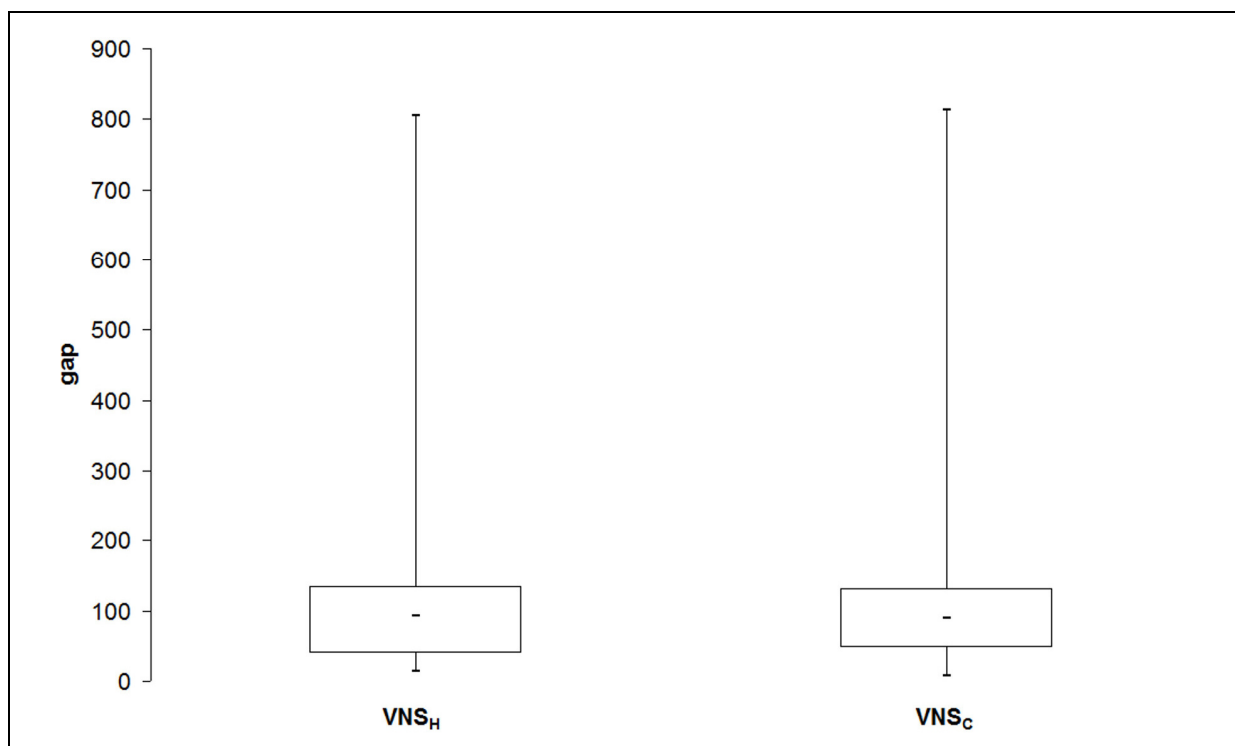
exemplares e ambas meta-heurísticas (VNS_H e VNS_C) têm a mesma solução para 1 exemplar (Tabela 29).

A meta-heurística VNS_H é melhor em relação à qualidade das soluções (gap), porém as soluções produzidas pela versão VNS_C são suavemente mais robustas (desvio). Entretanto, pode-se observar que sintonização obtida pelo método HORA promove melhora 0,36% na qualidade das soluções (delta) em relação à versão sintonizada pelo método de Corrida.

O tempo de execução da meta-heurística VNS_C ($t \cong 289$) é em média 11 segundos mais rápido do que a VNS_H ($t \cong 300$). Este déficit de desempenho pode estar relacionado as diferenças na sintonização dos parâmetros, uma vez que na versão VNS_H adota-se um número maior de iterações (parâmetro n) e de estruturas de vizinhanças (parâmetro k), que podem justificar o consumo maior no tempo de execução.

A partir dos *boxplots* (Figura 31) pode-se observar que as soluções encontradas por ambos métodos são semelhantes. No entanto, a meta-heurística VNS_C tem amplitude menor do que a versão VNS_H , ao passo que esta última apresenta maior variabilidade em relação à qualidade das soluções. Ambas meta-heurísticas (VNS_H e VNS_C) apresentam assimetrias positivas e moderadas, sendo a versão VNS_C mais assimétrica.

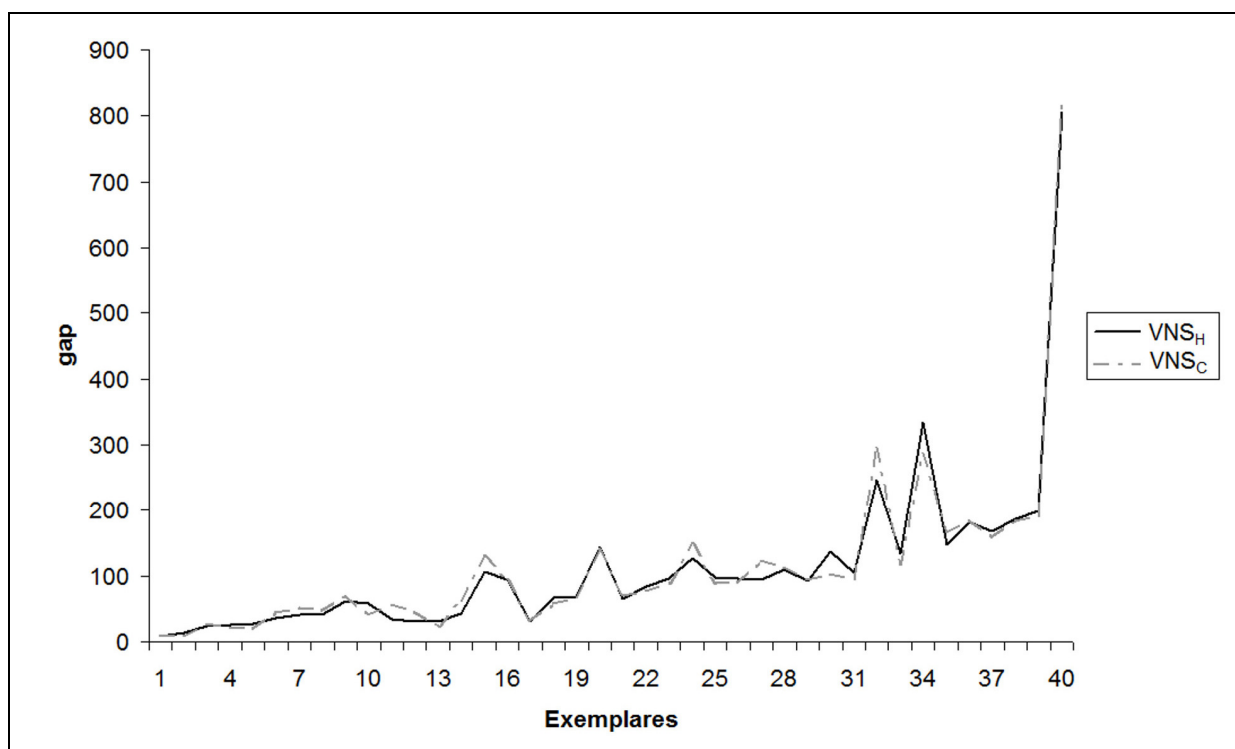
Figura 31 - Desempenho das meta-heurísticas VNS_H e VNS_C .



A qualidade das soluções de ambas meta-heurísticas (VNS_H e VNS_C) é visualmente semelhante sobre os exemplares do *benchmark* TSP (Figura 32). Destaca-se no gráfico que as soluções encontradas são de qualidade razoável e a degradação no desempenho (qualidade das soluções) de ambas versões (VNS_H e VNS_C) com o aumento do número de cidades dos exemplares.

De acordo com os resultados computacionais, a meta-heurística VNS_H é suavemente melhor (gap e delta) do que a versão sintonizada pelo método de Corrida. Entretanto, o teste de hipóteses revela com confiança de 90% que os resultados produzidos por ambas meta-heurísticas (VNS_H e VNS_C) são estatisticamente semelhantes. Novamente, o método HORA apresenta ampla vantagem em termos do tempo do processo de sintonização, aproximadamente, 9 vezes mais rápido que o tempo do método de Corrida.

Figura 32 - Qualidade das soluções das meta-heurísticas VNS_H e VNS_C .



6.4.3.2 Problema de Sequenciamento de Atividades

A presente seção apresenta os resultados dos estudos de sintonização da meta-heurística VNS para diferentes *benchmarks* do TWTP com 40, 50 e 100 atividades.

As sintonizações da meta-heurística VNS para os *benchmarks* wt40, wt50 e wt100 do TWTP são apresentados nas Tabelas 30, 31 e 32, respectivamente.

Tabela 30 - wt40: Sintonização da meta-heurística VNS sob diferentes métodos.

Parâmetro	HORA	Corrida
n	183 ± 56	197 ± 59
k	5 ± 1	5 ± 1
δ	3 ± 0	4 ± 1
Tempo total	558	2946

Tabela 31 - wt50: Sintonização da meta-heurística VNS sob diferentes métodos.

Parâmetro	HORA	Corrida
n	264 ± 90	247 ± 90
k	5 ± 1	4 ± 1
δ	3 ± 1	3 ± 1
Tempo total	1143	6021

Tabela 32 - wt100: Sintonização da meta-heurística VNS sob diferentes métodos.

Parâmetro	HORA	Corrida
n	234 ± 79	239 ± 82
k	3 ± 1	4 ± 1
δ	5 ± 0	6 ± 0
Tempo total	905	4817

A partir dos resultados (Tabelas 30, 31 e 32) pode-se observar semelhança nos valores obtidos por ambos métodos (HORA e Corrida). Esta característica é mais evidente para os parâmetros identificados como significantes nos estudos experimentais, por exemplo, k (número de estruturas de vizinhança) e δ (distância entre as estruturas de vizinhança). As sintonizações obtidas pelo método HORA para os *benchmarks* wt40 e wt50 são semelhantes, mas esta característica não é observada nas sintonizações do método de Corrida. Assim como observado anteriormente, o método HORA é mais eficaz em termos do tempo do processo de sintonização para os diferentes problemas selecionados.

As sintonizações da meta-heurística VNS para os *benchmarks* wt40, wt50 e wt100 são utilizadas na configuração de diversas versões do algoritmo. Com base nos ajustes obtidos (Tabelas 30, 31 e 32) cada versão foi executada cinco vezes sobre 125 exemplares de cada *benchmark*, com os seguintes critérios de parada: número máximo de iterações sem alterações no valor da função objetivo (200 iterações), e tempo máximo de execução (300 segundos).

Os resultados computacionais completos de todas as versões da meta-heurística VNS são apresentados no Apêndice B. Devido à extensão dos resultados, apresenta-se um resumo na Tabela 33. As colunas desta tabela correspondem ao número de exemplares em que uma

versão encontra soluções de qualidade superior a outra (melhor), qualidade média das soluções (gap), robustez em relação à melhor solução encontrada solução média (desvio), melhora do método HORA em relação ao método de Corrida (delta) e tempo médio de execução em segundos (t). Nos resultados resumidos, as versões VNS_{H40} e VNS_{C40} correspondem as sintonizações obtidas pelos métodos HORA e Corrida, respectivamente, para o *benchmark* wt40, enquanto que, VNS_{H50} e VNS_{C50} são as versões sintonizadas por ambos métodos para o wt50. Para o wt100, as duas versões são VNS_{H100} e VNS_{C100} , sintonizadas de acordo com os métodos HORA e Corrida, respectivamente.

Tabela 33 - TWTP: Resultados computacionais resumidos de diferentes versões da meta-heurística VNS (*benchmarks* wt40, wt50 e wt100).

Versões	melhor	gap	desvio	delta	t
VNS_{H40}	17/125	0,65	1,17	--	16
VNS_{C40}	19/125	0,54	1,81	0,12	10
VNS_{H50}	21/125	1,50	0,71	--	44
VNS_{C50}	24/125	1,32	0,74	0,18	25
VNS_{H100}	47/125	2,16	2,51	--	217
VNS_{C100}	28/125	2,11	2,52	0,05	193

A análise dos resultados (Tabela 33) revela que as meta-heurísticas VNS sintonizadas pelo método de Corrida se destacam suavemente em relação as versões sintonizadas pelo método de Corrida.

As versões VNS_C encontram soluções melhores (coluna "melhor") para os *benchmarks* wt40 e wt50, enquanto que a VNS_H é melhor para o wt100. No entanto, em relação à qualidade das soluções encontradas (gap) pode-se observar que as meta-heurísticas VNS_C produzem resultados mais próximos das soluções para os problemas selecionados.

Ao analisar a robustez das soluções encontradas (desvio) pode-se observar que as meta-heurísticas VNS_H apresentam os menores resultados em todos os *benchmarks*. As sintonizações do método de Corrida (VNS_C) produzem melhora na qualidade das soluções (delta) em relação ao método HORA. No entanto, esta melhora é menor do que 0,20%, no pior caso.

O método de Corrida se destaca em relação ao tempo de execução, isto é, as versões VNS_C são, em geral, mais rápidas do que as versões sintonizadas pelo método HORA. A partir dos resultados pode-se observar que a diferença cresce a medida que o número de atividades dos problemas aumenta.

Em síntese, as soluções produzidas pelas sintonizações da meta-heurística VNS pelo método de Corrida se destacam para os *benchmarks* wt40 e wt50, enquanto que a versão sintonizada pelo método HORA se destaca para o wt100. No entanto, de acordo com o teste de hipóteses, somente os resultados produzidos para o wt100 são estatisticamente significantes, sendo que o método HORA apresenta o melhor desempenho. Portanto, há evidências estatísticas ao nível de significância de 10% para afirmar que os resultados das versões VNS_C e VNS_H são semelhantes para os *benchmarks* wt40 e wt50. Entretanto, o método HORA se destaca em termos de tempo do processo de sintonização.

6.4.4 Análise dos Estudos Computacionais

A partir dos resultados computacionais apresentados na Seção 6.4.3 pode-se observar que o método HORA é competitivo em relação ao método de Corrida. No entanto, o primeiro é mais eficaz em termos de tempo do processo de sintonização.

Ao comparar as soluções produzidas por ambos métodos (HORA e Corrida) para os *benchmarks* de problemas de otimização estudados, observa-se que as versões das diferentes meta-heurísticas sintonizadas pelo método HORA se destacam em relação ao método de Corrida (Tabela 34). Nesta comparação considera-se o valor médio da função objetivo e sua classificação correspondente.

As classificações foram elaboradas a partir de resultados dos testes de hipóteses, tal que para as soluções estatisticamente significantes são atribuídos os valores 1 ou 2, para a menor ou maior solução, respectivamente. Caso contrário, para soluções estatisticamente insignificantes, isto é, aquelas consideradas semelhantes diante dos testes de hipóteses, é atribuído o valor 1,5.

Na Tabela 34, os valores em destaque (negrito) correspondem aos exemplares nos quais o método HORA encontra soluções melhores (estatisticamente significativas) em relação ao método de Corrida em termos do valor médio da função objetivo. Os valores entre parêntesis são as classificações correspondentes, descritas anteriormente.

Tabela 34 - Qualidade das soluções produzidas por diferentes meta-heurísticas sintonizadas pelos métodos HORA e Corrida.

Meta-heurísticas	TSP	wt40	wt50	wt100	Média
SA_H	37381 (1)	37673 (1,5)	52947 (1,5)	218754 (1)	1,25
SA_C	38556 (2)	37684 (1,5)	52940 (1,5)	218898 (2)	1,75
GA_H	106261 (1,5)	37698 (2)	52998 (1,5)	221358 (1,5)	1,63
GA_C	107933 (1,5)	37682 (1)	52996 (1,5)	221423 (1,5)	1,38
VNS_H	71593 (1,5)	37691 (1,5)	53013 (1,5)	218497 (1)	1,38
VNS_C	71400 (1,5)	37683 (1,5)	53015 (1,5)	218562 (2)	1,63

A partir dos resultados (Tabela 34) pode-se distinguir o desempenho das meta-heurísticas em relação ao método de sintonização (HORA e Corrida). Ao analisar o valor médio da função objetivo pode-se observar que as meta-heurísticas SA e VNS apresentam melhores resultados quando sintonizadas pelo método HORA, enquanto que a sintonização pelo método de Corrida é melhor para o GA. As análises nos valores médios das classificações indicam que as sintonizações obtidas pelo método HORA apresentam melhora no desempenho das meta-heurísticas SA e VNS.

7. ESTUDO COMPLEMENTAR

7.1 INTRODUÇÃO

O presente capítulo apresenta um estudo computacional complementar para avaliação da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa. Neste estudo considera-se a meta-heurística híbrida Busca por Agrupamento (em Inglês, *Clustering Search* - CS) um algoritmo já bem configurado para solução do problema do Caixeiro Viajante com Lucro. O desempenho do algoritmo CS é avaliado antes e após a aplicação da metodologia de sintonização proposta.

Este capítulo descreve o funcionamento básico do algoritmo CS, cuja ideia é identificar áreas promissoras do espaço de busca através de soluções obtidas com uma meta-heurística e agrupa-las para exploração com heurísticas de busca local. O capítulo apresenta, também, uma breve introdução sobre o Problema do Caixeiro Viajante com Lucro, uma generalização do TSP em que não é necessário visitar todos os vértices e cada vértice está associado a um prêmio. Este capítulo é finalizado com a apresentação e análise dos resultados experimentais obtidos na resolução de diferentes exemplares do TSP com Lucro com a meta-heurística híbrida CS.

7.1.1 Busca por Agrupamento

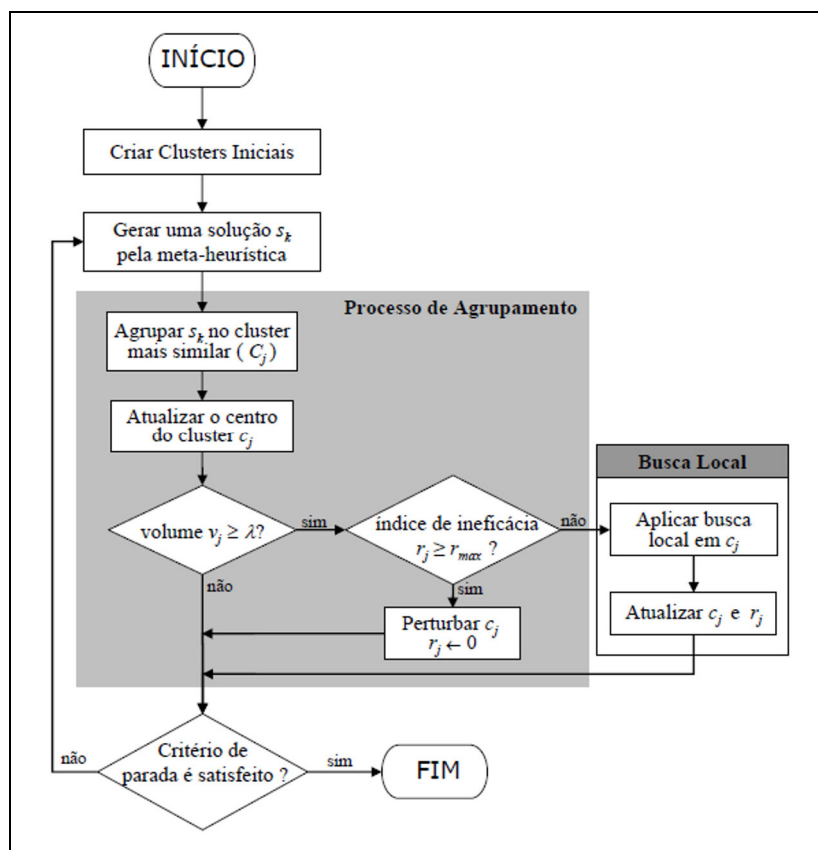
O algoritmo Busca por Agrupamento é uma generalização da Pesquisa Evolucionária por Agrupamento (em Inglês, *Evolutionary Clustering Search* - ECS), proposta por Oliveira e Lorena (2004 e 2007), que emprega agrupamento para detectar áreas promissoras do espaço de busca. Tais áreas devem ser descobertas rapidamente a fim de se aplicar uma estratégia de pesquisa sobre elas. No ECS, um processo de agrupamento é executado simultaneamente a um algoritmo evolucionário, com a finalidade de identificar indivíduos que merecem atenção especial.

No CS (CHAVES, 2009), o algoritmo evolucionário é substituído por meta-heurísticas distintas, tais como SA, ILS, GRASP, Busca Tabu e outros. O CS procura dividir o espaço de busca e localizar as regiões promissoras por meio do enquadramento dessas regiões em grupos (em Inglês, *clusters*). Um grupo pode ser definido por três atributos $C = (c; q; r)$. O centro c é uma solução que representa o agrupamento C , identificando a sua localização no

espaço de busca. Ao invés de armazenar todas as soluções no agrupamento, apenas parte das informações são inseridas no centro do agrupamento. O volume q é a quantidade de soluções agrupadas em C . Um grupo se torna promissor quando o volume atinge certo limite λ . O índice de ineficácia r é uma variável de controle para identificar se a busca local está melhorando o centro do agrupamento C . O valor de r indica o número de vezes consecutivas que a busca local é aplicada sobre C sem melhoria da solução. Este atributo evita que a busca local seja executada mais de r_{max} vezes em regiões ruins ou regiões que já tenham sido suficientemente exploradas.

O CS é um método iterativo que possui três componentes principais: uma meta-heurística, um processo de agrupamento e um método de busca local (Figura 33).

Figura 33 - Fluxograma do método CS.



FONTE: (CHAVES, 2009)

A meta-heurística funciona como um gerador de solução em tempo integral. O algoritmo é executado de forma independente e fornece um conjunto de soluções para o processo de agrupamento. A cada iteração, uma solução s_k é gerada e enviada para o processo de agrupamento. Essa solução é anexada ao agrupamento C_j mais similar, isto é, aquele mais próximo da solução s_k . No CS, a proximidade entre as soluções é medida a partir de uma

métrica de distância, por exemplo, a distância Hamming (HAMMING, 1950). Cada nova solução agrupada demanda atualização no centro de C_j , tal que esse centro se desloca no espaço de busca, e uma análise do volume q_j do agrupamento. Caso o volume tenha atingido o limite λ definido *a priori*, esse grupo pode estar em uma região promissora. O processo de busca local é executado a cada vez que se encontra uma região promissora e permite intensificar a exploração nesta região. Se a busca local não obtém sucesso em r_{\max} aplicações, então, o centro de C_j é perturbado para escapar desta região. Por outro lado, se r_j é menor que r_{\max} , a busca local é aplicada para intensificar a busca na vizinhança de C_j . A busca local obtém sucesso quando encontra uma solução melhor (c_j^*) neste agrupamento. Ao final do processo retorna-se para a meta-heurística que irá gerar outras soluções (CHAVES, 2009; CHAVES e LORENA, 2007).

O critério de parada do CS pode ser definido pelo número máximo de iterações do processo de agrupamento ou o critério de parada da meta-heurística escolhida.

7.1.2 Problema do Caixeiro Viajante com Lucro

O TSP está entre os problemas de otimização combinatória mais estudados da literatura. Dentre suas extensões, um caso especial ocorre quando as cidades são selecionadas de acordo com um prêmio, que é atribuído quando acontece uma visita. Esta característica dá origem a uma série de problemas reunidos sob o nome de Problema do Caixeiro Viajante com Lucro (em Inglês, *Traveling Salesman Problem with Profit* - TSPP) (FEILLET et al., 2005).

O TSPP pode ser visto como um TSP biobjetivo, em que os objetivos são opostos. Isto é, um deles consiste em percorrer todas as rotas a fim de coletar os prêmios, enquanto que no outro, os custos das visitas devem ser minimizados, tal que algumas rotas podem deixar de ser percorridas. Visto assim, resolver o TSPP deve resultar em encontrar uma fronteira de Pareto, ou seja, um conjunto de soluções viáveis tal que nenhum objetivo possa ser melhorado sem deteriorar o outro. Na prática, a maioria das pesquisas trata de problemas com um objetivo único, de modo que os dois objetivos são ponderados e combinados linearmente, ou um dos objetivos é transformado em restrição com um valor limite especificado.

Intuitivamente, pode-se considerar que uma solução do TSP está associada a uma solução do TSPP, em que o objetivo é coletar todos os prêmios. Os problemas TSPP pertencem à classe NP-difícil (FEILLET et al., 2005).

O TSP pode ser definido na forma de um grafo completo $G = (V, A)$, em que V é um conjunto de n vértices e A é um conjunto de arcos, no caso de grafos direcionados, ou de arestas, no caso grafos não direcionados. Para cada vértice $v_i \in V$ há um prêmio associado ($p_{v_i} \geq 0$) e associado a cada arco ou aresta $(i, j) \in A$ existe uma distância d_{ij} . O vértice v_1 representa a origem e, por definição seu prêmio é nulo, isto é, $p_{v_1} = 0$. O objetivo é determinar um percurso hamiltoniano que contenha a origem v_1 e satisfaça a maximização dos prêmios coletados e a minimização dos custos de deslocamentos no percurso.

Os problemas que compõem o TSP podem ser classificados em:

- TSP com Rotas Lucrativas (DELL'AMICO et al., 1995), onde os objetivos são combinados visando encontrar um percurso para minimizar os custos de deslocamento subtraindo-se os prêmios coletados;
- TSP Orientado (TSILIGIRIDES, 1984; GOLDEN et al., 1987), em que o objetivo é encontrar um percurso para maximizar o prêmio, tal que os custos associados ao percurso não excedam um valor máximo; e
- TSP com Coleta de Prêmios (BALAS, 1989), cujo objetivo é encontrar um percurso para minimizar os custos, de modo que o valor total dos prêmios coletados não seja menor do que um valor pré-definido.

Em sua versão mais geral, o TSP com Coleta de Prêmios (em Inglês, *Prize-collecting Traveling Salesman Problem* - PCTSP) consiste em coletar certa quantidade de prêmios pelos vértices visitados durante o percurso, sujeito à aplicação de penalidades pelos vértices não visitados. Isto é, no PCTSP é preciso minimizar o somatório dos custos do percurso somado às penalidades pagas nos vértices não visitados e incluir no percurso um número suficiente de vértices que permita coletar um prêmio mínimo (p_{\min}) pré-estabelecido. Na literatura é possível encontrar trabalhos que não atribuem penalidades aos vértices não visitados, sendo esses problemas conhecidos como TSP com Quota (AWERBUCH et al., 1998).

A formulação do modelo de otimização do PCTSP é a seguinte:

$$\text{minimizar } \sum_{v_i \in V} \sum_{v_j \in V \setminus \{v_i\}} d_{ij} v_{ij} - \sum_{c_i \in C} \gamma_i (1 - y_i), \quad (38)$$

Subjeito à:

$$\sum_{v_j \in V} v_{ij} = y_i, \quad \forall v_i \in V, \quad (39)$$

$$\sum_{v_j \in V} v_{ij} = y_j, \forall v_j \in V, \quad (40)$$

$$\text{Eliminação de sub-rotas} \quad (41)$$

$$\sum_{v_j \in V} p_i y_i \geq p_{\min}, \quad (42)$$

$$y_{v_1} = 1, \quad (43)$$

$$v_{ij} \in \{0, 1\}, \forall i, j \in A, \quad (44)$$

$$y_i \in \{0, 1\}, \forall v_i \in V, \quad (45)$$

em que d_{ij} corresponde à distância ou custo do percurso entre as cidades i e j , c_{ij} é uma variável de decisão que assume valor 1 se a aresta (i, j) aparece no percurso, isto é, se existe um caminho entre as cidades i e j , e valor 0 (zero) caso contrário, e y_i é uma variável de decisão que assume valor 1 se o vértice v_i está presente na solução, e valor 0 (zero) caso contrário. As restrições 39 e 40 são chamadas restrições de atribuição e garantem que cada vértice seja visitado no máximo uma vez. A restrição 42 é conhecida como restrição de cobertura generalizada, e assegura que a soma dos prêmios coletados seja maior ou igual que o prêmio mínimo (p_{\min}) definido *a priori*. A restrição 43 assegura que a origem seja visitada. As restrições 44 e 45 asseguram que as variáveis c_{ij} e y_i são binárias.

As restrições 41 podem ser tratadas de diferentes maneiras. Neste trabalho optou-se por utilizar o mesmo conjunto de restrições adotado em Chaves (2009), propostas em (TORRES e BRITO, 2003) para proibir a formação de sub-rotas. Nesse conjunto de restrições (46-51), a variável f_{ij} ($\forall i, j \in A, i \neq j$) representa a quantidade de fluxo (valor em prêmios) escoado em uma aresta (i, j) :

$$\sum_{v_j \in V \setminus \{v_1\}} f_{1j} = 0, \quad (46)$$

$$\sum_{v_j \in V \setminus \{v_i\}} f_{ij} = \sum_{v_j \in V \setminus \{v_i\}} f_{ji} + p_i y_i, \forall v_i \in V \setminus \{v_1\}, \quad (47)$$

$$\sum_{v_j \in V \setminus \{v_i\}} f_{j1} = \sum_{v_j \in V \setminus \{v_i\}} p_j y_j, \quad (48)$$

$$f_{ij} > v_{ij} - 1, \forall v_i \in V \setminus \{v_1\}, \forall v_j \in V, \quad (49)$$

$$\left(\sum_{v_j \in V} p_j \right) v_{ij} \geq f_{ij}, \forall v_i \in V \setminus \{v_1\}, \forall v_j \in V, \quad (50)$$

$$f_{ij} \geq 0, \forall i, j \in A. \quad (51)$$

As restrições 46, 47 e 48 garantem a conectividade da solução, ou seja, evitam rotas desconexas da origem. As variáveis de fluxo f_{ij} impedem que sub-rotas sejam criadas, associando a quantidade de prêmio do vértice visitado até a aresta que sai deste. A quantidade de fluxo que sai do vértice origem (v_1) tem que ser igual a 0, e a quantidade de fluxo que volta para v_1 tem que ser igual à soma dos prêmios coletados na rota. As restrições 49 e 50 conectam as variáveis c_{ij} e f_{ij} , fazendo com que a rota gerada por ambas seja a mesma. As restrições 49 garantem que se uma aresta fizer parte da solução, a quantidade de fluxo escoada por esta tem que ser maior que 0, e as restrições 50 asseguram que o valor do fluxo escoado por uma aresta não será maior que o total de prêmios de todos os vértices da rota. Por fim, as restrições 51 garantem que as variáveis f_{ij} sejam não negativas.

7.2 ESTUDOS COMPUTACIONAIS

Os estudos computacionais da presente pesquisa foram realizados em ambiente de computação científica gratuito Scilab (URL: <http://www.scilab.org>) em um computador pessoal com processador Intel Core i7 3.6GHz, memória RAM de 8GB e sistema operacional Windows 7 64 bits.

Estes estudos foram realizados de maneira complementar para avaliar a qualidade da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa, quando aplicada em algoritmos já bem configurados. Os algoritmos aqui considerados correspondem a duas versões do CS desenvolvidas em Chaves (2009), que utilizam as meta-heurísticas SA e GA, respectivamente, para gerar soluções para o processo de agrupamento¹⁴. Os parâmetros selecionados para os estudos de sintonização, bem como suas descrições e valores adotados na literatura (CHAVES, 2009) são apresentados nas Tabelas 35 e 36.

¹⁴ Nesta pesquisa foram sintonizados os parâmetros das meta-heurísticas SA e GA que compõem o CS, mantendo-se inalterados os demais parâmetros do algoritmo relacionados ao processo de agrupamento (número de agrupamentos, limite λ e outros).

Tabela 35 - Parâmetros da meta-heurística híbrida CS-SA adotados na literatura.

Parâmetros	Descrições	Valores
T_0	Valor da temperatura inicial	1000000
SA_{max}	Número de iterações a cada estágio da temperatura	1000
α	Taxa de resfriamento da temperatura	0,95

Fonte: (CHAVES, 2009).

Tabela 36 - Parâmetros da meta-heurística híbrida CS-GA adotados na literatura.

Parâmetros	Descrições	Valores
p_c	Probabilidade cruzamento	0,95
p_m	Probabilidade de mutação	0,001
μ	Tamanho da população	80
N	Número de gerações	1000

Fonte: (CHAVES, 2009).

O PCTSP não possui uma biblioteca pública de exemplares. Portanto, os exemplares utilizados nesta pesquisa são adaptações de 14 exemplares clássicos do *benchmark* TSP obtidos na TSPLIB (REINELT, 1991) em PCTSP correspondentes. Os prêmios e as penalidades foram gerados ao acaso com a distribuição de probabilidades uniforme. Os prêmios variam entre 1 e 100, as penalidades (γ) entre 1 e W , sendo $W = \{100; 1000; 10000\}$, e o valor do prêmio mínimo foi definido como $\sigma = \{0,2; 0,5; 0,8\}$. Detalhes sobre o processo de transformação dos exemplares podem ser obtidos em Chaves (2009).

As melhores soluções conhecidas para os exemplares de pequeno porte (*burma14*, *ullyses22*, *berlin52*, *st70*, *pr76*, *kroA100*, *kroB100*, *bier127* e *pr152*) foram calculadas pelo software CPLEX versão 1.11.1 (ILOG, 2007) na pesquisa de Chaves (2009). Devido à complexidade do problema, as soluções para os demais exemplares (*gr202*, *tsp225*, *a280*, *lin318* e *gr431*) foram obtidas por Chaves (2009) após vinte execuções do CS.

O processo de sintonização das meta-heurísticas híbridas CS-SA e CS-GA inicia-se a partir de estudos experimentais sobre um conjunto de treinamento formado por 4 exemplares do PCTSP (por exemplo, *berlin52*, *st70*, *bier127* e *pr152*), selecionados arbitrariamente do *benchmark* PCTSP gentilmente cedido pelo Prof. Antonio Augusto Chaves a partir da URL: <http://www.sjc.unifesp.br/docente/chaves>. Todos exemplares que formam o conjunto de treinamento são simétricos, com número variado de cidades e distância Euclidiana entre as cidades.

Os parâmetros são estudados por meio de planejamentos fatoriais e da Metodologia da Superfície de Respostas. Os estudos experimentais foram repetidos para cada exemplar que

forma o conjunto de treinamento, tal que o resultado final são 4 alternativas distintas de configurações, cada uma relacionada a um exemplar. A partir das configurações identificadas define-se um espaço de busca de parâmetros, limitado pelos valores máximo e mínimo de cada parâmetro identificados a partir de estudos experimentais.

A sequência da metodologia para sintonização de meta-heurísticas consiste em aplicar o método HORA para ajuste fino dos parâmetros das meta-heurísticas. Assim como nos estudos de sintonização anteriores, a cada ciclo do método heurístico são criadas 4 configurações candidatas arbitrárias no espaço de busca e, para cada uma delas, a meta-heurística foi executada durante 15 segundos sobre um conjunto de treinamento expandido formado por 14 exemplares do *benchmark* PCTSP.

O processo de sintonização foi repetido 10 vezes. Os resultados das sintonizações são apresentados em termos de média e desvio padrão ($\mu \pm \sigma$) calculados após dez execuções repetições do processo de sintonização.

As próximas Seções (7.2.1 e 7.2.2) apresentam as sintonizações das meta-heurísticas híbridas CS-SA e CS-GA pelo método HORA para o problema PCTSP. Os resultados computacionais produzidos após 5 execuções de cada meta-heurística são comparados com a literatura (CHAVES, 2009) referentes ao mesmo algoritmo com a sintonização original.

Os resultados foram avaliados estatisticamente por meio de testes de hipóteses t bilaterais para duas médias pareadas ao nível de significância de 10%, conforme descrito na Seção 6.4.2.

7.2.1 Sintonização da Meta-heurística CS-SA

O presente estudo trata da sintonização de 3 parâmetros da meta-heurística híbrida CS-SA por meio da metodologia para sintonização de meta-heurísticas previamente apresentada. As configurações iniciais para os estudos experimentais são apresentadas na Tabela 37.

Tabela 37 - Parâmetros da meta-heurística híbrida CS-SA selecionados para sintonização.

Parâmetros	Valores
T_0	[500000; 2000000]
SA_{\max}	[500; 1500]
α	[0,900; 0,980]

A sintonização da meta-heurística híbrida CS-SA para o PCTSP por meio do método HORA é apresentada na Tabela 38.

Tabela 38 - PCTSP: Sintonização da meta-heurística híbrida CS-SA pelo método HORA.

Parâmetros	HORA
T_0	1327324 ± 401413
SA_{\max}	987 ± 39
α	$0,947 \pm 0,001$
Tempo total	647

A partir dos resultados (Tabela 38) pode-se observar que os valores obtidos pelo método HORA são muito próximos da sintonização originalmente adotada na literatura (CHAVES, 2009). A maior diferença observada refere-se ao valor de T_0 , maior do que na sintonização original (Tabela 35). Assim como observado nos estudos de sintonização anteriores com outras meta-heurísticas e problemas, o método HORA é eficaz em termos de tempo do processo de sintonização, tal que a sintonização para a meta-heurística híbrida CS-SA é encontrada em 647 segundos.

A sintonização obtida pelo método HORA (Tabela 38) é empregada na configuração de uma versão do CS-SA, denominada CS-SA_H. O algoritmo foi executado cinco vezes sobre 42 exemplares do *benchmark* PCTSP.

Os resultados computacionais completos são apresentados no Apêndice C. Nesses resultados, são apresentadas as soluções obtidas pela meta-heurística CS-SA sintonizada pelo método HORA (CS-SA_H) e as soluções originalmente publicadas na literatura (CHAVES, 2009), denominada CS-SA_C. Devido à extensão dos resultados, apresenta-se um resumo (Tabela 39) por meio de valores médios dos seguintes parâmetros: número de exemplares em que uma versão encontra soluções de qualidade superior a outra (melhor), qualidade média das soluções (gap), robustez em relação à melhor solução encontrada solução média (desvio) e melhora do método HORA em relação à literatura (delta).

Tabela 39 - PCTSP: Resultados computacionais resumidos de diferentes versões do CS-SA.

	Versões	melhor	gap	desvio	delta
$\gamma \in [1, 100]$	CS-SA_H	10/42	1,85	0,92	--
	CS-SA_C	16/42	1,63	0,76	0,20
$\gamma \in [1, 1000]$	CS-SA_H	12/42	0,34	0,24	--
	CS-SA_C	9/42	0,32	0,38	0,02
$\gamma \in [1, 10000]$	CS-SA_H	12/42	0,23	0,17	-0,08
	CS-SA_C	5/42	0,32	0,19	--

A partir dos resultados (Tabela 39) pode-se observar que versão CS-SA_H encontra soluções de melhor qualidade (coluna "melhor") para os exemplares com maiores penalidades ($\gamma \in [1, 1000]$ e $\gamma \in [1, 10000]$).

A versão original da meta-heurística híbrida (CS-SA_C) se destaca em relação à qualidade das soluções (gap). No entanto, a versão CS-SA_H é melhor para os exemplares com maiores penalidades ($\gamma \in [1, 10000]$). As soluções da meta-heurística CS-SA_C são mais robustas (desvio), mas a robustez da versão CS-SA_H cresce progressivamente de acordo com o aumento no número de penalidades.

De acordo com os resultados a versão original da meta-heurística se destaca em termos de melhora da qualidade das soluções (delta). No entanto, a vantagem decresce conforme aumentam as penalidades, tal que a meta-heurística híbrida CS-SA_H apresenta melhora de 0,08% para os exemplares com maiores penalidades ($\gamma \in [1, 10000]$).

Em síntese, a sintonização da meta-heurística original (CS-SA_C) encontra as melhores soluções, mas a versão sintonizada pelo método HORA é competitiva em relação à qualidade das soluções (gap) e melhora na qualidade das soluções (delta). Tais observações são confirmadas por meio de testes de hipóteses ao nível de significância de 10% para comparação dos desempenhos das meta-heurísticas híbrida CS-SA antes (original) e após a sintonização pelo método HORA. Isto é, de acordo com os testes de hipóteses não é possível estabelecer uma diferença estatística entre os resultados computacionais produzidos pelas diferentes versões da CS-SA. Diante das evidências estatísticas tais resultados são considerados semelhantes.

7.2.2 Sintonização da Meta-heurística CS-GA

O presente estudo trata da sintonização de 4 parâmetros da meta-heurística híbrida CS-GA por meio da metodologia para sintonização de meta-heurísticas proposta nesta pesquisa. As configurações iniciais para os estudos experimentais são apresentadas na Tabela 40.

Tabela 40 - Parâmetros da meta-heurística híbrida CS-GA selecionados para sintonização.

Parâmetros	Valores
p_c	[0,40; 0,90]
p_m	[0,001; 0,025]
μ	[10; 100]
n	[500; 1500]

A sintonização da meta-heurística híbrida CS-GA para o PCTSP por meio do método HORA são apresentadas na Tabela 41.

Tabela 41 - PCTSP: Sintonização da meta-heurística híbrida CS-GA pelo método HORA.

Parâmetros	HORA
p_c	$0,691 \pm 0,212$
p_m	$0,022 \pm 0,007$
μ	93 ± 24
n	1310 ± 78
Tempo total	1304

A partir dos resultados (Tabela 41) pode-se observar que o método HORA encontra a sintonização para a meta-heurística CS-GA em 1304 segundos. Ao comparar estes resultados com a sintonização adotada originalmente na literatura (Tabela 36) é possível identificar divergências entre os valores obtidos pelo método HORA, principalmente, para os parâmetros p_c (probabilidade de cruzamento) e n (número de gerações).

A sintonização obtida por meio do método HORA (Tabela 41) é empregada na configuração de uma versão da meta-heurística híbrida CS-SA. O algoritmo foi executado cinco vezes sobre os 14 exemplares que compõem o *benchmark* PCTSP.

Os resultados computacionais completos são apresentados no Apêndice C. Nesses resultados, são apresentadas as soluções obtidas pela meta-heurística híbrida CS-GA sintonizada pelo método HORA (CS-GA_H) e as soluções originalmente publicadas na literatura (CHAVES, 2009), denominada CS-GA_C. Devido à extensão dos resultados, apresenta-se um resumo por meio de valores médios dos parâmetros nas Tabela 42. As colunas desta tabela correspondem ao número de exemplares em que uma versão encontra soluções de qualidade superior a outra (coluna "melhor"), qualidade média das soluções (gap), robustez em relação à melhor solução encontrada solução média (desvio) e melhora do método HORA em relação ao CS-GA original (delta).

A partir dos resultados (Tabela 42) pode-se observar que versão CS-GA_H produz as melhores soluções em termos de todos os parâmetros analisados (melhor, gap, desvio e delta). A meta-heurística híbrida CS-GA_H encontra soluções de melhor qualidade (coluna "melhor") para todos os exemplares do *benchmark* PCTSP, independente do valor das penalidades.

Tabela 42 - PCTSP: Resultados computacionais resumidos de diferentes versões do CS-GA.

	Versões	melhor	gap	desvio	delta
$\gamma \in [1, 100]$	CS-GA_H	27/45	0,72	0,65	-1,23
	CS-GA_C	0/45	2,07	0,96	--
$\gamma \in [1, 1000]$	CS-GA_H	18/45	0,05	0,07	-0,26
	CS-GA_C	1/45	0,31	0,28	--
$\gamma \in [1, 10000]$	CS-GA_H	15/45	-0,08	0,07	-0,22
	CS-GA_C	0/45	0,14	0,21	--

As soluções produzidas pela versão sintonizada pelo método HORA também apresenta soluções de melhor qualidade (gap) e soluções mais robustas (desvio) quando comparada com a versão original (CS-GA_C). Os melhores resultados (gap e desvio) são observados para os exemplares com número maior de penalidades, em que a versão CS-GA_H apresenta melhora de 0,08% na qualidade das soluções (gap) quando comparada com os resultados originais.

A sintonização da meta-heurística híbrida CS-GA pelo método HORA apresenta melhora na qualidade das soluções (delta). Em comparação com a versão original (CS-GA_C), os melhores resultados são observados para os exemplares com menores penalidades. Isto é, o CS-GA_H é 1,23% melhor em relação à versão original.

Em síntese, a meta-heurística CS-GA sintonizada pelo método HORA é melhor do que a versão original. A partir dos resultados computacionais e baseado em testes de hipóteses, conclui-se com 90% de confiança que a versão CS-GA_H apresenta melhor desempenho do que a versão original da meta-heurística CS-GA.

8. CONCLUSÃO

8.1 INTRODUÇÃO

A presente pesquisa propôs uma metodologia para sintonização de meta-heurísticas a partir da utilização conjunta de DOE e método de Corrida. Na fase inicial desta metodologia emprega-se DOE sobre um conjunto de treinamento formado por exemplares de uma classe de problema de otimização, com características representativas do problema. Esta metodologia utiliza DOE como uma ferramenta de exploração de uma meta-heurística sobre dado problema, cujo objetivo é definir um espaço de busca de parâmetros em uma região promissora.

Nesta abordagem considera-se que os estudos experimentais sobre um conjunto de treinamento pode gerar diversidade aos parâmetros, ao mesmo tempo que os aproxima de valores ideais para sua sintonização. Por isso, a definição de um espaço de busca de parâmetros é fundamental para intensificar a busca sobre regiões promissoras com soluções de qualidade superior.

A próxima fase da metodologia proposta emprega o método HORA, com a finalidade de explorar de maneira eficaz o espaço de busca de parâmetros previamente definido e automatizar o processo de sintonização. O HORA emprega o método de Corrida para avaliar e descartar as configurações candidatas de qualidade inferior durante o processo de sintonização. No entanto, o diferencial entre eles (HORA e Corrida) é a criação das alternativas no espaço de busca. Isto é, no método HORA as configurações candidatas são criadas sob demanda durante os ciclos do algoritmo. Esta característica garante uma dinâmica no número de alternativas avaliadas, tal que em algumas situações ele aumenta e em outras, diminui. Assim, o processo de avaliação torna-se mais eficiente.

A metodologia para sintonização de meta-heurísticas proposta nesta pesquisa foi aplicada sobre diferentes algoritmos (SA, GA e VNS) para resolução de dois problemas clássicos de otimização combinatória (TSP e TWTP). As sintonizações foram realizadas por meio dos métodos HORA e Corrida, e seus resultados comparados. Na primeira abordagem, as meta-heurísticas foram sintonizadas para resolução do TSP, enquanto que na segunda, a sintonização destinou-se à resolução do TWTP com número variado de atividades (40, 50 e 100 atividades).

8.2 REVISÃO DAS QUESTÕES DE PESQUISA

Os estudos sobre a utilização de técnicas para definição da metodologia para sintonização de meta-heurísticas, bem como para desenvolvimento do método HORA, permitiram encontrar respostas para as seguintes questões de pesquisa:

- Qual a influência dos parâmetros no desempenho das meta-heurísticas?
- Qual a configuração ideal para uma meta-heurística obter um desempenho satisfatório?
- Existe uma configuração única da meta-heurística para diferentes problemas?
- A metodologia para sintonização de meta-heurísticas é eficaz para um algoritmo já bem configurado?
- Os resultados obtidos demonstram que a metodologia pode ser uma ferramenta promissora?

A partir da aplicação da metodologia proposta pode-se identificar a influência dos parâmetros no desempenho das meta-heurísticas e determinar a sua configuração ideal para solução dos problemas de otimização. De acordo com os resultados das sintonizações, os parâmetros identificados como significantes para o desempenho das meta-heurísticas convergem para uma região comum do espaço de buscas de parâmetros, tal que seus valores são semelhantes.

A partir dos estudos experimentais com DOE é possível identificar que os parâmetros selecionados para os estudos de sintonização da meta-heurística SA (T_0 , valor da temperatura inicial; SA_{\max} , número de iterações a cada estágio da temperatura; e α , taxa de arrefecimento da temperatura) têm influência significativa no desempenho do algoritmo, independente do problema sob análise (TSP ou TWTP). Em relação à meta-heurística GA, os estudos experimentais indicam que os parâmetros p_m (probabilidade de cruzamento), p_c (probabilidade de mutação) e μ (tamanho da população) têm influência significativa no desempenho do algoritmo para solução do TSP. Os parâmetros p_m , μ e n (número de gerações) têm influência significativa no desempenho do GA sobre o TWTP. No entanto, para a meta-heurística VNS, os estudos experimentais indicam que o parâmetro k (número de estruturas de vizinhança)

influencia o desempenho do algoritmo sobre o TSP, enquanto que os parâmetros k e δ (distância entre as estruturas de vizinhança) influenciam o desempenho da VNS para o TWTP.

De acordo com os resultados dos processos de sintonizações, o método HORA é eficaz para encontrar as configurações ideais das diferentes meta-heurísticas (SA, GA e VNS) para resolução de dois problemas otimização (TSP e TWTP). Entretanto, a partir dos resultados não é possível identificar uma configuração única dos algoritmos para os problemas selecionados, mas os resultados indicam que a sintonização das meta-heurísticas para cada problema situa-se em uma região comum do espaço de buscas de parâmetros.

Os resultados computacionais demonstram que o HORA é competitivo em relação ao método de Corrida, mas é mais eficaz em termos de tempo do processo de sintonização. A comparação da qualidade das soluções produzidas por ambos métodos (HORA e Corrida) revelam que as sintonizações obtidas pelo método HORA apresentam melhora no desempenho das meta-heurísticas SA e VNS.

A eficácia da metodologia para sintonização de meta-heurísticas foi avaliada a partir de um estudo complementar para ajuste fino de um algoritmo já bem configurado. Para este estudo considerou-se a meta-heurística híbrida CS, em que o processo de agrupamento é realizado pelas meta-heurísticas SA e GA. Os resultados obtidos foram comparados com a literatura considerando-se os mesmos algoritmos (CS-SA e CS-GA) com as suas sintonizações originais. A meta-heurística híbrida CS-SA sintonizada pelo método HORA produz resultados competitivos em relação à configuração original da literatura, tal que as evidências estatísticas indicam que ambos são semelhantes. Os resultados da meta-heurística híbrida CS-GA sintonizada pelo método HORA são melhores do que os resultados da versão original. As análises estatísticas indicam com 90% de confiança, que a sintonização pelo método HORA melhora o desempenho da CS-GA e encontra soluções de melhor qualidade para diversos exemplares do TSP com Lucro.

Portanto, a partir dos resultados apresentados nesta pesquisa conclui-se que a metodologia para sintonização de meta-heurística por meio do método HORA é uma abordagem promissora que pode ser aplicada sobre diferentes meta-heurísticas para resolução de uma diversidade de problemas de otimização.

8.3 RESUMO DAS CONTRIBUIÇÕES

A principal contribuição desta pesquisa foi a definição de uma metodologia para sintonização de meta-heurísticas, que agrega conceitos de DOE e Corrida, e o desenvolvimento do método HORA. Por meio deste método heurístico é possível explorar de maneira eficaz o espaço de busca de parâmetros e perseguir diferentes configurações na vizinhança de uma configuração promissora.

Uma outra contribuição desta pesquisa corresponde à redução do esforço e tempo investidos na sintonização de meta-heurísticas, bem como na confiabilidade para escolha das configurações dos parâmetros e na possibilidade de replicação das sintonizações.

Durante o desenvolvimento da pesquisa foram efetuadas revisões bibliográficas e descrições a respeito de temas relacionados com a pesquisa, tais como:

- Meta-heurísticas utilizadas ao longo da pesquisa;
- Planejamento de experimentos;
- Algoritmos de Corrida;
- Problema de sintonização de meta-heurísticas; e
- Abordagens contemporâneas empregadas na sintonização de meta-heurísticas.

Destacam-se também como contribuições científicas desta pesquisa a redação de artigos científicos, apresentações de trabalhos em conferências nacionais e internacionais e participações em eventos regionais de Pós-Graduação:

- Um publicação em revista internacional:
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Improving the Performance of Metaheuristics: An Approach Combining Response Surface Methodology and Racing Algorithms. **International Journal of Engineering Mathematics**, Nasr City, v. 2015, p. 1-9, 2015.
- Um artigo submetido para avaliação em revista internacional CAIE (IF 1,783):
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. A Statistical Approach for the Fine-tuning of Metaheuristics: A Case Study Combining Design of Experiments and Racing Algorithms. **Computers & Industrial Engineering**, New York, 12 p., 2016.

- Um artigo submetido para avaliação em revista nacional G&P (Qualis B3):
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Uma Abordagem Estatística para a Sintonia-fina de Meta-heurísticas por meio de Planejamento de Experimentos. **Gestão e Produção**, São Carlos, 36 p., 2014.
- Uma publicação em conferência internacional:
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. A Statistical Approach for the Fine-tuning of Metaheuristics: A Case Study Combining Design of Experiments and Racing Algorithms. In: 45TH INTERNATIONAL CONFERENCE ON COMPUTERS & INDUSTRIAL ENGINEERING, 45., 2015, Metz. **45th International Conference on Computers and Industrial Engineering 2015 (CIE45)**, Red Hook: Curran Associates Inc., 2015. v. 1, p. 680-687.
- Duas publicações em conferências nacionais:
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Melhorando o Desempenho de Meta-heurísticas com o Auxílio de Planejamento de Experimentos. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 47., 2015, Porto de Galinhas. **Anais do XLVII SBPO**, Rio de Janeiro: SOBRAPO, 2015. p. 1874-1882.
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Sintonia-fina de Meta-heurísticas - Abordagem por meio de Planejamento de Experimentos. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 46., 2014, Salvador. **Anais do XLVI SBPO**, Rio de Janeiro: SOBRAPO, 2014. p. 3146.
- Duas publicações em jornadas de Pós-Graduação:
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Abordagem Estatística no Estudo da Sintonia-Fina de Meta-heurísticas. In: JORNADA CIENTÍFICA DA PÓS-GRADUAÇÃO DA UNESP, 2015, Guaratinguetá. **Livro de resumos**, Guaratinguetá: Faculdade de Engenharia de Guaratinguetá, 2015. p. 1-3.
 - BARBOSA, E. B. M.; SENNE, E. L. F.; SILVA, M. B. Análise do Desempenho de Meta-heurísticas por meio de Planejamento de Experimentos. In: JORNADA CIENTÍFICA DA PÓS-GRADUAÇÃO DA

UNESP, 2014, Guaratinguetá. **Livro de resumos**, Guaratinguetá: Faculdade de Engenharia de Guaratinguetá, 2014. p. 1.

8.4 TRABALHOS FUTUROS

Uma meta-heurística define alguns princípios de alto nível para explorar o espaço de soluções de problemas de otimização. Entretanto, para resolver um novo problema, algumas medidas devem ser tomadas antes de sua execução, tais como a definição da função objetivo, desenvolvimento de estruturas de vizinhança e sintonização de parâmetros.

A metodologia para sintonização de meta-heurísticas proposta nesta pesquisa envolve um campo de investigação destinado ao desenvolvimento de métodos, técnicas e procedimentos para otimização de meta-heurísticas. Embora os resultados obtidos sejam satisfatórios para os algoritmos e problemas selecionados, algumas melhorias podem ser consideradas.

Na fase inicial, a metodologia proposta conduz diferentes estudos experimentais sobre um conjunto de treinamento restrito formado por n exemplares de um problema de otimização. Por exemplo, nesta pesquisa adota-se $n = 4$ exemplares para compor o conjunto de treinamento para os estudos experimentais. Uma sugestão para melhoria consiste em considerar um número grande de exemplares, tal qual ocorre durante o processo de sintonização do método de Corrida, para a fase de estudos experimentais. Uma quantidade maior de exemplares de um mesmo problema deve aumentar a diversidade dos parâmetros, aproximar de seus valores ideais e determinar um espaço de busca mais preciso para aplicação do método HORA.

No contexto teórico, o método HORA realiza a avaliação das configurações candidatas por meio da estatística não paramétrica de Friedman. Uma sugestão de melhoria é a realização de estudos para definição da técnica estatística para avaliar a qualidade das alternativas no espaço de busca. Algumas técnicas empregadas na literatura com o método de Corrida (BIRATTARI et al., 2002 e 2009) são a correlação de Bonferroni e a estatística não paramétrica de Wilcoxon. No entanto, pode-se considerar também a avaliação de outras técnicas como, por exemplo, o teste de Tukey e o teste de Kruskal-Wallis.

Uma outra sugestão consiste na realização de estudos para analisar o problema de pseudo-replicação (em Inglês, *pseudoreplication*), isto é, a adoção incorreta dos graus de liberdade da inferência estatística, que resulta em altas taxas de falso positivo e, em algumas

situações, na eliminação de configurações candidatas de qualidade superior (CAMPELO et al., 2016).

Outros estudos podem considerar a estrutura de vizinhança para criação das soluções candidatas no espaço de busca empregada pelo método HORA. Na versão desenvolvida para esta pesquisa, as alternativas são criadas arbitrariamente no espaço de busca sem considerar qualquer informação prévia sobre os parâmetros. No entanto, um direcionamento com informações obtidas a partir dos estudos experimentais, por exemplo, sobre a significância dos parâmetros no desempenho das meta-heurísticas e suas interações, pode promover melhoria na qualidade das soluções obtidas.

A adaptação do método HORA para realização de sintonização de parâmetros *on-line* também deve ser explorada. A abordagem *on-line* permite a realização de ajustes nos valores dos parâmetros em tempo de execução. Essa capacidade de auto adaptação tem gerado interesse crescente na comunidade e pode contribuir para tornar o processo de sintonização completamente automático.

Sugere-se também a ampliação das pesquisas com o método HORA para avaliar a qualidade das soluções obtidas, principalmente, sobre os algoritmos já bem configurados e destinados à resolução de uma diversidade de problemas de otimização combinatória.

Outras sugestões estão relacionadas ao ambiente de computação científica. Isto é, para esta pesquisa adotou-se o software gratuito Scilab para desenvolvimento dos algoritmos computacionais. No entanto, tem-se observado no contexto científico uma forte tendência para adoção do software R (URL: <http://www.r-project.org>), um ambiente gratuito destinado à computação estatística e gráficos. Portanto, uma melhoria visando disseminação da metodologia para sintonização de meta-heurísticas consiste em adaptar os programas Scilab para o ambiente R. Ainda nesse contexto, para popularizar esta metodologia sugere-se o desenvolvimento de uma interface eletrônica amigável na Web, que, em síntese, receba o código-fonte dos programas e devolva a sua sintonização ideal.

REFERÊNCIAS BIBLIOGRÁFICAS

AARTS, E. H. L.; LENSTRA, J. K. **Local search in combinatorial optimization**. New Jersey: John Wiley & Sons Inc., 1997. 528 p.

ADESON-DIAZ, B.; LAGUNA, M. Fine-tuning of algorithms using fractional experimental designs and local search. **Operations Research**, Baltimore, v. 54, n. 1, p. 99-114, 2006.

ANGEL, R. D.; CAUDLE, W.L.; NOONAN, R.; WHINSTON, A. Computer assisted school bus scheduling. **Management Science**, Providence, v. 18, p. 279-88, 1972.

APPLEGATE, D. L.; BIXBY, R. E.; CHVÁTAL, V.; COOK, W. J. **The traveling salesman problem: a computational study**. 2nd ed., Princeton Series in Applied Mathematics, Princeton: Princeton University Press, 2007.

ARIN, A.; RABADI, G.; UNAL, R. Comparative studies on design of experiments for tuning parameters in a genetic algorithm for a scheduling problem. **International Journal of Experimental Design and Process Optimisation**, Olney, v. 2, n. 2, p. 103-124, 2011.

ARKIN, E.; CHIANG, Y. ; MITCHELL, J.; SKIENA, S.; YANG, T. On the maximum scatter TSP. In: 8TH ANNUAL ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS, 1997, New Orleans. **SODA '97 Proceedings of the eighth annual ACM-SIAM symposium on discrete algorithms**. Philadelphia: Society For Industrial And Applied Mathematics Philadelphia, 1997. v. 1, p. 211-220.

ARROYO, J. E. C.; SANTOS, A. G.; SILVA, F. L. S.; ARAÚJO, A. F. A GRASP With path relinking for the single machine total weighted tardiness problem. In: EIGHTH INTERNATIONAL CONFERENCE ON HYBRID INTELLIGENT SYSTEMS, 8., 2008, Barcelona. **8th International Conference on Hybrid Intelligent Systems (HIS)**, 2008. Red Hook: Curran Associates, 2008. v. 1, p. 726-731.

AWERBUCH, B.; AZAR, Y.; BLUM, A.; VEMPALA, S. New approximation guarantees for minimum weight k-trees and prize-collecting salesmen. **SIAM Journal on Computing**, Philadelphia, v. 28, n. 1, p. 254-262, 1998.

BABU, P.; PERIDY, L.; PINSON, E. A branch and bound algorithm to minimize total weighted tardiness on a single processor. **Annals of Operations Research**, Amsterdam, v. 129, p. 33-46, 2004.

BALAPRAKASH, P.; BIRATTARI, M.; STÜTZLE, T.; DORIGO, M. Improvement strategies for the F-Race algorithm: sampling design and iterative refinement. In: 4TH INTERNATIONAL WORKSHOP ON HYBRID METAHEURISTICS, 4., 2007, Dortmund. **Proceedings of the 4th International Workshop on Hybrid Metaheuristics (HM 2007)**, Berlin: Springer-Verlag, 2007. p. 108-122.

BALAS, E. The prize collecting traveling salesman problem. **Networks**, New York, v. 19, p. 621-636, 1989.

BARR, R.S.; GOLDEN, B.L.; KELLY, J.P.; RESENDE, M. G. C.; STEWART, W. R. Designing and reporting computational experiments with heuristic methods. **Journal of Heuristics**, Boston, v. 1, n. 1, p. 9-32, 1995.

BARTZ-BEIELSTEIN, T.; PARSOPOULOS, K. E.; VRAHATIS, M. N. Particle swarm optimizers for Pareto optimization with enhanced archiving techniques. In: THE 2003 CONGRESS ON EVOLUTIONARY COMPUTATION, 2003, Canberra. **Proceedings of the 2003 Congress on Evolutionary Computation (CEC'03)**, 2003, New York: IEEE, 2003. v. 2, p. 1780-1787.

BARTZ-BEIELSTEIN, T.; PARSOPOULOS, K. E.; VRAHATIS, M. N. Design and analysis of optimization algorithms using computational statistics. **Applied Numerical Analysis & Computational Mathematics**, New York, v. 1, n. 3, p. 413-433, 2004.

BARTZ-BEIELSTEIN, T.; PARSOPOULOS, K. E.; VRAHATIS, M. N. Analysis of particle swarm optimization using computational statistics. In: INTERNATIONAL CONFERENCE OF NUMERICAL ANALYSIS AND APPLIED MATHEMATICS 2004, Chalkis. **Proceedings of ICNAAM**, 2004, Weinheim: Wiley-VCH, 2004, p. 34-37.

BARVINOK, A.; TAMIR, A.; FEKETE, S.; WOEGINGER, G.; JOHNSON, D.; WOODROOFE, R. The geometric maximum traveling salesman problem. **Journal of the ACM**, New York, v. 50, n. 5, p. 641-664, 2003.

BEASLEY, J. E. OR-Library: Distributing test problems by electronic mail. **Journal of the Operational Research Society**, Oxford, v. 41, n. 11, p. 1069-1072, 1990.

BERES, D. L.; HAWKINS, D. M. Plackett-Burman technique for sensitivity analysis of many-parametered models. **Ecological Modeling**, Amsterdam, v. 141, p. 171-183, 2001.

BERTSIMAS, D.; TSITSIKLIS, J. Simulated annealing. **Statistical Science**, Hayward, v. 8, n. 1, p. 10-15, 1993.

BIONDI NETO, L.; BECCENERI, J. C. ; SILVA, J. D. S.; SILVA NETO, J. A. Fundamentos de otimização e inteligência artificial. In: SILVA NETO, A. J., BECCENERI, J. C. **Técnicas de inteligência computacional inspiradas na natureza** - aplicação em problemas inversos em transferência radiativa. 41 ed. São Carlos: SBMAC, 2009. Cap. 4. p. 35-42.

BIRATTARI, M.; STÜTZLE, T.; PAQUETE, L.; VARRENTRAPP, K. A racing algorithm for configuring metaheuristics. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, 2002, New York. **Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002)**. New York: Morgan Kaufmann, 2002, p. 11-18.

BIRATTARI, M.; YUAN, Z.; BALAPRAKASH, P.; STÜTZLE, T. **F-Race and iterated F-Race**: an overview. Bruxelles: Iridia Technical Report Series, 2009. 21 p.

BIRATTARI, M. **Tuning metaheuristics**: a machine learning perspective. 2nd ed. New York: Springer, 2009. 221 p.

BONFERRONI, C. E. **Il calcolo delle assicurazioni su gruppi di teste**: studi in onore del professore Salvatore Ortu Carboni, Roma: Tipografia del Senato, 1935, 50 p.

BONFERRONI, C. E. **Teoria statistica delle classi e calcolo delle probabilità**. Firenze: Istituto superiore di scienze economiche e commerciali di Firenze 8, 1936. 6 p.

BLUM, C.; ROLI, A. Metaheuristics in combinatorial optimization: overview and conceptual comparison. **ACM Computing Surveys**, New York, v. 35, n. 3, p. 268-308, 2003.

BOZEJKO, W.; GRABOWSKI, J.; WODECKI, M. Block approach - Tabu search algorithm for single machine total weighted tardiness problem. **Computers & Industrial Engineering**, New York, v. 50, p. 1-14, 2006.

BOX, G. E. P.; WILSON, K. B. On the experimental attainment of optimum conditions. **Journal of the Royal Statistical Society Series B**, Oxford, v. 13, p. 1-45, 1951.

BOX, G. E. P.; HUNTER, W. G.; HUNTER, J. S. **Statistics for experimenters: an introduction to design, data analysis, and model building**. New Jersey: John Wiley & Sons, 1978.

BRADLEY, N. **The response surface methodology**. 2007. 84 f. Dissertação (Mestrado em Matemática) - Indiana University South Bend, Indiana, 2007.

BREST, J.; GREINER, S.; BOSKOVIC, B.; MERNIK, M.; ZUMER, V. Self-Adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. **IEEE Transactions on Evolutionary Computation**, New York, v. 10, n. 6, p. 646-657, 2006.

CAMPELO, F.; TRINDADE, A. R.; LÓPEZ-IBÁÑEZ, M. Pseudoreplication in racing methods for tuning metaheuristics. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, 2016, Denver. **Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2016)**. 2016. No prelo.

CHAN, K. Y.; AYDIN, M. E.; FORGATY, T. C. Main effect fine-tuning of the mutation operator and the neighbourhood function for uncapacitated facility location problems. **Soft Computing**, New York, v. 10, n. 11, p. 1075-1090, 2006.

CHAVES, A. A.; LORENA, L. A. N. Aplicação do algoritmo clustering search aos traveling salesman problems with profits. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 39., 2007, Fortaleza. **Anais do XXXIX SBPO**, Rio de Janeiro: SOBRAPO, p. 1472-1483.

CHAVES, A. A. **Uma meta-heurística híbrida com busca por agrupamentos aplicada a problemas de otimização combinatória**. 2009. 197 f. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009.

CERNY, V. Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. **Journal of Optimization Theory and Applications**, New York, v. 45, p. 41-51, 1985.

COLEY, D. A. **Introduction to genetic algorithms for scientists and engineers**. London: Wspc, 1999, 244 p.

CONOVER, W. J. **Practical nonparametric statistics**. 3rd ed. , New York: Wiley, 1999, 584 p.

COOK, S. A. The complexity of theorem-proving procedures. In: **THIRD ANNUAL ACM SYMPOSIUM ON THEORY OF COMPUTING**, New York, 1971. **STOC '71 Proceedings of the Third Annual ACM Symposium on Theory of Computing**, New York: ACM, 1971, p. 151-158.

COSCIONE, A. R.; ANDRADE, J. C.; MAY, G. M. O modelamento estatístico de misturas: experimento tutorial usando voltametria de redissolução anódica. **Química Nova**, São Paulo, v. 28, n. 6, p. 1116-1122, 2005.

CRAUWELS, H. A. J.; POTTS, C. N.; VAN WASSENHOVE, L. N. Local search heuristics for single machine total weighted tardiness scheduling problem. **Infoms Journal on Computing**, Linthicum, v. 10, p. 341-350, 1998.

COY, S.; GOLDEN, B. L.; RUNGER, G. C.; WASIL, E. A. Using experimental design to find effective parameter settings for heuristics. **Journal of Heuristics**, Boston, v. 7, n. 1, p. 77-97, 2000.

DANTZIG, G.B.; FULKERSON, D.R.; JOHNSON, S.M. Solution of a large-scale traveling salesman problem. **Operations Research**, Baltimore, v. 2, p. 393-410, 1954.

DEAN, A.; E VOSS, D. **Design and analysis of experiments**. New York: Springer, 1999. 742 p.

DELL'AMICO, M.; MAFFIOLI, F.; VÄRBRAND, P. On prize collecting tours and the asymmetric traveling salesman problem. **International Transactions in Operational Research**, Oxford, v. 2, n. 3, p. 297-308, 1995.

DERIGS, U. **Optimization and operations research**. v. 2, Paris: EOLSS Publishers Co. Ltd., 2009. 273 p.

DOBSLAW, F. A parameter tuning framework for metaheuristics based on design of experiments and artificial neural networks. In: **2010 SIXTH INTERNATIONAL CONFERENCE ON NATURAL COMPUTATION**, 6., 2010, Cairo. **Proceeding of the ICNC 2010**. New York: IEEE, 2010, p. 1-4.

DUARTE, J. B. **Princípios sobre delineamentos em experimentação agrícola**. 1996. 66 f. Dissertação (Mestrado em Estatística) - Universidade Federal de Goiás, Goiânia, 1996.

EIBEN, A. E.; SMITH, J. E. **Introduction to evolutionary computing**. Berlin: Springer, 2003. 300 p.

EIBEN, A. E.; SMITH, J. E. Evolutionary algorithm parameters and methods to tune them. In: HAMADI, Y.; MONFROY, E.; SAUBION, F. **Autonomous Search**, Berlin: Springer Berlin Heidelberg, 2011. Chapter 2. p. 15-36.

FEILLET, D.; DEJAX, P.; GENDREAU, M. Traveling salesman problems with profits. **Transportation Science**, Baltimore, v. 2, n. 39, p. 188-205, 2005.

FIELD, A. **Descobriendo estadística usando SPSS**. 2 ed. Porto Alegre: Artmed, 2009. 688 p.

GARCÍA, S.; FERNÁNDEZ, A.; LUENGO, J.; HERRERA, F. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. **Information Sciences**, New York, v. 180, p. 2044-2064, 2010.

GLOVER, F. Future paths for integer programming and links to artificial intelligence. **Comp. Oper. Res.**, New York, v. 13, p. 533-549, 1986.

GLOVER, F.; LAGUNA, M. **Tabu search**. New York: Springer, 1997. 382 p.

GLOVER, F.; LAGUNA, M.; MARTÍ, R. Scatter search and path relinking: advances and applications. In: GLOVER, F.; KOCHENBERGER, G. **Handbook of metaheuristics**. v. 57. New York: Springer, 2003. Cap. 4. p. 1-21. (International Series in Operations Research & Management Science)

GLOVER, F.; KELLY, J.; LAGUNA, M. Practical introduction to simulation optimization. In: THE 2003 WINTER SIMULATION CONFERENCE, 2003, New Orleans. **Proceedings of the 2003 Winter Simulation Conference**. New York: Springer, 2003, p. 71-78.

GOLDEN, B. L.; LEVY, L.; VOHRA, R. The orienteering problem. **Naval Research Logistics**, New York, v. 34, n. 3, p. 307-318, 1987.

GRAHAM, R. L.; LAWLER, E. L.; LENSTRA, J. K.; RINNOOY KAN, A. H. G. Optimization and approximation in deterministic sequencing and scheduling: a survey. **Annals of Discrete Math**, Amsterdam, v. 4, p. 287-326, 1979.

HAMMING, R. W. Error detecting and error correcting codes. **Bell System Technical Journal**, New York, v. 26, n. 2, p. 147-160, 1950.

HANSEN, P.; MLADENOVIC, N. An introduction to variable neighborhood search. In: VOB, S.; MARTELLO, S.; OSMAN, I.; ROUCAIROL, C. **Metaheuristics: advances and trends in local search paradigms for optimization**, Berlin: Kluwer Academic Publishers, Chapter 30, p. 433-458, 1999.

HANSEN, P.; MLADENOVIC, N. Variable neighborhood search: principles and applications. **European Journal of Operational Research**, Amsterdam, v. 130, p. 449-467, 2001.

HOEFFDING, W. Probability inequalities for sums of bounded random variables. **Journal of the American Statistical Association**, New York, v. 58, n. 301, p. 13-30, 1963.

HOLLAND, J. H. **Adaptation in natural and artificial systems**. Boston: University of Michigan Press, 1975. 211 p.

HUTTER, F.; HAMADI, Y.; HOOS, H.; LEYTON-BROWN, K. Performance prediction and automated tuning of randomized and parametric algorithms. In: 12th. INTERNATIONAL CONFERENCE ON PRINCIPLES AND PRACTICE OF CONSTRAINT

PROGRAMMING, 12., 2006, Nantes. **Principles and Practice of Constraint Programming - CP 2006**, Berlin: Springer Berlin Heidelberg, 2006. p. 213-228.

HUTTER, F.; HOOS, H.; STÜTZLE, T. Automatic algorithm configuration based on local search. In: Twenty-Second Conference on Artificial Intelligence (AAAI-07), 22., 2007, Vancouver. **Proceedings of the 22nd National Conference on Artificial Intelligence**, Palo Alto: AAAI Press, 2007. p. 1152-1157.

HUTTER, F.; HOOS, H.; LEYTON-BROWN, K.; STÜTZLE, T. ParamILS: an automatic algorithm configuration framework. **Journal of Artificial Intelligence Research**, El Segundo, v. 36, p. 267-306, 2009.

ILOG. **Ilog Cplex 11.0**. Paris: ILOG Inc., 2006. 478 p.

JONES, T. **Evolutionary algorithms, fitness landscapes and search**. 1995. 250 f. Tese (Doutorado em Ciência da Computação), University of New Mexico, Albuquerque, 1995.

JONES, T. **One operator, one landscape**. Santa Fe: Santa Fe Institute Technical Report, 1995. 16 p.

JOUGLET, A.; BAPTISTE, P.; CARLIER, J. Exact procedures for single machine total cost scheduling. In: IEEE INT. CONF. SYSTEMS, MAN AND CYBERNETICS, 2002, Yasmine Hammamet. **Proceedings of the 2002 IEEE International Conference on Systems, Man and Cybernetics**, New York: IEEE, 2002. p. 1-4.

KHOSHJAHAN, Y.; NAJAFI, A. A.; AFSHAR-NADJAFI, B. Resource constrained project scheduling problem with discounted earliness–tardiness penalties: mathematical modeling and solving procedure. **Computers & Industrial Engineering**, New York, v. 66, n. 2, p. 293-300, 2013.

KHURI, A. I.; MUKHOPADHYAY, S. Response surface methodology. **Wires Computational Statistics**, New York, p. 128-149, 2010.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing. **Science**, London, v. 220, n. 4598, p. 671-680, 1983.

KOULAMAS, C. The single-machine total tardiness scheduling problem: Review and extensions. **European Journal of Operational Research**, Amsterdam, v. 202, p. 1-7, 2010.

KRAMER, O.; GLOGER, B.; GOEBELS, A. An experimental analysis of evolution strategies and particle swarm optimisers using design of experiments. In: GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, 9., 2007, New York. **Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'07)**. New York: ACM, 2007, p. 674-681.

KRAMER, O. Evolutionary self-adaptation: a survey of operators and strategy parameters. **Evolutionary Intelligence** 3, Berlin, p. 51–65, 2010.

LAPORTE, G. What you should know about the vehicle routing problem. **Naval Research Logistics**, New York, v. 54, p. 811-819, 2007.

- LENSTRA, J. K.; RINNOOYKAN, A. H. G.; BRUCKER, P. Complexity of machine scheduling problems. In: HAMMER, P. L.; JOHNSON, E. L.; KORTE, B. H.; NEMHAUSER, G. L. **Annals of Discrete Mathematics**, Amsterdam: Elsevier, 1977, p. 343–362.
- LESSMANN, S.; CASERTA, M.; ARANGO, I. Tuning metaheuristics: a data mining based approach for particle swarm optimization. **Expert Systems with Applications**, New York, v. 38, n. 10, p. 12826-12838, 2011.
- LIU, N.; ABDELRAHMAN, M.; RAMASWAMY, S. A genetic algorithm for single machine total weighted tardiness scheduling problem. **International Journal of Intelligent Control and Systems**, Singapore, v. 10, p. 218-225, 2005.
- MATAI, R.; SINGH, S. P.; MITTAL, M. L. Traveling salesman problem: an overview of applications, formulations, and solution approaches. In: DAVENDRA, D. **Traveling salesman problem: theory and applications**, Chapter 1, Rijeka: Intech, 2010, p. 1-24.
- MARON, O. **Hoeffding races: model selection for MRI classification**. 1994. 61 f. Tese (Doutorado em Ciência da Computação e Engenharia Elétrica) - Massachusetts Institute of Technology, Cambridge, 1994.
- MARON, O.; MOORE, A. W. Hoeffding races: accelerating model selection search for classification and function approximation. **Advances in Neural Information Processing Systems**, San Mateo, p. 59-66, 1994.
- MARON, O.; MOORE, A. W. The racing algorithm: model selection for lazy learners. **Artificial Intelligence review**, Dordrecht, v. 11, n. 1-5, p. 193-225, 1997.
- MENDENHALL, W.; SCHEAFFER, R. L.; E WACKERLY, D. D. **Mathematical statistics with applications**. Boston: Thomson Brooks, 2008. 944 p.
- METROPOLIS, N.; ROSENBLUTH, A.; ROSENBLUTH, M.; TELLER A.; TELLER, E. Equation of state calculations by fast computing machines. **Journal of Chemical Physics**, New York, vol. 21, p. 1087-1092, 1953.
- MILLER, D.; PEKNY, J. Exact solution of large asymmetric traveling salesman problems. **Science**, London, v. 251, p. 754-761, 1991.
- MITCHELL, M. **An introduction to genetic algorithms**. Massachusetts: MIT Press, 1998. 221 p.
- MONTGOMERY, D. C. **Design and analysis of experiments**. 8th ed. New Jersey: John Wiley & Sons Inc., 2012. 699 p.
- MOORE, A. W.; LEE, M. S. Efficient algorithms for minimizing cross validation error. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 11., 1994, New Brunswick. **Proceedings of the 11th International Conference on Machine Learning**, San Francisco: Morgan Kaufmann, 1994. p. 190-198.

NEARCHOU, A. C. Solving the single machine total weighted tardiness scheduling problem using a hybrid simulated annealing algorithm. In: 2004 2th. IEEE INT. CONF. ON INDUSTRIAL INFORMATICS, 2., 2004, Berlin. **2004 2nd IEEE International Conference on Industrial Informatics (INDIN '04)**, New York: IEEE, 2004. p. 513-516.

ÓLAFSSON, S. Metaheuristics. In: HENDERSON, S.; NELSON, B. **Handbooks in operations research and management science: simulation**. Amsterdam: Elsevier, 2006. Chapter 21, p. 633-654.

OLIVEIRA, A. C. M.; LORENA, L. A. N. Detecting promising areas by evolutionary clustering search. In: BAZZAN, A. L. C.; LABIDI, S. **Advances in artificial intelligence: SBIA 2004**. Berlin: Springer, 2004. Chapter 39. p. 385-394.

OLIVEIRA, A. C. M.; LORENA, L. A. N. Pattern sequencing problems by clustering search. In: SICHTMAN, J. S.; COELHO, H.; REZENDE, S. O. **Advances in artificial intelligence: IBERAMIA-SBIA 2006**. Berlin: Springer, Chapter 26, v. 4140, 2006, p. 218-227.

PLACKETT, R. L.; BURMAN, J. P. The design of optimum multifactorial experiments. **Biometrika**, London, v. 33, n. 4, p. 305-25, 1946.

PINEDO, M. **Scheduling: theory, algorithms and systems**. 2nd ed. New Jersey: Prentice Hall, 2002. 676 p.

PRESS, W. H.; TEUKOLSKY, S. A.; VETTERLING, W. T.; FLANNERY, B. P. **Numerical recipes: the art of scientific computing**. New York: Cambridge University Press, 1986. 1256 p.

REINELT, G. TSPLIB: A traveling salesman problem library. **ORSA Journal on Computing**, Baltimore, v. 3, n. 4, p. 376-384, 1991.

RIDGE, E.; KUDENKO, D. Tuning the performance of the MMAS heuristic. In: STÜTZLE, T.; BIRATTARI, M.; HOOS, H. H. **Engineering stochastic local search algorithms: designing, implementing and analyzing effective heuristics**. New York: Springer, v. 4638, 2007. p. 46-60.

ROTHLAUF, F. **Design of modern heuristics**. Berlin: Springer-Verlag Berlin Heidelberg, 2011. 225 p.

SCHMIDT, G. Scheduling with limited machine availability. **European Journal of Operational Research**, Amsterdam, v. 121, p. 1-15, 2000.

SHAHSAVAR, M.; NAJAFI, A. A.; NIAKI, S. T. A. Statistical design of genetic algorithms for combinatorial optimization problems. **Mathematical Problems in Engineering**, Nasr City, v. 2011, p. 1-17, 2011.

SHESKIN, D. **Handbook of parametric and nonparametric statistical procedures**. 2nd ed. Boca Raton: Chapman & Hall/CRC, 2000. 1926 p.

SMITH, J. E. Self-adaptation in evolutionary algorithms for combinatorial optimisation. adaptive and multilevel metaheuristics. In: COTTA, C.; SEVAUX, M.; SÖRENSEN, K. **Adaptive and Multilevel Metaheuristics**, New York: Springer, 2008. Chapter 1, p. 31-57.

SMIT, S. K.; EIBEN, A. E. Comparing parameter tuning methods for evolutionary algorithms. In: 2009 IEEE CONGRESS ON EVOLUTIONARY COMPUTATION, 2009, Trondheim. **2009 IEEE Congress on Evolutionary Computation (CEC'09)**, 2009. New York: IEEE. p. 399-406.

SÖRENSEN, K. Metaheuristics - The metaphor exposed. **International Transactions in Operational Research**, Oxford, v. 22 1, p. 3-18, 2015.

STÜTZLE, T. **Iterated local search for the quadratic assignment problem**. Darmstadt: Technische Universität Darmstadt Technical Report, 1999. 27 p. (AIDA-99-03).

STYLES, J.; HOOS, H. Using racing to automatically configure algorithms for scaling performance. In: NICOSIA, G.; PARDALOS, P. **Learning and intelligent optimization**, Berlin: Springer Berlin Heidelberg, 2013. Chapter 40, v. 7997, p. 382-388.

TAGUCHI, G.; KONISHI, S. **Orthogonal arrays and linear graphs**. Michigan: American Supplier Institute Press, 1987. 486 p.

TALBI, E.G. **Metaheuristics: from design to implementation**. New Jersey: John Wiley & Sons Inc., 2003. 593 p.

THEDE, S. M. An introduction to genetic algorithms. **Journal of Computing Sciences in Colleges**, Evansville, v. 20, n. 1, p. 115-123, 2004.

TORRES, R. D.; BRITO, J. A. M. Problemas de coleta de prêmios seletiva. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 35., 2003, Natal. **Anais do XXXV SBPO**, Rio de Janeiro: SOBRAPO, 2003. p. 1359-1371.

TSILIGIRIDES, T. Heuristic methods applied to orienteering. **Journal of the Operational Research Society**, Oxford, v. 35, n. 9, p. 797-809, 1984.

XU, J.; KELLY, J. A network flow-based Tabu search heuristic for the vehicle routing problem. **Transportation Science**, Baltimore, v. 30, p. 379-393, 1996.

YUAN, Z.; GALLAGHER, M. Combining meta-eas and racing for difficult ea parameter tuning tasks. In: LOBO, F. G., LIMA, C. F., MICHALEWICZ, Z. **Parameter setting in evolutionary algorithms**, Berlin: Springer Berlin Heidelberg, 2007. Chapter 6, v. 54, p. 121-142.

YUAN, Z.; STÜTZLE, T.; BIRATTARI, M. **MADS/F-Race: mesh adaptive direct search meets F-Race**. Bruxelles: Iridia Technical Report Series, 2010. 10 p.

WANG, X.; TANG, L. A population-based variable neighborhood search for the single machine total weighted tardiness problem. **Computers & Operations Research**, New York, v. 36, p. 2105-2110, 2009.

APÊNCICE A - Metodologia para Sintonização de Meta-heurísticas

A.1 META-HEURÍSTICA GA

Este apêndice apresenta a sequência de passos empregada pela metodologia para sintonização de meta-heurísticas. Para este estudo, considera-se a meta-heurística GA aplicada sobre o TWTP com 40 atividades. Os parâmetros selecionados para sintonização, bem como seus intervalos iniciais para os estudos experimentais são apresentados na Tabela 43.

Tabela 43 - Parâmetros da meta-heurística GA selecionados para sintonização.

Parâmetros	Descrições	Valores
p_c	Probabilidade cruzamento	[0,40; 0,90]
p_m	Probabilidade de mutação	[0,001; 0,025]
μ	Tamanho da população	[10; 100]
n	Número de gerações	[100; 1000]

O processo de sintonização inicia-se a partir de estudos experimentais com DOE sobre um conjunto de treinamento formado por n exemplares ($n > 1$) selecionados arbitrariamente do *benchmark* wt40 da biblioteca OR-Library (BEASLEY, 1990). Este *benchmark* é composto por 125 exemplares com 40 atividades numerados sequencialmente a partir de 1 até 125. Nesta pesquisa, o conjunto de treinamento, com $n = 4$ exemplares, é formado pelos seguintes exemplares: 19, 42, 63 e 87.

Para os estudos experimentais com DOE, os parâmetros das meta-heurísticas são associados aos fatores dos planejamentos fatoriais e seus valores, aos níveis (coluna Valores, Tabela 43). A matriz experimental do fatorial completo $2^{k=4}$ com um ponto central, em que k corresponde ao número de fatores, é apresentada na Tabela 44. Nesta tabela, os parâmetros e níveis são apresentados em termos de variáveis codificadas. Isto é, os fatores A , B , C e D estão associados aos parâmetros p_c , p_m , μ e n , respectivamente, da meta-heurística GA, os valores +1 e -1, são os níveis alto e baixo dos fatores (coluna Valores, Tabela 43), e 0 (zero) é o ponto central. Os resultados do estudo experimental com duas réplicas para o exemplar 42 do conjunto de treinamento, expressos em termos do valor da função objetivo, são apresentados na Tabela 44.

Tabela 44 - Planejamento fatorial 2^4 para sintonização da meta-heurística GA.

Exp.	A	B	C	D	Réplica 1	Réplica 2
1	-1	-1	-1	-1	84913	84115
2	+1	-1	-1	-1	85302	84035
3	-1	+1	-1	-1	83502	83744
4	+1	+1	-1	-1	84275	83192
5	-1	-1	+1	-1	82622	82281
6	+1	-1	+1	-1	81696	82215
7	-1	+1	+1	-1	82009	81660
8	+1	+1	+1	-1	81855	81777
9	-1	-1	-1	+1	85050	83246
10	+1	-1	-1	+1	83303	83734
11	-1	+1	-1	+1	82568	82241
12	+1	+1	-1	+1	81937	81652
13	-1	-1	+1	+1	81795	82093
14	+1	-1	+1	+1	81749	82215
15	-1	+1	+1	+1	81666	81643
16	+1	+1	+1	+1	81670	81819
17	0	0	0	0	82624	82361

Na Tabela 45 é apresentada a ANOVA correspondente ao planejamento fatorial 2^4 (Tabela 44). No contexto de DOE, a ANOVA auxilia na identificação da influência dos fatores no desempenho das meta-heurísticas. Nesta tabela, SQ corresponde à soma dos quadrados dos fatores e de suas interações; gl representa o número de graus de liberdade; QM é o quadrado médio; F , o valor computado da estatística F; e valor- p estabelece a taxa de contribuição dos fatores no processo.

Tabela 45 - ANOVA de um planejamento fatorial 2^4 (Exemplar 42).

Fontes de Variação	SQ	gl	QM	F	valor- p
A	724507	1	724507	0,7905	0,383161
B	29700851	1	29700851	32,4052	0,000009
C	99422626	1	99422626	108,4755	0,000000
D	12586399	1	12586399	13,7324	0,001165
AxB	94287	1	94287	0,1029	0,751306
AxC	136634	1	136634	0,1491	0,702970
AxD	195156	1	195156	0,2129	0,648818
BxC	17406475	1	17406475	18,9914	0,000231
BxD	4233323	1	4233323	4,6188	0,042380
CxD	12341754	1	12341754	13,4655	0,001273
Resíduos	21080532	23	916545		
Total	197922543	33			

A partir da ANOVA observa-se que os fatores B , C e D exercem influência significativa (valor- $p < 0,10$) no desempenho da meta-heurística GA sobre o exemplar 42 do *benchmark* wt40. Na ANOVA é possível identificar também a presença de interações significativas entre os seguintes fatores: $B \times C$ e $B \times D$ e $C \times D$.

Em síntese, os resultados da ANOVA confeccionados para os demais exemplares do conjunto de treinamento revelam que os mesmos fatores (B , C e D) têm influência significativa no desempenho do algoritmo, independente do exemplar selecionado. Entretanto, as interações entre os fatores são distintas, de acordo com o exemplar selecionado.

Em seguida, os parâmetros foram explorados a partir do ponto médio¹⁵ com o método do caminho de máxima ascensão¹⁶ (em Inglês, *steepest ascent method*) até encontrar uma região promissora, em que a qualidade das soluções é superior em relação à região inicialmente selecionada (Tabela 46).

Tabela 46 - Região promissora identificada durante o processo de sintonização da meta-heurística GA.

Parâmetros	Valores
p_c	[0,671; 0,693]
p_m	[0,020; 0,026]
μ	[65; 75]
n	[710; 870]

Nesta região (Tabela 46), são realizados estudos experimentais adicionais por meio da Metodologia da Superfície de Respostas, a partir de um planejamento fatorial 2^4 acrescido de cinco pontos centrais e quatro pontos axiais (Tabela 47), para detectar o ponto significativo da curvatura. A significância da curvatura implica que as configurações de parâmetros estão próximas de extremos (máximo ou mínimo), cujos valores são utilizados na definição de um espaço de busca de parâmetros.

Na Tabela 47, os níveis dos fatores estão codificados, tal que +1 e -1 correspondem aos níveis alto e baixo (coluna Valores, Tabela 46), respectivamente, os pontos centrais são representados por 0 (zero) e os pontos axiais do CCD são os valores +2 e -2. Os resultados do

¹⁵ $ponto_médio_p = (l_s + l_i) / 2$, em que p é um parâmetro da meta-heurística e l_s e l_i são seus limites superior e inferior, respectivamente.

¹⁶ O caminho de máxima ascensão é um procedimento de movimentos sequenciais em direção a uma região com aumento significativo na qualidade das soluções (MONTGOMERY, 2012).

estudo experimental sobre o exemplar 42 do conjunto de treinamento são apresentados na Tabela 47, em termos do valor da função objetivo.

Tabela 47 - Planejamento CCD para sintonização da meta-heurística GA.

Exp.	A	B	C	D	Resultados
1	-1	-1	-1	-1	81873
2	-1	-1	-1	+1	82089
3	-1	-1	+1	-1	81737
4	-1	-1	+1	+1	81815
5	-1	+1	-1	-1	81680
6	-1	+1	-1	+1	81713
7	-1	+1	+1	-1	81701
8	-1	+1	+1	+1	81674
9	+1	-1	-1	-1	81832
10	+1	-1	-1	+1	82072
11	+1	-1	+1	-1	81761
12	+1	-1	+1	+1	81851
13	+1	+1	-1	-1	82020
14	+1	+1	-1	+1	81794
15	+1	+1	+1	-1	81606
16	+1	+1	+1	+1	81638
17	-2	0	0	0	81758
18	+2	0	0	0	81762
19	0	-2	0	0	82194
20	0	+2	0	0	81512
21	0	0	-2	0	81742
22	0	0	+2	0	81887
23	0	0	0	-2	81815
24	0	0	0	+2	82056
25	0	0	0	0	81955
26	0	0	0	0	82003
27	0	0	0	0	82057
28	0	0	0	0	82134
29	0	0	0	0	82134

A ANOVA do CCD é apresentada na Tabela 48. A partir dos resultados pode-se observar que os fatores *B*, *C* e *D* (lineares e quadráticos) exercem influência significativa no desempenho da meta-heurística GA para solução do exemplar 42. A falta de ajuste insignificante ($\alpha > 0,10$) identificada nestes resultados indica que o modelo de regressão polinomial de segunda ordem descrito pela ANOVA é adequado para explicar a relação de causa e efeito entre resposta e fatores.

Tabela 48 - ANOVA de um planejamento CCD (Exemplar 42).

Fontes de Variação	SQ	gl	QM	F	valor-p
A	3750,0	1	3750,0	0,59578	0,483263
A²	167545,0	1	167545,0	26,61852	0,006701
B	274776,0	1	274776,0	43,65474	0,002720
B²	84619,1	1	84619,1	13,44377	0,021456
C	41666,7	1	41666,7	6,61975	0,061790
C²	115546,1	1	115546,1	18,35725	0,012801
D	35113,5	1	35113,5	5,57862	0,077501
D²	34534,9	1	34534,9	5,48670	0,079173
AxB	5184,0	1	5184,0	0,82360	0,415459
AxC	11772,2	1	11772,2	1,87030	0,243243
AxD	1681,0	1	1681,0	0,26707	0,632565
BxC	812,2	1	812,2	0,12905	0,737584
BxD	41209,0	1	41209,0	6,54703	0,062723
CxD	506,3	1	506,3	0,08043	0,790789
Falta de ajuste	194992,4	10	19499,2	3,09792	0,143573
Resíduos	25177,2	4	6294,3		
Total	909019,0	28			

Os estudos experimentais descritos anteriormente são repetidos para os n exemplares que formam o conjunto de treinamento, tal que o resultado final são 4 alternativas distintas de configurações, cada uma relacionada a um exemplar (Tabela 49).

Tabela 49 - Configurações da meta-heurística GA para os exemplares do conjunto de treinamento.

Parâmetros	Exemplar 19	Exemplar 42	Exemplar 63	Exemplar 87
p_c	0,684	0,698	0,690	0,725
p_m	0,024	0,014	0,028	0,057
μ	69	70	73	101
n	775	1267	828	1134

As configurações identificadas garantem diversidade aos parâmetros e permitem definir um intervalo de valores para cada um deles, denominado espaço de busca de parâmetros (Tabela 50), cujos limites são os valores máximo e mínimo de cada parâmetro identificados por meio de estudos experimentais.

Tabela 50 - Espaço de busca de parâmetro da meta-heurística GA.

Parâmetros	Espaço de Busca	Níveis
p_c	[0,684; 0,725]	4
p_m	[0.014; 0,057]	4
μ	[69; 101]	3
n	[775; 1267]	4
Total		192

Destaca-se nestes resultados que os limites dos parâmetros p_m , μ e n situam-se fora dos limites delimitados inicialmente para os estudos experimentais (Tabela 43). Esta extrapolação é justificada pelos pontos axiais presentes nos estudos experimentais com a Metodologia da Superfície de Respostas, cuja superação dos limites visa garantir rotatividade aos planejamentos e estimativas adequadas dos termos quadráticos dos modelos de regressão polinomiais.

A sequência da metodologia para sintonização de meta-heurísticas consiste em aplicar o método HORA para obter a sintonização dos parâmetros do GA. No método HORA não há definições *a priori* do número de alternativas utilizadas no processo de sintonização. Isto é, a cada ciclo do método heurístico são criadas m configurações candidatas arbitrárias no espaço de busca (nesta pesquisa, $m = 4$) e, para cada uma delas, a meta-heurística é executada durante 15 segundos sobre um conjunto de treinamento expandido formado por todos os exemplares do *benchmark* wt40.

A qualidade das sintonizações obtidas pelo HORA foi avaliada a partir de comparações com o método de Corrida aplicado sobre os mesmos algoritmos, conjuntos de treinamento e espaços de busca de parâmetros. No método de Corrida, cada parâmetro é definido em um intervalo discreto com número de níveis arbitrários na espaço de busca de parâmetros (Tabela 49). Em síntese, o método de Corrida seleciona a melhor configuração candidata dentre as diversas alternativas criadas *a priori*. O GA é executado por durante 15 segundos com cada configuração candidata sobre o mesmo conjunto de treinamento expandido utilizado na sintonização com método HORA.

As sintonizações da meta-heurística GA para solução do *benchmark* wt40 por meio da metodologia proposta nesta pesquisa são apresentadas na Tabela 50.

Tabela 51 - wt40: Sintonização da meta-heurística GA sob diferentes métodos.

Parâmetros	HORA	Corrida
p_c	$0,699 \pm 0,211$	$0,695 \pm 0,210$
p_m	$0,040 \pm 0,012$	$0,051 \pm 0,010$
μ	80 ± 11	90 ± 13
n	983 ± 115	1087 ± 144
Tempo total	875	11700

APÊNCICE B - Resultados Computacionais - Problema de Sequenciamento de Atividades

B.1 META-HEURÍSTICA SA

Tabela 52 - wt40: Resultados computacionais da meta-heurística SA.

Exemplar	melhor	SA _H				SA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	913	913	3,20	0,00	51	913	6,09	0,00	50	0,00
2	1225	1225	8,21	0,00	62	1263	5,57	3,10	32	-3,01
3	537	573	26,39	6,70	44	573	13,19	6,70	29	0,00
4	2094	2121	1,05	1,29	38	2121	4,19	1,29	36	0,00
5	990	990	0,00	0,00	38	990	4,04	0,00	7	0,00
6	6955	6955	0,06	0,00	86	6955	0,01	0,00	71	0,00
7	6324	6324	1,56	0,00	68	6324	1,92	0,00	32	0,00
8	6865	6865	2,57	0,00	79	6865	0,76	0,00	59	0,00
9	16225	16225	0,84	0,00	81	16225	0,47	0,00	50	0,00
10	9737	9741	1,66	0,04	92	9741	1,85	0,04	85	0,00
11	17465	17525	0,74	0,34	100	17465	0,67	0,00	87	0,34
12	19312	19312	0,40	0,00	114	19312	0,47	0,00	87	0,00
13	29256	29274	0,54	0,06	99	29323	0,69	0,23	87	-0,17
14	14377	14377	1,00	0,00	95	14425	0,80	0,33	87	-0,33
15	26914	26923	0,56	0,03	103	27042	0,44	0,48	83	-0,44
16	72317	72378	0,15	0,08	118	72436	0,14	0,16	121	-0,08
17	78623	78711	0,08	0,11	106	78747	0,04	0,16	144	-0,05
18	74310	74318	0,14	0,01	117	74310	0,06	0,00	106	0,01
19	77122	77314	0,17	0,25	113	77432	0,07	0,40	88	-0,15
20	63229	63285	0,41	0,09	118	63296	0,44	0,11	67	-0,02
21	77774	77774	0,02	0,00	127	77774	0,01	0,00	154	0,00
22	100484	100484	0,04	0,00	118	100484	0,01	0,00	95	0,00
23	135618	135618	0,01	0,00	126	135618	0,01	0,00	142	0,00
24	119947	119947	0,02	0,00	125	119947	0,02	0,00	90	0,00
25	128747	128747	0,01	0,00	124	128747	0,00	0,00	112	0,00
26	108	108	6,67	0,00	33	108	13,33	0,00	10	0,00
27	64	70	12,86	9,38	27	70	0,00	9,38	30	0,00
28	15	15	132,00	0,00	18	15	88,00	0,00	1	0,00
29	47	47	401,28	0,00	37	47	165,11	0,00	34	0,00
30	98	98	64,69	0,00	30	105	48,95	7,14	32	-6,67
31	6575	6575	3,55	0,00	78	6575	0,90	0,00	52	0,00
32	4098	4098	1,36	0,00	72	4098	0,01	0,00	37	0,00
33	5468	5538	3,53	1,28	99	5504	0,94	0,66	79	0,62
34	2648	2648	5,60	0,00	70	2648	2,24	0,00	52	0,00
35	5290	5349	0,42	1,12	69	5340	0,31	0,95	58	0,17
36	19732	19860	1,37	0,65	104	19860	1,45	0,65	86	0,00
37	17349	17349	0,14	0,00	100	17379	1,26	0,17	106	-0,17
38	24499	24563	0,84	0,26	106	24524	1,41	0,10	62	0,16

Tabela 52 - wt40: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	melhor	SA _H				SA _C				
		sol	desvio	gap	t	sol	desvio	gap	t	delta
39	19008	19008	2,36	0,00	115	19053	1,72	0,24	87	-0,24
40	19611	19636	0,26	0,13	109	19799	1,16	0,96	112	-0,82
41	57640	57770	0,23	0,23	120	57640	0,42	0,00	112	0,23
42	81462	81610	0,12	0,18	125	81526	0,05	0,08	134	0,10
43	65134	65159	0,09	0,04	123	65134	0,19	0,00	116	0,04
44	78139	78326	0,25	0,24	122	78231	0,16	0,12	112	0,12
45	66579	66703	0,26	0,19	122	66581	0,43	0,00	150	0,18
46	64451	64451	0,04	0,00	128	64451	0,05	0,00	121	0,00
47	113999	113999	0,03	0,00	122	113999	0,02	0,00	125	0,00
48	74323	74326	0,15	0,00	127	74326	0,03	0,00	101	0,00
49	110295	110299	0,02	0,00	119	110295	0,01	0,00	106	0,00
50	95616	95619	0,01	0,00	124	95616	0,02	0,00	117	0,00
51	0	0	0,00	0,00	22	0	0,00	0,00	5	0,00
52	0	0	0,00	0,00	4	0	0,00	0,00	1	0,00
53	0	0	0,00	0,00	3	0	0,00	0,00	2	0,00
54	0	0	0,00	0,00	3	0	0,00	0,00	1	0,00
55	0	0	0,00	0,00	6	0	9,60	0,00	2	0,00
56	2099	2282	2,50	8,72	63	2282	1,70	8,72	49	0,00
57	2260	2409	15,72	6,59	107	2348	15,58	3,89	57	2,60
58	4936	5180	6,42	4,94	88	5180	3,13	4,94	82	0,00
59	3784	3880	7,98	2,54	71	3882	8,53	2,59	51	-0,05
60	3289	3384	3,40	2,89	105	3289	8,94	0,00	77	2,89
61	20281	20609	1,23	1,62	114	20494	2,50	1,05	95	0,56
62	13403	13572	2,44	1,26	111	13558	1,31	1,16	96	0,10
63	19771	19791	1,18	0,10	116	19786	2,15	0,08	106	0,03
64	24346	24528	2,23	0,75	117	24346	1,54	0,00	106	0,75
65	14905	14905	1,20	0,00	92	14905	1,21	0,00	62	0,00
66	65386	65386	0,51	0,00	111	65386	0,20	0,00	118	0,00
67	65756	65834	0,17	0,12	122	65888	0,16	0,20	112	-0,08
68	78451	78451	0,24	0,00	121	78452	0,18	0,00	137	0,00
69	81627	81688	0,19	0,07	123	81688	0,11	0,07	147	0,00
70	68242	68280	0,09	0,06	125	68249	0,04	0,01	125	0,05
71	90486	90486	0,05	0,00	124	90486	0,12	0,00	103	0,00
72	115249	115249	0,10	0,00	120	115249	0,02	0,00	140	0,00
73	68529	68529	0,01	0,00	127	68529	0,02	0,00	127	0,00
74	79006	79006	0,05	0,00	123	79013	0,15	0,01	132	-0,01
75	98110	98110	0,05	0,00	120	98110	0,07	0,00	130	0,00
76	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
77	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
78	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
79	0	0	0,00	0,00	3	0	0,00	0,00	2	0,00
80	0	0	0,00	0,00	5	0	0,00	0,00	2	0,00
81	684	909	37,78	32,89	49	889	29,18	29,97	52	2,25
82	172	216	86,48	25,58	84	172	247,91	0,00	35	25,58
83	798	876	33,17	9,77	64	932	33,13	16,79	38	-6,01

Tabela 52 - wt40: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	SA _H					SA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
84	617	1418	2,93	129,82	84	1001	68,17	62,24	32	41,66
85	776	865	4,88	11,47	83	844	13,84	8,76	46	2,49
86	10262	10340	2,58	0,76	110	10267	4,58	0,05	90	0,71
87	18646	18745	1,70	0,53	118	18651	1,28	0,03	106	0,50
88	10021	10276	1,75	2,54	115	10102	1,26	0,81	96	1,72
89	25881	26090	0,59	0,81	118	25923	0,75	0,16	112	0,64
90	8159	8336	2,77	2,17	114	8324	2,03	2,02	76	0,14
91	47683	47712	0,72	0,06	116	47760	0,13	0,16	92	-0,10
92	43004	43028	0,31	0,06	120	43138	0,23	0,31	106	-0,25
93	55730	55760	0,19	0,05	115	55752	0,27	0,04	132	0,01
94	59494	59519	0,17	0,04	116	59557	0,19	0,11	97	-0,06
95	42688	42729	0,41	0,10	119	42726	0,70	0,09	132	0,01
96	126048	126056	0,02	0,01	123	126048	0,01	0,00	108	0,01
97	114686	114686	0,01	0,00	125	114686	0,00	0,00	83	0,00
98	112102	112102	0,01	0,00	124	112102	0,00	0,00	110	0,00
99	98206	98206	0,02	0,00	110	98206	0,06	0,00	88	0,00
100	157296	157328	0,01	0,02	113	157300	0,03	0,00	126	0,02
101	0	0	0,00	0,00	2	0	0,00	0,00	2	0,00
102	0	0	0,00	0,00	6	0	0,00	0,00	2	0,00
103	0	0	0,00	0,00	17	0	0,00	0,00	15	0,00
104	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
105	0	0	0,00	0,00	4	0	0,00	0,00	2	0,00
106	0	0	0,00	0,00	30	0	0,00	0,00	12	0,00
107	516	759	26,85	47,09	79	695	12,66	34,69	71	9,21
108	3354	3822	18,83	13,95	111	3494	19,55	4,17	97	9,39
109	0	0	0,00	0,00	19	0	0,00	0,00	4	0,00
110	0	0	0,00	0,00	22	0	0,00	0,00	16	0,00
111	31478	31610	0,55	0,42	123	31478	0,79	0,00	112	0,42
112	21169	21342	0,87	0,82	121	21236	0,70	0,32	104	0,50
113	27077	27113	0,73	0,13	121	27139	0,39	0,23	132	-0,10
114	19648	19765	0,94	0,60	109	19681	0,46	0,17	89	0,43
115	13774	13808	4,99	0,25	120	13808	2,66	0,25	94	0,00
116	46770	46844	0,23	0,16	117	46791	0,17	0,04	137	0,11
117	50364	50462	0,32	0,19	123	50450	0,23	0,17	92	0,02
118	25460	25494	0,36	0,13	123	25479	1,06	0,07	132	0,06
119	66707	66718	0,09	0,02	126	66707	0,07	0,00	109	0,02
120	69019	69042	0,16	0,03	126	69042	0,16	0,03	127	0,00
121	122266	122281	0,03	0,01	126	122370	0,04	0,09	134	-0,07
122	82456	82488	0,06	0,04	117	82456	0,05	0,00	103	0,04
123	75118	75182	0,18	0,09	127	75141	0,19	0,03	138	0,05
124	73041	73041	0,23	0,00	116	73041	0,28	0,00	111	0,00
125	104531	104531	0,06	0,00	122	104531	0,02	0,00	124	0,00
Média			1,23	1,25	68		1,13	1,62	40	-0,30

Tabela 53 - wt50: Resultados computacionais da meta-heurística SA.

Exemplar	melhor	SA _H				SA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	2134	2134	4,43	0,00	76	2134	1,98	0,00	21	0,00
2	1996	2008	0,12	0,60	78	1998	1,03	0,10	61	0,50
3	2583	2583	1,27	0,00	66	2583	3,19	0,00	14	0,00
4	2691	2691	0,00	0,00	72	2691	0,19	0,00	14	0,00
5	1518	1604	0,97	5,67	81	1518	2,50	0,00	77	5,67
6	26276	26279	0,64	0,01	180	26299	0,56	0,09	106	-0,08
7	11403	11403	0,59	0,00	160	11403	1,94	0,00	95	0,00
8	8499	8544	1,72	0,53	136	8544	0,98	0,53	56	0,00
9	9884	9884	0,28	0,00	120	9918	0,36	0,34	51	-0,34
10	10655	10655	0,70	0,00	153	10655	1,29	0,00	89	0,00
11	43504	43504	0,65	0,00	201	43506	0,63	0,00	173	0,00
12	36378	36461	0,21	0,23	184	36461	0,82	0,23	172	0,00
13	45383	45383	0,50	0,00	200	45501	0,45	0,26	117	-0,26
14	51785	51880	0,33	0,18	168	51845	0,34	0,12	129	0,07
15	38934	38934	0,41	0,00	192	38934	0,07	0,00	151	0,00
16	87902	87906	0,17	0,00	204	87902	0,12	0,00	175	0,00
17	84260	84375	0,07	0,14	204	84278	0,12	0,02	175	0,12
18	104795	104926	0,25	0,13	206	104804	0,32	0,01	175	0,12
19	89299	89474	0,15	0,20	194	89299	0,30	0,00	160	0,20
20	72316	72316	0,27	0,00	195	72329	0,33	0,02	175	-0,02
21	214546	214547	0,01	0,00	202	214561	0,02	0,01	182	-0,01
22	150800	150801	0,01	0,00	203	150808	0,03	0,01	181	0,00
23	224025	224025	0,01	0,00	203	224026	0,01	0,00	181	0,00
24	116015	116018	0,04	0,00	205	116052	0,02	0,03	181	-0,03
25	240179	240181	0,01	0,00	205	240179	0,04	0,00	180	0,00
26	2	2	1840,00	0,00	77	10	816,00	400,00	43	-80,00
27	4	4	0,00	0,00	39	4	0,00	0,00	3	0,00
28	755	879	14,56	16,42	86	819	3,86	8,48	64	7,33
29	99	104	68,08	5,05	70	104	70,96	5,05	51	0,00
30	22	22	0,00	0,00	23	22	0,00	0,00	9	0,00
31	9934	10015	2,87	0,82	122	9988	1,99	0,54	105	0,27
32	7178	7270	3,31	1,28	133	7270	1,81	1,28	60	0,00
33	4674	4674	2,51	0,00	116	4674	3,14	0,00	48	0,00
34	4017	4035	5,80	0,45	132	4035	5,52	0,45	59	0,00
35	6459	6710	2,11	3,89	175	6611	4,11	2,35	94	1,50
36	34892	35002	2,49	0,32	183	34950	1,18	0,17	109	0,15
37	22739	22740	0,24	0,00	160	22740	1,99	0,00	117	0,00
38	29467	29492	1,18	0,08	175	29492	0,76	0,08	107	0,00
39	49352	49436	0,45	0,17	196	49431	0,26	0,16	137	0,01
40	26423	26485	0,63	0,23	192	26430	2,04	0,03	85	0,21
41	71111	71124	0,29	0,02	214	71118	0,50	0,01	178	0,01
42	90163	90210	0,16	0,05	207	90205	0,16	0,05	141	0,01
43	84126	84126	0,43	0,00	213	84361	0,43	0,28	139	-0,28
44	123893	124074	0,15	0,15	211	123991	0,12	0,08	151	0,07
45	79883	79981	0,19	0,12	213	79916	0,10	0,04	178	0,08

Tabela 53 - wt50: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	melhor	SA _H				SA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
46	157505	157520	0,03	0,01	217	157508	0,04	0,00	181	0,01
47	133289	133295	0,03	0,00	219	133289	0,01	0,00	182	0,00
48	191099	191099	0,03	0,00	215	191102	0,01	0,00	182	0,00
49	150279	150289	0,01	0,01	210	150288	0,03	0,01	180	0,00
50	198076	198078	0,00	0,00	215	198076	0,03	0,00	173	0,00
51	0	0	0,00	0,00	12	0	0,00	0,00	3	0,00
52	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
53	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
54	0	0	0,00	0,00	19	0	0,00	0,00	5	0,00
55	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
56	1258	1318	2,90	4,77	126	1318	3,55	4,77	54	0,00
57	3679	3955	2,19	7,50	165	3952	6,44	7,42	109	0,08
58	2522	2522	9,83	0,00	90	2522	4,94	0,00	56	0,00
59	3770	4162	6,74	10,40	184	4035	11,01	7,03	114	3,15
60	5904	6052	7,16	2,51	139	5926	10,95	0,37	75	2,13
61	25212	25666	1,92	1,80	200	25633	2,83	1,67	119	0,13
62	17337	17648	3,07	1,79	200	17817	3,84	2,77	172	-0,95
63	30729	30860	1,68	0,43	178	31117	1,20	1,26	172	-0,83
64	18082	18594	1,10	2,83	198	18189	2,44	0,59	170	2,23
65	25028	25560	1,33	2,13	188	25490	1,06	1,85	142	0,27
66	76878	76956	0,15	0,10	199	76974	0,14	0,12	174	-0,02
67	85413	85873	0,12	0,54	207	85523	0,56	0,13	172	0,41
68	92756	92868	0,20	0,12	207	92802	0,33	0,05	146	0,07
69	77930	77937	0,26	0,01	193	77966	0,25	0,05	179	-0,04
70	74750	74904	0,35	0,21	207	74889	0,25	0,19	178	0,02
71	150580	150586	0,07	0,00	212	150580	0,06	0,00	180	0,00
72	131680	131703	0,03	0,02	205	131698	0,07	0,01	179	0,00
73	98494	98563	0,06	0,07	203	98593	0,07	0,10	180	-0,03
74	135394	135471	0,02	0,06	204	135486	0,04	0,07	181	-0,01
75	135677	135744	0,08	0,05	203	135744	0,05	0,05	182	0,00
76	0	0	0,00	0,00	7	0	0,00	0,00	3	0,00
77	0	0	0,00	0,00	10	0	0,00	0,00	4	0,00
78	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
79	0	0	0,00	0,00	4	0	0,00	0,00	2	0,00
80	0	0	0,00	0,00	7	0	0,00	0,00	3	0,00
81	816	1228	7,54	50,49	131	1228	6,53	50,49	70	0,00
82	4879	5399	3,26	10,66	167	5040	10,73	3,30	151	7,12
83	973	1280	9,97	31,55	142	1306	20,09	34,22	163	-1,99
84	508	703	31,27	38,39	119	561	15,86	10,43	86	25,31
85	3780	3796	6,00	0,42	121	4006	14,16	5,98	85	-5,24
86	20751	20874	1,26	0,59	196	20859	2,97	0,52	173	0,07
87	36053	36211	0,73	0,44	199	36368	1,02	0,87	176	-0,43
88	28268	28443	1,17	0,62	195	28536	0,51	0,95	173	-0,33
89	28846	29177	1,95	1,15	194	29332	1,26	1,68	175	-0,53
90	15451	15785	1,06	2,16	189	15737	0,65	1,85	139	0,31

Tabela 53 - wt50: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	melhor	SA _H				SA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
91	89298	89482	0,19	0,21	200	89477	0,18	0,20	177	0,01
92	66340	66403	0,45	0,09	195	66527	0,42	0,28	176	-0,19
93	61060	61144	0,31	0,14	198	61071	0,67	0,02	177	0,12
94	42453	42610	0,32	0,37	198	42718	0,57	0,62	175	-0,25
95	56522	56724	0,83	0,36	200	56812	0,86	0,51	158	-0,15
96	177909	177909	0,04	0,00	204	177968	0,05	0,03	180	-0,03
97	139591	139643	0,07	0,04	203	139598	0,04	0,01	180	0,03
98	148906	148927	0,18	0,01	199	149017	0,23	0,07	178	-0,06
99	179264	179281	0,05	0,01	203	179268	0,09	0,00	180	0,01
100	120108	120171	0,12	0,05	202	120149	0,08	0,03	178	0,02
101	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
102	0	0	0,00	0,00	7	0	0,00	0,00	4	0,00
103	0	0	0,00	0,00	3	0	0,00	0,00	1	0,00
104	0	0	0,00	0,00	9	0	0,00	0,00	4	0,00
105	0	0	57,60	0,00	81	0	105,80	0,00	60	0,00
106	0	0	0,00	0,00	44	0	0,00	0,00	17	0,00
107	1717	1962	11,14	14,27	182	1940	15,01	12,99	65	1,13
108	0	0	6,00	0,00	66	0	0,00	0,00	30	0,00
109	6185	6447	4,11	4,24	185	6703	2,54	8,38	144	-3,82
110	1295	1359	11,67	4,94	150	1522	5,52	17,53	66	-10,71
111	27310	27569	0,72	0,95	186	27367	1,69	0,21	173	0,74
112	15867	15867	1,26	0,00	180	15912	3,62	0,28	96	-0,28
113	35106	35849	1,03	2,12	194	36014	1,60	2,59	142	-0,46
114	15467	15964	2,90	3,21	175	16078	2,36	3,95	163	-0,71
115	10574	11402	1,44	7,83	169	11150	2,17	5,45	97	2,26
116	35727	36087	0,70	1,01	192	36008	1,51	0,79	174	0,22
117	71922	71964	0,23	0,06	196	71951	0,24	0,04	179	0,02
118	65433	65632	0,48	0,30	194	65547	0,33	0,17	177	0,13
119	106043	106097	0,14	0,05	190	106106	0,05	0,06	179	-0,01
120	101665	101755	0,23	0,09	183	101828	0,10	0,16	179	-0,07
121	78315	78415	0,13	0,13	197	78391	0,23	0,10	172	0,03
122	119925	119930	0,09	0,00	198	120239	0,16	0,26	180	-0,26
123	101157	101170	0,26	0,01	196	101181	0,16	0,02	179	-0,01
124	139488	139594	0,08	0,08	197	139571	0,10	0,06	181	0,02
125	110392	110426	0,12	0,03	196	110477	0,15	0,08	179	-0,05
Média			1,58	0,81	78		1,24	1,06	57	-0,21

Tabela 54 - wt100: Resultados computacionais da meta-heurística SA.

Exemplar	melhor	SA _H				SA _C				
		sol	desvio	gap	t	sol	desvio	gap	t	delta
1	5988	6077	1,79	1,49	126	6031	1,69	0,72	105	0,76
2	6170	6240	0,96	1,13	147	6182	5,78	0,19	86	0,94
3	4267	4308	3,00	0,96	135	4297	3,35	0,70	97	0,26
4	5011	5011	5,35	0,00	128	5011	0,90	0,00	74	0,00
5	5283	5283	8,35	0,00	112	5283	2,22	0,00	75	0,00
6	58258	58665	1,25	0,70	148	58680	0,82	0,72	77	-0,03
7	50972	51252	1,07	0,55	153	52085	0,42	2,18	116	-1,60
8	59434	59774	0,48	0,57	137	59631	1,11	0,33	140	0,24
9	40978	41095	1,36	0,29	152	41087	1,05	0,27	113	0,02
10	53208	53576	1,27	0,69	154	53714	0,94	0,95	128	-0,26
11	181649	183122	1,03	0,81	146	184601	0,86	1,63	145	-0,80
12	234179	237757	0,45	1,53	156	236352	0,49	0,93	146	0,59
13	178840	181390	0,62	1,43	156	182088	0,22	1,82	146	-0,38
14	157476	159400	0,53	1,22	158	158689	0,68	0,77	146	0,45
15	172995	174589	0,64	0,92	157	175198	0,42	1,27	130	-0,35
16	407703	412983	0,25	1,30	157	414111	0,62	1,57	116	-0,27
17	332804	336919	0,29	1,24	161	336566	0,41	1,13	134	0,10
18	544838	550603	0,40	1,06	148	550470	0,12	1,03	130	0,02
19	477684	482930	0,18	1,10	163	480729	0,42	0,64	106	0,46
20	406094	411800	0,31	1,41	146	410582	0,38	1,11	149	0,30
21	898925	901134	0,14	0,25	165	902650	0,08	0,41	68	-0,17
22	556873	561284	0,13	0,79	151	560081	0,24	0,58	68	0,21
23	539716	543783	0,36	0,75	134	542460	0,27	0,51	100	0,24
24	744287	746791	0,13	0,34	152	746670	0,17	0,32	123	0,02
25	585306	590511	0,10	0,89	171	588715	0,32	0,58	157	0,31
26	8	8	1880,00	0,00	108	8	1400,00	0,00	42	0,00
27	718	848	11,89	18,11	139	929	4,09	29,39	81	-8,72
28	27	27	500,74	0,00	92	27	362,22	0,00	58	0,00
29	480	554	16,61	15,42	124	482	40,08	0,42	107	14,94
30	50	50	0,00	0,00	102	50	36,40	0,00	104	0,00
31	24202	24814	2,80	2,53	148	24538	2,65	1,39	125	1,12
32	25469	27049	3,23	6,20	154	25868	7,59	1,57	142	4,57
33	32964	33930	3,39	2,93	155	34981	3,56	6,12	142	-3,00
34	22215	24418	1,43	9,92	145	24252	3,86	9,17	142	0,68
35	19114	20124	3,47	5,28	149	20903	2,66	9,36	140	-3,73
36	108293	111707	1,69	3,15	157	111176	2,06	2,66	139	0,48
37	181850	186463	1,13	2,54	157	184626	1,84	1,53	127	0,99
38	90440	94754	1,41	4,77	157	97093	0,75	7,36	105	-2,41
39	151701	156375	1,37	3,08	150	157511	0,86	3,83	144	-0,72
40	129728	132206	1,59	1,91	158	135466	2,02	4,42	145	-2,41
41	462324	465037	0,66	0,59	165	467908	0,11	1,21	152	-0,61
42	425875	429238	0,34	0,79	158	428976	0,34	0,73	141	0,06
43	320537	324913	0,51	1,37	160	324473	1,01	1,23	81	0,14
44	360193	365474	0,50	1,47	158	365793	0,14	1,55	150	-0,09
45	306040	310929	0,79	1,60	142	310664	0,74	1,51	150	0,09

Tabela 54 - wt100: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	SA _H					SA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
46	829828	832571	0,14	0,33	174	833091	0,18	0,39	156	-0,06
47	623356	625416	0,34	0,33	160	625315	0,20	0,31	79	0,02
48	748988	753891	0,15	0,65	171	755146	0,12	0,82	79	-0,17
49	656693	662142	0,25	0,83	150	659328	0,16	0,40	119	0,43
50	599269	601686	0,60	0,40	155	603314	0,21	0,67	155	-0,27
51	0	0	0,00	0,00	32	0	0,00	0,00	8	0,00
52	0	0	0,00	0,00	28	0	0,00	0,00	11	0,00
53	0	0	0,00	0,00	15	0	0,00	0,00	8	0,00
54	0	0	0,00	0,00	60	0	0,00	0,00	19	0,00
55	0	0	0,00	0,00	44	0	0,00	0,00	20	0,00
56	9046	10909	6,11	20,59	143	9870	16,23	9,11	102	10,53
57	11539	11792	6,64	2,19	150	12328	8,31	6,84	115	-4,35
58	16313	19418	10,27	19,03	156	19954	10,29	22,32	139	-2,69
59	7965	8453	6,42	6,13	155	8156	11,37	2,40	139	3,64
60	19912	22471	7,36	12,85	154	21710	4,99	9,03	108	3,51
61	86793	95862	1,98	10,45	154	94894	1,64	9,33	144	1,02
62	87067	94797	2,49	8,88	164	94081	3,23	8,06	103	0,76
63	96563	104511	4,08	8,23	164	105863	2,97	9,63	108	-1,28
64	100788	104485	1,91	3,67	156	104231	1,33	3,42	142	0,24
65	56510	61347	3,58	8,56	160	60779	2,83	7,55	118	0,93
66	243872	249084	0,71	2,14	144	249185	0,62	2,18	139	-0,04
67	401023	407373	0,30	1,58	154	407410	0,74	1,59	99	-0,01
68	399085	405708	0,66	1,66	173	406478	0,68	1,85	153	-0,19
69	309232	315875	1,01	2,15	167	314551	1,28	1,72	152	0,42
70	222684	229875	0,95	3,23	150	228458	1,59	2,59	113	0,62
71	640816	645401	0,13	0,72	178	643547	0,23	0,43	158	0,29
72	611362	615766	0,22	0,72	174	615209	0,43	0,63	158	0,09
73	623429	626105	0,41	0,43	179	628201	0,17	0,77	158	-0,33
74	584628	590730	0,20	1,04	167	590174	0,20	0,95	156	0,09
75	575274	578222	0,28	0,51	174	577505	0,52	0,39	155	0,12
76	0	0	0,00	0,00	51	0	0,00	0,00	18	0,00
77	0	0	0,00	0,00	27	0	0,00	0,00	8	0,00
78	0	0	0,00	0,00	30	0	0,00	0,00	19	0,00
79	0	0	0,00	0,00	34	0	0,00	0,00	26	0,00
80	0	0	0,00	0,00	36	0	0,00	0,00	20	0,00
81	1400	3190	50,23	127,86	148	5452	12,67	289,43	139	-41,49
82	317	2932	31,93	824,92	150	1941	56,20	512,30	137	51,06
83	1146	3208	41,74	179,93	154	5018	14,27	337,87	59	-36,07
84	136	356	63,37	161,76	133	163	401,47	19,85	109	118,40
85	284	3870	15,06	1262,68	126	3696	20,77	1201,41	128	4,71
86	66850	78123	2,69	16,86	161	79716	6,73	19,25	140	-2,00
87	84229	94251	4,81	11,90	162	95484	5,66	13,36	146	-1,29
88	55544	62811	6,26	13,08	152	64420	5,37	15,98	72	-2,50
89	54612	58791	5,51	7,65	147	61206	3,46	12,07	143	-3,95
90	75061	83475	3,36	11,21	165	82910	2,13	10,46	109	0,68

Tabela 54 - wt100: Resultados computacionais da meta-heurística SA. (cont.)

Exemplar	SA _H					SA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
91	248699	256721	1,10	3,23	162	252294	1,90	1,45	98	1,75
92	311022	317294	1,03	2,02	167	316408	1,13	1,73	153	0,28
93	326258	333987	0,17	2,37	165	333692	0,60	2,28	115	0,09
94	273993	281098	0,80	2,59	152	279889	1,32	2,15	101	0,43
95	316870	324111	1,06	2,29	159	323125	0,80	1,97	126	0,31
96	495516	498824	0,30	0,67	161	498420	0,37	0,59	140	0,08
97	636903	641094	0,31	0,66	174	641596	0,36	0,74	156	-0,08
98	680082	683301	0,14	0,47	147	681866	0,14	0,26	157	0,21
99	622464	625872	0,40	0,55	171	626213	0,26	0,60	157	-0,05
100	449545	452730	0,49	0,71	175	454587	0,20	1,12	154	-0,41
101	0	0	0,00	0,00	36	0	0,00	0,00	35	0,00
102	0	0	0,00	0,00	13	0	0,00	0,00	11	0,00
103	0	0	0,00	0,00	33	0	0,00	0,00	13	0,00
104	0	0	0,00	0,00	24	0	0,00	0,00	21	0,00
105	0	0	0,00	0,00	16	0	0,00	0,00	7	0,00
106	0	0	5,60	0,00	119	0	0,00	0,00	135	0,00
107	1193	6944	14,11	482,06	142	5654	25,31	373,93	139	22,82
108	0	1748	31,29	1748,00	152	1466	63,30	1466,00	111	19,24
109	232	1317	210,86	467,67	142	1740	41,86	650,00	137	-24,31
110	0	3210	48,56	3210,00	154	1200	205,57	1200,00	139	167,50
111	159123	172560	0,74	8,44	167	170537	1,12	7,17	150	1,19
112	174367	180986	1,87	3,80	167	184999	1,10	6,10	94	-2,17
113	91169	100933	1,73	10,71	166	101023	2,45	10,81	146	-0,09
114	168266	176877	1,53	5,12	173	178227	1,18	5,92	148	-0,76
115	70190	79658	2,37	13,49	170	80494	3,60	14,68	102	-1,04
116	370614	375575	1,11	1,34	166	375835	0,39	1,41	153	-0,07
117	324437	331716	0,78	2,24	176	330957	0,88	2,01	149	0,23
118	246237	254542	0,52	3,37	165	254630	0,97	3,41	151	-0,03
119	293571	299030	1,61	1,86	159	302550	0,48	3,06	154	-1,16
120	267316	273156	0,72	2,18	177	273101	0,79	2,16	151	0,02
121	471214	478896	0,27	1,63	172	479694	0,19	1,80	156	-0,17
122	570459	573861	0,31	0,60	178	573946	0,19	0,61	118	-0,01
123	397029	401133	0,67	1,03	164	402801	0,54	1,45	93	-0,41
124	431115	434581	0,43	0,80	177	435712	0,38	1,07	122	-0,26
125	560754	567964	0,36	1,29	179	568686	0,51	1,41	124	-0,13
Média			10,37	1,99	197		9,47	3,57	155	-0,91

B.2 META-HEURÍSTICA GA

Tabela 55 - wt40: Resultados computacionais da meta-heurística GA.

Exemplar	melhor	GA _H				GA _C				
		sol	desvio	gap	t	sol	desvio	gap	t	delta
1	913	913	3,20	0,00	51	913	6,09	0,00	50	0,00
2	1225	1225	8,21	0,00	62	1263	5,57	3,10	32	-3,01
3	537	573	26,39	6,70	44	573	13,19	6,70	29	0,00
4	2094	2121	1,05	1,29	38	2121	4,19	1,29	36	0,00
5	990	990	0,00	0,00	38	990	4,04	0,00	7	0,00
6	6955	6955	0,06	0,00	86	6955	0,01	0,00	71	0,00
7	6324	6324	1,56	0,00	68	6324	1,92	0,00	32	0,00
8	6865	6865	2,57	0,00	79	6865	0,76	0,00	59	0,00
9	16225	16225	0,84	0,00	81	16225	0,47	0,00	50	0,00
10	9737	9741	1,66	0,04	92	9741	1,85	0,04	85	0,00
11	17465	17525	0,74	0,34	100	17465	0,67	0,00	87	0,34
12	19312	19312	0,40	0,00	114	19312	0,47	0,00	87	0,00
13	29256	29274	0,54	0,06	99	29323	0,69	0,23	87	-0,17
14	14377	14377	1,00	0,00	95	14425	0,80	0,33	87	-0,33
15	26914	26923	0,56	0,03	103	27042	0,44	0,48	83	-0,44
16	72317	72378	0,15	0,08	118	72436	0,14	0,16	121	-0,08
17	78623	78711	0,08	0,11	106	78747	0,04	0,16	144	-0,05
18	74310	74318	0,14	0,01	117	74310	0,06	0,00	106	0,01
19	77122	77314	0,17	0,25	113	77432	0,07	0,40	88	-0,15
20	63229	63285	0,41	0,09	118	63296	0,44	0,11	67	-0,02
21	77774	77774	0,02	0,00	127	77774	0,01	0,00	154	0,00
22	100484	100484	0,04	0,00	118	100484	0,01	0,00	95	0,00
23	135618	135618	0,01	0,00	126	135618	0,01	0,00	142	0,00
24	119947	119947	0,02	0,00	125	119947	0,02	0,00	90	0,00
25	128747	128747	0,01	0,00	124	128747	0,00	0,00	112	0,00
26	108	108	6,67	0,00	33	108	13,33	0,00	10	0,00
27	64	70	12,86	9,38	27	70	0,00	9,38	30	0,00
28	15	15	132,00	0,00	18	15	88,00	0,00	1	0,00
29	47	47	401,28	0,00	37	47	165,11	0,00	34	0,00
30	98	98	64,69	0,00	30	105	48,95	7,14	32	-6,67
31	6575	6575	3,55	0,00	78	6575	0,90	0,00	52	0,00
32	4098	4098	1,36	0,00	72	4098	0,01	0,00	37	0,00
33	5468	5538	3,53	1,28	99	5504	0,94	0,66	79	0,62
34	2648	2648	5,60	0,00	70	2648	2,24	0,00	52	0,00
35	5290	5349	0,42	1,12	69	5340	0,31	0,95	58	0,17
36	19732	19860	1,37	0,65	104	19860	1,45	0,65	86	0,00
37	17349	17349	0,14	0,00	100	17379	1,26	0,17	106	-0,17
38	24499	24563	0,84	0,26	106	24524	1,41	0,10	62	0,16
39	19008	19008	2,36	0,00	115	19053	1,72	0,24	87	-0,24
40	19611	19636	0,26	0,13	109	19799	1,16	0,96	112	-0,82
41	57640	57770	0,23	0,23	120	57640	0,42	0,00	112	0,23
42	81462	81610	0,12	0,18	125	81526	0,05	0,08	134	0,10
43	65134	65159	0,09	0,04	123	65134	0,19	0,00	116	0,04
44	78139	78326	0,25	0,24	122	78231	0,16	0,12	112	0,12

Tabela 55 - wt40: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	GA _H					GA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
45	66579	66703	0,26	0,19	122	66581	0,43	0,00	150	0,18
46	64451	64451	0,04	0,00	128	64451	0,05	0,00	121	0,00
47	113999	113999	0,03	0,00	122	113999	0,02	0,00	125	0,00
48	74323	74326	0,15	0,00	127	74326	0,03	0,00	101	0,00
49	110295	110299	0,02	0,00	119	110295	0,01	0,00	106	0,00
50	95616	95619	0,01	0,00	124	95616	0,02	0,00	117	0,00
51	0	0	0,00	0,00	22	0	0,00	0,00	5	0,00
52	0	0	0,00	0,00	4	0	0,00	0,00	1	0,00
53	0	0	0,00	0,00	3	0	0,00	0,00	2	0,00
54	0	0	0,00	0,00	3	0	0,00	0,00	1	0,00
55	0	0	0,00	0,00	6	0	9,60	0,00	2	0,00
56	2099	2282	2,50	8,72	63	2282	1,70	8,72	49	0,00
57	2260	2409	15,72	6,59	107	2348	15,58	3,89	57	2,60
58	4936	5180	6,42	4,94	88	5180	3,13	4,94	82	0,00
59	3784	3880	7,98	2,54	71	3882	8,53	2,59	51	-0,05
60	3289	3384	3,40	2,89	105	3289	8,94	0,00	77	2,89
61	20281	20609	1,23	1,62	114	20494	2,50	1,05	95	0,56
62	13403	13572	2,44	1,26	111	13558	1,31	1,16	96	0,10
63	19771	19791	1,18	0,10	116	19786	2,15	0,08	106	0,03
64	24346	24528	2,23	0,75	117	24346	1,54	0,00	106	0,75
65	14905	14905	1,20	0,00	92	14905	1,21	0,00	62	0,00
66	65386	65386	0,51	0,00	111	65386	0,20	0,00	118	0,00
67	65756	65834	0,17	0,12	122	65888	0,16	0,20	112	-0,08
68	78451	78451	0,24	0,00	121	78452	0,18	0,00	137	0,00
69	81627	81688	0,19	0,07	123	81688	0,11	0,07	147	0,00
70	68242	68280	0,09	0,06	125	68249	0,04	0,01	125	0,05
71	90486	90486	0,05	0,00	124	90486	0,12	0,00	103	0,00
72	115249	115249	0,10	0,00	120	115249	0,02	0,00	140	0,00
73	68529	68529	0,01	0,00	127	68529	0,02	0,00	127	0,00
74	79006	79006	0,05	0,00	123	79013	0,15	0,01	132	-0,01
75	98110	98110	0,05	0,00	120	98110	0,07	0,00	130	0,00
76	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
77	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
78	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
79	0	0	0,00	0,00	3	0	0,00	0,00	2	0,00
80	0	0	0,00	0,00	5	0	0,00	0,00	2	0,00
81	684	909	37,78	32,89	49	889	29,18	29,97	52	2,25
82	172	216	86,48	25,58	84	172	247,91	0,00	35	25,58
83	798	876	33,17	9,77	64	932	33,13	16,79	38	-6,01
84	617	1418	2,93	129,82	84	1001	68,17	62,24	32	41,66
85	776	865	4,88	11,47	83	844	13,84	8,76	46	2,49
86	10262	10340	2,58	0,76	110	10267	4,58	0,05	90	0,71
87	18646	18745	1,70	0,53	118	18651	1,28	0,03	106	0,50
88	10021	10276	1,75	2,54	115	10102	1,26	0,81	96	1,72
89	25881	26090	0,59	0,81	118	25923	0,75	0,16	112	0,64

Tabela 55 - wt40: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	GA _H					GA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
90	8159	8336	2,77	2,17	114	8324	2,03	2,02	76	0,14
91	47683	47712	0,72	0,06	116	47760	0,13	0,16	92	-0,10
92	43004	43028	0,31	0,06	120	43138	0,23	0,31	106	-0,25
93	55730	55760	0,19	0,05	115	55752	0,27	0,04	132	0,01
94	59494	59519	0,17	0,04	116	59557	0,19	0,11	97	-0,06
95	42688	42729	0,41	0,10	119	42726	0,70	0,09	132	0,01
96	126048	126056	0,02	0,01	123	126048	0,01	0,00	108	0,01
97	114686	114686	0,01	0,00	125	114686	0,00	0,00	83	0,00
98	112102	112102	0,01	0,00	124	112102	0,00	0,00	110	0,00
99	98206	98206	0,02	0,00	110	98206	0,06	0,00	88	0,00
100	157296	157328	0,01	0,02	113	157300	0,03	0,00	126	0,02
101	0	0	0,00	0,00	2	0	0,00	0,00	2	0,00
102	0	0	0,00	0,00	6	0	0,00	0,00	2	0,00
103	0	0	0,00	0,00	17	0	0,00	0,00	15	0,00
104	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
105	0	0	0,00	0,00	4	0	0,00	0,00	2	0,00
106	0	0	0,00	0,00	30	0	0,00	0,00	12	0,00
107	516	759	26,85	47,09	79	695	12,66	34,69	71	9,21
108	3354	3822	18,83	13,95	111	3494	19,55	4,17	97	9,39
109	0	0	0,00	0,00	19	0	0,00	0,00	4	0,00
110	0	0	0,00	0,00	22	0	0,00	0,00	16	0,00
111	31478	31610	0,55	0,42	123	31478	0,79	0,00	112	0,42
112	21169	21342	0,87	0,82	121	21236	0,70	0,32	104	0,50
113	27077	27113	0,73	0,13	121	27139	0,39	0,23	132	-0,10
114	19648	19765	0,94	0,60	109	19681	0,46	0,17	89	0,43
115	13774	13808	4,99	0,25	120	13808	2,66	0,25	94	0,00
116	46770	46844	0,23	0,16	117	46791	0,17	0,04	137	0,11
117	50364	50462	0,32	0,19	123	50450	0,23	0,17	92	0,02
118	25460	25494	0,36	0,13	123	25479	1,06	0,07	132	0,06
119	66707	66718	0,09	0,02	126	66707	0,07	0,00	109	0,02
120	69019	69042	0,16	0,03	126	69042	0,16	0,03	127	0,00
121	122266	122281	0,03	0,01	126	122370	0,04	0,09	134	-0,07
122	82456	82488	0,06	0,04	117	82456	0,05	0,00	103	0,04
123	75118	75182	0,18	0,09	127	75141	0,19	0,03	138	0,05
124	73041	73041	0,23	0,00	116	73041	0,28	0,00	111	0,00
125	104531	104531	0,06	0,00	122	104531	0,02	0,00	124	0,00
Média			7,73	2,66	89		6,96	1,74	79	0,69

Tabela 56 - wt50: Resultados computacionais da meta-heurística GA.

Exemplar	melhor	GA _H				GA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	2134	2134	4,43	0,00	76	2134	1,98	0,00	21	0,00
2	1996	2008	0,12	0,60	78	1998	1,03	0,10	61	0,50
3	2583	2583	1,27	0,00	66	2583	3,19	0,00	14	0,00
4	2691	2691	0,00	0,00	72	2691	0,19	0,00	14	0,00
5	1518	1604	0,97	5,67	81	1518	2,50	0,00	77	5,67
6	26276	26279	0,64	0,01	180	26299	0,56	0,09	106	-0,08
7	11403	11403	0,59	0,00	160	11403	1,94	0,00	95	0,00
8	8499	8544	1,72	0,53	136	8544	0,98	0,53	56	0,00
9	9884	9884	0,28	0,00	120	9918	0,36	0,34	51	-0,34
10	10655	10655	0,70	0,00	153	10655	1,29	0,00	89	0,00
11	43504	43504	0,65	0,00	201	43506	0,63	0,00	173	0,00
12	36378	36461	0,21	0,23	184	36461	0,82	0,23	172	0,00
13	45383	45383	0,50	0,00	200	45501	0,45	0,26	117	-0,26
14	51785	51880	0,33	0,18	168	51845	0,34	0,12	129	0,07
15	38934	38934	0,41	0,00	192	38934	0,07	0,00	151	0,00
16	87902	87906	0,17	0,00	204	87902	0,12	0,00	175	0,00
17	84260	84375	0,07	0,14	204	84278	0,12	0,02	175	0,12
18	104795	104926	0,25	0,13	206	104804	0,32	0,01	175	0,12
19	89299	89474	0,15	0,20	194	89299	0,30	0,00	160	0,20
20	72316	72316	0,27	0,00	195	72329	0,33	0,02	175	-0,02
21	214546	214547	0,01	0,00	202	214561	0,02	0,01	182	-0,01
22	150800	150801	0,01	0,00	203	150808	0,03	0,01	181	0,00
23	224025	224025	0,01	0,00	203	224026	0,01	0,00	181	0,00
24	116015	116018	0,04	0,00	205	116052	0,02	0,03	181	-0,03
25	240179	240181	0,01	0,00	205	240179	0,04	0,00	180	0,00
26	2	2	1840,00	0,00	77	10	816,00	400,00	43	-80,00
27	4	4	0,00	0,00	39	4	0,00	0,00	3	0,00
28	755	879	14,56	16,42	86	819	3,86	8,48	64	7,33
29	99	104	68,08	5,05	70	104	70,96	5,05	51	0,00
30	22	22	0,00	0,00	23	22	0,00	0,00	9	0,00
31	9934	10015	2,87	0,82	122	9988	1,99	0,54	105	0,27
32	7178	7270	3,31	1,28	133	7270	1,81	1,28	60	0,00
33	4674	4674	2,51	0,00	116	4674	3,14	0,00	48	0,00
34	4017	4035	5,80	0,45	132	4035	5,52	0,45	59	0,00
35	6459	6710	2,11	3,89	175	6611	4,11	2,35	94	1,50
36	34892	35002	2,49	0,32	183	34950	1,18	0,17	109	0,15
37	22739	22740	0,24	0,00	160	22740	1,99	0,00	117	0,00
38	29467	29492	1,18	0,08	175	29492	0,76	0,08	107	0,00
39	49352	49436	0,45	0,17	196	49431	0,26	0,16	137	0,01
40	26423	26485	0,63	0,23	192	26430	2,04	0,03	85	0,21
41	71111	71124	0,29	0,02	214	71118	0,50	0,01	178	0,01
42	90163	90210	0,16	0,05	207	90205	0,16	0,05	141	0,01
43	84126	84126	0,43	0,00	213	84361	0,43	0,28	139	-0,28
44	123893	124074	0,15	0,15	211	123991	0,12	0,08	151	0,07
45	79883	79981	0,19	0,12	213	79916	0,10	0,04	178	0,08
46	157505	157520	0,03	0,01	217	157508	0,04	0,00	181	0,01
47	133289	133295	0,03	0,00	219	133289	0,01	0,00	182	0,00

Tabela 56 - wt50: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	melhor	GA _H				GA _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
48	191099	191099	0,03	0,00	215	191102	0,01	0,00	182	0,00
49	150279	150289	0,01	0,01	210	150288	0,03	0,01	180	0,00
50	198076	198078	0,00	0,00	215	198076	0,03	0,00	173	0,00
51	0	0	0,00	0,00	12	0	0,00	0,00	3	0,00
52	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
53	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
54	0	0	0,00	0,00	19	0	0,00	0,00	5	0,00
55	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
56	1258	1318	2,90	4,77	126	1318	3,55	4,77	54	0,00
57	3679	3955	2,19	7,50	165	3952	6,44	7,42	109	0,08
58	2522	2522	9,83	0,00	90	2522	4,94	0,00	56	0,00
59	3770	4162	6,74	10,40	184	4035	11,01	7,03	114	3,15
60	5904	6052	7,16	2,51	139	5926	10,95	0,37	75	2,13
61	25212	25666	1,92	1,80	200	25633	2,83	1,67	119	0,13
62	17337	17648	3,07	1,79	200	17817	3,84	2,77	172	-0,95
63	30729	30860	1,68	0,43	178	31117	1,20	1,26	172	-0,83
64	18082	18594	1,10	2,83	198	18189	2,44	0,59	170	2,23
65	25028	25560	1,33	2,13	188	25490	1,06	1,85	142	0,27
66	76878	76956	0,15	0,10	199	76974	0,14	0,12	174	-0,02
67	85413	85873	0,12	0,54	207	85523	0,56	0,13	172	0,41
68	92756	92868	0,20	0,12	207	92802	0,33	0,05	146	0,07
69	77930	77937	0,26	0,01	193	77966	0,25	0,05	179	-0,04
70	74750	74904	0,35	0,21	207	74889	0,25	0,19	178	0,02
71	150580	150586	0,07	0,00	212	150580	0,06	0,00	180	0,00
72	131680	131703	0,03	0,02	205	131698	0,07	0,01	179	0,00
73	98494	98563	0,06	0,07	203	98593	0,07	0,10	180	-0,03
74	135394	135471	0,02	0,06	204	135486	0,04	0,07	181	-0,01
75	135677	135744	0,08	0,05	203	135744	0,05	0,05	182	0,00
76	0	0	0,00	0,00	7	0	0,00	0,00	3	0,00
77	0	0	0,00	0,00	10	0	0,00	0,00	4	0,00
78	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
79	0	0	0,00	0,00	4	0	0,00	0,00	2	0,00
80	0	0	0,00	0,00	7	0	0,00	0,00	3	0,00
81	816	1228	7,54	50,49	131	1228	6,53	50,49	70	0,00
82	4879	5399	3,26	10,66	167	5040	10,73	3,30	151	7,12
83	973	1280	9,97	31,55	142	1306	20,09	34,22	163	-1,99
84	508	703	31,27	38,39	119	561	15,86	10,43	86	25,31
85	3780	3796	6,00	0,42	121	4006	14,16	5,98	85	-5,24
86	20751	20874	1,26	0,59	196	20859	2,97	0,52	173	0,07
87	36053	36211	0,73	0,44	199	36368	1,02	0,87	176	-0,43
88	28268	28443	1,17	0,62	195	28536	0,51	0,95	173	-0,33
89	28846	29177	1,95	1,15	194	29332	1,26	1,68	175	-0,53
90	15451	15785	1,06	2,16	189	15737	0,65	1,85	139	0,31
91	89298	89482	0,19	0,21	200	89477	0,18	0,20	177	0,01
92	66340	66403	0,45	0,09	195	66527	0,42	0,28	176	-0,19
93	61060	61144	0,31	0,14	198	61071	0,67	0,02	177	0,12
94	42453	42610	0,32	0,37	198	42718	0,57	0,62	175	-0,25

Tabela 56 - wt50: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	GA _H					GA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
95	56522	56724	0,83	0,36	200	56812	0,86	0,51	158	-0,15
96	177909	177909	0,04	0,00	204	177968	0,05	0,03	180	-0,03
97	139591	139643	0,07	0,04	203	139598	0,04	0,01	180	0,03
98	148906	148927	0,18	0,01	199	149017	0,23	0,07	178	-0,06
99	179264	179281	0,05	0,01	203	179268	0,09	0,00	180	0,01
100	120108	120171	0,12	0,05	202	120149	0,08	0,03	178	0,02
101	0	0	0,00	0,00	4	0	0,00	0,00	3	0,00
102	0	0	0,00	0,00	7	0	0,00	0,00	4	0,00
103	0	0	0,00	0,00	3	0	0,00	0,00	1	0,00
104	0	0	0,00	0,00	9	0	0,00	0,00	4	0,00
105	0	0	57,60	0,00	81	0	105,80	0,00	60	0,00
106	0	0	0,00	0,00	44	0	0,00	0,00	17	0,00
107	1717	1962	11,14	14,27	182	1940	15,01	12,99	65	1,13
108	0	0	6,00	0,00	66	0	0,00	0,00	30	0,00
109	6185	6447	4,11	4,24	185	6703	2,54	8,38	144	-3,82
110	1295	1359	11,67	4,94	150	1522	5,52	17,53	66	-10,71
111	27310	27569	0,72	0,95	186	27367	1,69	0,21	173	0,74
112	15867	15867	1,26	0,00	180	15912	3,62	0,28	96	-0,28
113	35106	35849	1,03	2,12	194	36014	1,60	2,59	142	-0,46
114	15467	15964	2,90	3,21	175	16078	2,36	3,95	163	-0,71
115	10574	11402	1,44	7,83	169	11150	2,17	5,45	97	2,26
116	35727	36087	0,70	1,01	192	36008	1,51	0,79	174	0,22
117	71922	71964	0,23	0,06	196	71951	0,24	0,04	179	0,02
118	65433	65632	0,48	0,30	194	65547	0,33	0,17	177	0,13
119	106043	106097	0,14	0,05	190	106106	0,05	0,06	179	-0,01
120	101665	101755	0,23	0,09	183	101828	0,10	0,16	179	-0,07
121	78315	78415	0,13	0,13	197	78391	0,23	0,10	172	0,03
122	119925	119930	0,09	0,00	198	120239	0,16	0,26	180	-0,26
123	101157	101170	0,26	0,01	196	101181	0,16	0,02	179	-0,01
124	139488	139594	0,08	0,08	197	139571	0,10	0,06	181	0,02
125	110392	110426	0,12	0,03	196	110477	0,15	0,08	179	-0,05
Média			17,33	1,99	153		9,62	4,92	119	-0,37

Tabela 57 - wt100: Resultados computacionais da meta-heurística GA.

Exemplar	melhor	GA _H				GA _C				
		sol	desvio	gap	t	sol	desvio	gap	t	delta
1	5988	6077	1,79	1,49	126	6031	1,69	0,72	105	0,76
2	6170	6240	0,96	1,13	147	6182	5,78	0,19	86	0,94
3	4267	4308	3,00	0,96	135	4297	3,35	0,70	97	0,26
4	5011	5011	5,35	0,00	128	5011	0,90	0,00	74	0,00
5	5283	5283	8,35	0,00	112	5283	2,22	0,00	75	0,00
6	58258	58665	1,25	0,70	148	58680	0,82	0,72	77	-0,03
7	50972	51252	1,07	0,55	153	52085	0,42	2,18	116	-1,60
8	59434	59774	0,48	0,57	137	59631	1,11	0,33	140	0,24
9	40978	41095	1,36	0,29	152	41087	1,05	0,27	113	0,02
10	53208	53576	1,27	0,69	154	53714	0,94	0,95	128	-0,26
11	181649	183122	1,03	0,81	146	184601	0,86	1,63	145	-0,80
12	234179	237757	0,45	1,53	156	236352	0,49	0,93	146	0,59
13	178840	181390	0,62	1,43	156	182088	0,22	1,82	146	-0,38
14	157476	159400	0,53	1,22	158	158689	0,68	0,77	146	0,45
15	172995	174589	0,64	0,92	157	175198	0,42	1,27	130	-0,35
16	407703	412983	0,25	1,30	157	414111	0,62	1,57	116	-0,27
17	332804	336919	0,29	1,24	161	336566	0,41	1,13	134	0,10
18	544838	550603	0,40	1,06	148	550470	0,12	1,03	130	0,02
19	477684	482930	0,18	1,10	163	480729	0,42	0,64	106	0,46
20	406094	411800	0,31	1,41	146	410582	0,38	1,11	149	0,30
21	898925	901134	0,14	0,25	165	902650	0,08	0,41	68	-0,17
22	556873	561284	0,13	0,79	151	560081	0,24	0,58	68	0,21
23	539716	543783	0,36	0,75	134	542460	0,27	0,51	100	0,24
24	744287	746791	0,13	0,34	152	746670	0,17	0,32	123	0,02
25	585306	590511	0,10	0,89	171	588715	0,32	0,58	157	0,31
26	8	8	1880,00	0,00	108	8	1400,00	0,00	42	0,00
27	718	848	11,89	18,11	139	929	4,09	29,39	81	-8,72
28	27	27	500,74	0,00	92	27	362,22	0,00	58	0,00
29	480	554	16,61	15,42	124	482	40,08	0,42	107	14,94
30	50	50	0,00	0,00	102	50	36,40	0,00	104	0,00
31	24202	24814	2,80	2,53	148	24538	2,65	1,39	125	1,12
32	25469	27049	3,23	6,20	154	25868	7,59	1,57	142	4,57
33	32964	33930	3,39	2,93	155	34981	3,56	6,12	142	-3,00
34	22215	24418	1,43	9,92	145	24252	3,86	9,17	142	0,68
35	19114	20124	3,47	5,28	149	20903	2,66	9,36	140	-3,73
36	108293	111707	1,69	3,15	157	111176	2,06	2,66	139	0,48
37	181850	186463	1,13	2,54	157	184626	1,84	1,53	127	0,99
38	90440	94754	1,41	4,77	157	97093	0,75	7,36	105	-2,41
39	151701	156375	1,37	3,08	150	157511	0,86	3,83	144	-0,72
40	129728	132206	1,59	1,91	158	135466	2,02	4,42	145	-2,41
41	462324	465037	0,66	0,59	165	467908	0,11	1,21	152	-0,61
42	425875	429238	0,34	0,79	158	428976	0,34	0,73	141	0,06
43	320537	324913	0,51	1,37	160	324473	1,01	1,23	81	0,14
44	360193	365474	0,50	1,47	158	365793	0,14	1,55	150	-0,09
45	306040	310929	0,79	1,60	142	310664	0,74	1,51	150	0,09
46	829828	832571	0,14	0,33	174	833091	0,18	0,39	156	-0,06
47	623356	625416	0,34	0,33	160	625315	0,20	0,31	79	0,02

Tabela 57 - wt100: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	GA _H					GA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
48	748988	753891	0,15	0,65	171	755146	0,12	0,82	79	-0,17
49	656693	662142	0,25	0,83	150	659328	0,16	0,40	119	0,43
50	599269	601686	0,60	0,40	155	603314	0,21	0,67	155	-0,27
51	0	0	0,00	0,00	32	0	0,00	0,00	8	0,00
52	0	0	0,00	0,00	28	0	0,00	0,00	11	0,00
53	0	0	0,00	0,00	15	0	0,00	0,00	8	0,00
54	0	0	0,00	0,00	60	0	0,00	0,00	19	0,00
55	0	0	0,00	0,00	44	0	0,00	0,00	20	0,00
56	9046	10909	6,11	20,59	143	9870	16,23	9,11	102	10,53
57	11539	11792	6,64	2,19	150	12328	8,31	6,84	115	-4,35
58	16313	19418	10,27	19,03	156	19954	10,29	22,32	139	-2,69
59	7965	8453	6,42	6,13	155	8156	11,37	2,40	139	3,64
60	19912	22471	7,36	12,85	154	21710	4,99	9,03	108	3,51
61	86793	95862	1,98	10,45	154	94894	1,64	9,33	144	1,02
62	87067	94797	2,49	8,88	164	94081	3,23	8,06	103	0,76
63	96563	104511	4,08	8,23	164	105863	2,97	9,63	108	-1,28
64	100788	104485	1,91	3,67	156	104231	1,33	3,42	142	0,24
65	56510	61347	3,58	8,56	160	60779	2,83	7,55	118	0,93
66	243872	249084	0,71	2,14	144	249185	0,62	2,18	139	-0,04
67	401023	407373	0,30	1,58	154	407410	0,74	1,59	99	-0,01
68	399085	405708	0,66	1,66	173	406478	0,68	1,85	153	-0,19
69	309232	315875	1,01	2,15	167	314551	1,28	1,72	152	0,42
70	222684	229875	0,95	3,23	150	228458	1,59	2,59	113	0,62
71	640816	645401	0,13	0,72	178	643547	0,23	0,43	158	0,29
72	611362	615766	0,22	0,72	174	615209	0,43	0,63	158	0,09
73	623429	626105	0,41	0,43	179	628201	0,17	0,77	158	-0,33
74	584628	590730	0,20	1,04	167	590174	0,20	0,95	156	0,09
75	575274	578222	0,28	0,51	174	577505	0,52	0,39	155	0,12
76	0	0	0,00	0,00	51	0	0,00	0,00	18	0,00
77	0	0	0,00	0,00	27	0	0,00	0,00	8	0,00
78	0	0	0,00	0,00	30	0	0,00	0,00	19	0,00
79	0	0	0,00	0,00	34	0	0,00	0,00	26	0,00
80	0	0	0,00	0,00	36	0	0,00	0,00	20	0,00
81	1400	3190	50,23	127,86	148	5452	12,67	289,43	139	-41,49
82	317	2932	31,93	824,92	150	1941	56,20	512,30	137	51,06
83	1146	3208	41,74	179,93	154	5018	14,27	337,87	59	-36,07
84	136	356	63,37	161,76	133	163	401,47	19,85	109	118,40
85	284	3870	15,06	1262,68	126	3696	20,77	1201,41	128	4,71
86	66850	78123	2,69	16,86	161	79716	6,73	19,25	140	-2,00
87	84229	94251	4,81	11,90	162	95484	5,66	13,36	146	-1,29
88	55544	62811	6,26	13,08	152	64420	5,37	15,98	72	-2,50
89	54612	58791	5,51	7,65	147	61206	3,46	12,07	143	-3,95
90	75061	83475	3,36	11,21	165	82910	2,13	10,46	109	0,68
91	248699	256721	1,10	3,23	162	252294	1,90	1,45	98	1,75
92	311022	317294	1,03	2,02	167	316408	1,13	1,73	153	0,28
93	326258	333987	0,17	2,37	165	333692	0,60	2,28	115	0,09
94	273993	281098	0,80	2,59	152	279889	1,32	2,15	101	0,43

Tabela 57 - wt100: Resultados computacionais da meta-heurística GA. (cont.)

Exemplar	GA _H					GA _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
95	316870	324111	1,06	2,29	159	323125	0,80	1,97	126	0,31
96	495516	498824	0,30	0,67	161	498420	0,37	0,59	140	0,08
97	636903	641094	0,31	0,66	174	641596	0,36	0,74	156	-0,08
98	680082	683301	0,14	0,47	147	681866	0,14	0,26	157	0,21
99	622464	625872	0,40	0,55	171	626213	0,26	0,60	157	-0,05
100	449545	452730	0,49	0,71	175	454587	0,20	1,12	154	-0,41
101	0	0	0,00	0,00	36	0	0,00	0,00	35	0,00
102	0	0	0,00	0,00	13	0	0,00	0,00	11	0,00
103	0	0	0,00	0,00	33	0	0,00	0,00	13	0,00
104	0	0	0,00	0,00	24	0	0,00	0,00	21	0,00
105	0	0	0,00	0,00	16	0	0,00	0,00	7	0,00
106	0	0	5,60	0,00	119	0	0,00	0,00	135	0,00
107	1193	6944	14,11	482,06	142	5654	25,31	373,93	139	22,82
108	0	1748	31,29	1748,00	152	1466	63,30	1466,00	111	19,24
109	232	1317	210,86	467,67	142	1740	41,86	650,00	137	-24,31
110	0	3210	48,56	3210,00	154	1200	205,57	1200,00	139	167,50
111	159123	172560	0,74	8,44	167	170537	1,12	7,17	150	1,19
112	174367	180986	1,87	3,80	167	184999	1,10	6,10	94	-2,17
113	91169	100933	1,73	10,71	166	101023	2,45	10,81	146	-0,09
114	168266	176877	1,53	5,12	173	178227	1,18	5,92	148	-0,76
115	70190	79658	2,37	13,49	170	80494	3,60	14,68	102	-1,04
116	370614	375575	1,11	1,34	166	375835	0,39	1,41	153	-0,07
117	324437	331716	0,78	2,24	176	330957	0,88	2,01	149	0,23
118	246237	254542	0,52	3,37	165	254630	0,97	3,41	151	-0,03
119	293571	299030	1,61	1,86	159	302550	0,48	3,06	154	-1,16
120	267316	273156	0,72	2,18	177	273101	0,79	2,16	151	0,02
121	471214	478896	0,27	1,63	172	479694	0,19	1,80	156	-0,17
122	570459	573861	0,31	0,60	178	573946	0,19	0,61	118	-0,01
123	397029	401133	0,67	1,03	164	402801	0,54	1,45	93	-0,41
124	431115	434581	0,43	0,80	177	435712	0,38	1,07	122	-0,26
125	560754	567964	0,36	1,29	179	568686	0,51	1,41	124	-0,13
Média			24,62	70,56	140		22,83	51,24	112	2,29

B.3 META-HEURÍSTICA VNS

Tabela 58 - wt40: Resultados computacionais da meta-heurística VNS.

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	913	913	1,49	0,00	15	930	0,00	1,86	14	-1,83
2	1225	1225	0,00	0,00	4	1225	0,00	0,00	1	0,00
3	537	537	2,68	0,00	7	537	2,68	0,00	7	0,00
4	2094	2094	0,00	0,00	0	2094	0,00	0,00	0	0,00
5	990	990	0,00	0,00	0	990	0,00	0,00	0	0,00
6	6955	6955	0,00	0,00	1	6955	0,00	0,00	0	0,00
7	6324	6324	0,00	0,00	4	6324	0,00	0,00	6	0,00
8	6865	6865	0,00	0,00	2	6865	0,00	0,00	2	0,00
9	16225	16225	0,00	0,00	4	16225	0,00	0,00	3	0,00
10	9737	9737	0,00	0,00	5	9737	0,00	0,00	1	0,00
11	17465	17465	0,00	0,00	4	17465	0,00	0,00	0	0,00
12	19312	19312	0,00	0,00	0	19312	0,00	0,00	0	0,00
13	29256	29313	0,00	0,19	23	29258	0,26	0,01	2	0,19
14	14377	14432	0,09	0,38	28	14377	0,37	0,00	20	0,38
15	26914	26914	0,00	0,00	3	26914	0,33	0,00	1	0,00
16	72317	72378	0,43	0,08	28	72378	0,22	0,08	22	0,00
17	78623	78747	0,07	0,16	32	78747	0,00	0,16	25	0,00
18	74310	74310	0,01	0,00	35	74310	0,16	0,00	27	0,00
19	77122	77432	0,00	0,40	25	77122	0,32	0,00	23	0,40
20	63229	63368	0,14	0,22	24	63368	0,43	0,22	23	0,00
21	77774	77774	0,01	0,00	38	77776	0,01	0,00	7	0,00
22	100484	100484	0,00	0,00	8	100484	0,03	0,00	3	0,00
23	135618	135618	0,00	0,00	6	135620	0,00	0,00	2	0,00
24	119947	119947	0,02	0,00	27	119947	0,02	0,00	9	0,00
25	128747	128747	0,00	0,00	6	128747	0,00	0,00	2	0,00
26	108	108	0,00	0,00	0	108	0,00	0,00	0	0,00
27	64	64	0,00	0,00	2	64	0,00	0,00	0	0,00
28	15	15	0,00	0,00	0	15	0,00	0,00	0	0,00
29	47	47	0,00	0,00	0	47	0,00	0,00	0	0,00
30	98	98	0,00	0,00	0	98	0,00	0,00	0	0,00
31	6575	6659	0,00	1,28	16	6659	0,14	1,28	15	0,00
32	4098	4098	0,02	0,00	14	4098	0,01	0,00	2	0,00
33	5468	5614	1,68	2,67	19	5614	1,68	2,67	15	0,00
34	2648	2648	0,00	0,00	1	2648	0,00	0,00	0	0,00
35	5290	5290	0,86	0,00	16	5341	0,09	0,96	17	-0,95
36	19732	19732	1,02	0,00	20	20095	0,00	1,84	19	-1,81
37	17349	17349	0,23	0,00	17	17349	0,23	0,00	10	0,00
38	24499	24563	0,00	0,26	19	24563	0,00	0,26	19	0,00
39	19008	19008	0,62	0,00	20	19049	1,78	0,22	29	-0,22
40	19611	19611	1,14	0,00	15	19611	1,52	0,00	5	0,00
41	57640	57640	0,33	0,00	32	57640	0,17	0,00	17	0,00
42	81462	81515	0,17	0,07	34	81462	0,22	0,00	23	0,07
43	65134	65134	0,00	0,00	3	65134	0,00	0,00	1	0,00
44	78139	78139	0,08	0,00	16	78139	0,04	0,00	2	0,00

Tabela 58 - wt40: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
45	66579	66579	0,00	0,00	13	66579	0,00	0,00	8	0,00
46	64451	64527	0,00	0,12	34	64479	0,06	0,04	31	0,07
47	113999	114000	0,01	0,00	25	114019	0,00	0,02	25	-0,02
48	74323	74323	0,00	0,00	10	74323	0,00	0,00	6	0,00
49	110295	110298	0,01	0,00	26	110298	0,01	0,00	4	0,00
50	95616	95616	0,01	0,00	2	95620	0,00	0,00	1	0,00
51	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
52	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
53	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
54	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
55	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
56	2099	2099	1,74	0,00	10	2099	6,97	0,00	3	0,00
57	2260	2338	5,77	3,45	23	2260	6,11	0,00	9	3,45
58	4936	6251	0,00	26,64	15	4936	21,31	0,00	7	26,64
59	3784	3784	0,00	0,00	8	3784	0,00	0,00	0	0,00
60	3289	3289	0,00	0,00	0	3289	0,00	0,00	0	0,00
61	20281	20335	0,98	0,27	25	20335	1,24	0,27	22	0,00
62	13403	13711	0,89	2,30	23	13625	1,40	1,66	21	0,63
63	19771	19771	0,76	0,00	20	19771	0,94	0,00	13	0,00
64	24346	24498	2,12	0,62	24	25146	0,01	3,29	18	-2,58
65	14905	14905	2,66	0,00	20	15157	1,50	1,69	25	-1,66
66	65386	65420	0,00	0,05	30	65420	0,34	0,05	27	0,00
67	65756	65807	0,18	0,08	27	65766	0,29	0,02	26	0,06
68	78451	78451	0,00	0,00	2	78451	0,00	0,00	1	0,00
69	81627	81627	0,01	0,00	28	81627	0,09	0,00	17	0,00
70	68242	68257	0,01	0,02	34	68257	0,00	0,02	29	0,00
71	90486	90486	0,01	0,00	17	90486	0,00	0,00	8	0,00
72	115249	115249	0,00	0,00	6	115249	0,07	0,00	2	0,00
73	68529	68529	0,01	0,00	10	68529	0,02	0,00	3	0,00
74	79006	79011	0,20	0,01	14	79013	0,23	0,01	4	0,00
75	98110	98362	0,00	0,26	28	98110	0,10	0,00	7	0,26
76	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
77	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
78	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
79	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
80	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
81	684	684	7,37	0,00	9	684	36,81	0,00	1	0,00
82	172	172	0,00	0,00	4	172	113,72	0,00	1	0,00
83	798	876	0,00	9,77	17	876	0,00	9,77	13	0,00
84	617	617	78,57	0,00	13	617	0,00	0,00	0	0,00
85	776	776	5,00	0,00	20	818	1,64	5,41	16	-5,13
86	10262	10477	0,03	2,10	26	10477	0,05	2,10	17	0,00
87	18646	18741	0,89	0,51	36	18646	0,61	0,00	15	0,51
88	10021	10053	6,09	0,32	29	10727	0,82	7,05	20	-6,28
89	25881	25912	0,78	0,12	29	25912	0,32	0,12	22	0,00
90	8159	8159	5,66	0,00	36	8159	2,89	0,00	21	0,00
91	47683	47683	0,89	0,00	34	47836	0,44	0,32	36	-0,32

Tabela 58 - wt40: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	VNS _H					VNS _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
92	43004	43172	0,04	0,39	32	43004	0,23	0,00	15	0,39
93	55730	56143	0,00	0,74	25	55938	0,27	0,37	22	0,37
94	59494	59494	0,45	0,00	36	59622	0,14	0,22	24	-0,21
95	42688	42823	0,09	0,32	27	42823	0,15	0,32	22	0,00
96	126048	126109	0,00	0,05	27	126059	0,03	0,01	2	0,04
97	114686	114695	0,00	0,01	6	114689	0,00	0,00	4	0,01
98	112102	112102	0,00	0,00	7	112102	0,00	0,00	2	0,00
99	98206	98206	0,04	0,00	17	98206	0,06	0,00	9	0,00
100	157296	157303	0,00	0,00	2	157299	0,00	0,00	1	0,00
101	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
102	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
103	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
104	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
105	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
106	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
107	516	598	0,00	15,89	17	598	0,00	15,89	17	0,00
108	3354	3623	8,63	8,02	25	3623	9,15	8,02	17	0,00
109	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
110	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
111	31478	31478	0,41	0,00	14	31612	0,20	0,43	21	-0,42
112	21169	21236	0,36	0,32	24	21236	0,00	0,32	23	0,00
113	27077	27137	0,19	0,22	28	27137	0,25	0,22	21	0,00
114	19648	19648	1,11	0,00	19	19648	1,09	0,00	6	0,00
115	13774	14036	2,32	1,90	22	13780	3,68	0,04	21	1,86
116	46770	46782	0,16	0,03	42	46782	0,16	0,03	31	0,00
117	50364	50364	0,02	0,00	25	50384	0,00	0,04	22	-0,04
118	25460	25510	0,03	0,20	27	25510	0,26	0,20	25	0,00
119	66707	66723	0,00	0,02	32	66707	0,04	0,00	25	0,02
120	69019	69092	0,02	0,11	31	69092	0,00	0,11	25	0,00
121	122266	122370	0,00	0,09	39	122370	0,00	0,09	37	0,00
122	82456	82456	0,06	0,00	24	82456	0,05	0,00	4	0,00
123	75118	75118	0,23	0,00	57	75118	0,23	0,00	27	0,00
124	73041	73527	0,00	0,67	45	73041	1,11	0,00	10	0,67
125	104531	104531	0,00	0,00	4	104531	0,01	0,00	2	0,00
Média			1,17	0,65	16		1,81	0,54	10	0,12

Tabela 59 - wt50: Resultados computacionais da meta-heurística VNS.

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	2134	2134	0,00	0,00	2	2134	0,00	0,00	0	0,00
2	1996	1998	0,52	0,10	35	1996	0,60	0,00	23	0,10
3	2583	2583	0,00	0,00	2	2583	0,00	0,00	0	0,00
4	2691	2691	0,00	0,00	0	2691	0,00	0,00	0	0,00
5	1518	1518	4,53	0,00	25	1518	4,53	0,00	7	0,00
6	26276	26276	0,00	0,00	28	26276	0,00	0,00	16	0,00
7	11403	11403	0,00	0,00	2	11403	0,00	0,00	1	0,00
8	8499	8499	0,11	0,00	8	8499	0,00	0,00	1	0,00
9	9884	9918	0,00	0,34	32	9918	0,00	0,34	27	0,00
10	10655	10655	0,00	0,00	14	10655	1,24	0,00	3	0,00
11	43504	43504	0,00	0,00	2	43504	0,00	0,00	1	0,00
12	36378	36378	0,42	0,00	49	36378	0,56	0,00	36	0,00
13	45383	45383	0,00	0,00	2	45383	0,00	0,00	1	0,00
14	51785	51974	0,06	0,36	56	51974	0,12	0,36	35	0,00
15	38934	38934	0,00	0,00	14	38934	0,00	0,00	3	0,00
16	87902	87902	0,28	0,00	50	87902	0,33	0,00	14	0,00
17	84260	84375	0,00	0,14	53	84375	0,00	0,14	40	0,00
18	104795	104795	0,06	0,00	43	104795	0,16	0,00	3	0,00
19	89299	89474	0,14	0,20	71	89474	0,00	0,20	41	0,00
20	72316	72316	0,00	0,00	18	72316	0,00	0,00	6	0,00
21	214546	214563	0,00	0,01	38	214562	0,00	0,01	15	0,00
22	150800	150802	0,00	0,00	23	150805	0,00	0,00	2	0,00
23	224025	224025	0,01	0,00	33	224025	0,00	0,00	6	0,00
24	116015	116015	0,00	0,00	14	116015	0,00	0,00	9	0,00
25	240179	240183	0,00	0,00	12	240182	0,00	0,00	5	0,00
26	2	2	0,00	0,00	0	2	0,00	0,00	0	0,00
27	4	4	0,00	0,00	0	4	0,00	0,00	0	0,00
28	755	755	0,00	0,00	2	755	0,00	0,00	1	0,00
29	99	99	0,00	0,00	7	99	0,00	0,00	0	0,00
30	22	22	0,00	0,00	0	22	0,00	0,00	0	0,00
31	9934	9934	0,00	0,00	4	9934	0,00	0,00	1	0,00
32	7178	7178	0,74	0,00	41	7187	0,81	0,13	28	-0,13
33	4674	4864	0,00	4,07	41	4674	2,44	0,00	23	4,07
34	4017	4017	0,00	0,00	2	4017	0,00	0,00	1	0,00
35	6459	6459	0,00	0,00	18	6459	0,00	0,00	4	0,00
36	34892	35425	2,71	1,53	58	35434	3,32	1,55	31	-0,03
37	22739	23001	1,31	1,15	47	22740	1,44	0,00	34	1,15
38	29467	29709	0,00	0,82	44	29709	0,00	0,82	32	0,00
39	49352	50525	0,00	2,38	45	50525	0,00	2,38	38	0,00
40	26423	26423	3,73	0,00	45	26423	3,40	0,00	15	0,00
41	71111	71111	0,58	0,00	54	71111	0,31	0,00	33	0,00
42	90163	90163	0,00	0,00	91	90163	0,01	0,00	33	0,00
43	84126	84126	0,05	0,00	35	84126	0,15	0,00	9	0,00
44	123893	124037	0,33	0,12	111	124020	0,30	0,10	43	0,01
45	79883	79900	0,05	0,02	103	79900	0,06	0,02	60	0,00
46	157505	157508	0,00	0,00	31	157509	0,00	0,00	6	0,00
47	133289	133289	0,01	0,00	25	133289	0,02	0,00	8	0,00

Tabela 59 - wt50: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	VNS _H					VNS _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
48	191099	191099	0,01	0,00	47	191108	0,03	0,00	17	0,00
49	150279	150279	0,01	0,00	7	150279	0,00	0,00	1	0,00
50	198076	198076	0,03	0,00	58	198076	0,02	0,00	7	0,00
51	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
52	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
53	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
54	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
55	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
56	1258	1258	1,91	0,00	16	1258	1,91	0,00	8	0,00
57	3679	3834	2,40	4,21	38	3947	0,04	7,28	26	-2,86
58	2522	2522	0,00	0,00	9	2522	2,46	0,00	0	0,00
59	3770	3770	12,06	0,00	41	3770	7,35	0,00	21	0,00
60	5904	5912	4,42	0,14	53	5904	10,74	0,00	7	0,14
61	25212	25212	4,25	0,00	53	25320	1,68	0,43	34	-0,43
62	17337	17533	2,34	1,13	67	17791	0,27	2,62	46	-1,45
63	30729	30916	0,78	0,61	63	30908	0,54	0,58	35	0,03
64	18082	18561	0,02	2,65	51	18561	0,01	2,65	39	0,00
65	25028	25228	0,39	0,80	74	25049	0,84	0,08	44	0,71
66	76878	76889	0,00	0,01	73	76889	0,01	0,01	49	0,00
67	85413	85862	0,17	0,53	93	85685	0,30	0,32	58	0,21
68	92756	93458	0,82	0,76	89	94054	1,08	1,40	55	-0,63
69	77930	77933	0,16	0,00	124	77955	0,15	0,03	81	-0,03
70	74750	75122	0,60	0,50	88	74780	0,65	0,04	53	0,46
71	150580	150580	0,00	0,00	26	150580	0,00	0,00	12	0,00
72	131680	131703	0,00	0,02	65	131703	0,02	0,02	59	0,00
73	98494	98530	0,13	0,04	89	98530	0,09	0,04	63	0,00
74	135394	135532	0,01	0,10	83	135555	0,00	0,12	69	-0,02
75	135677	135683	0,00	0,00	58	135683	0,00	0,00	9	0,00
76	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
77	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
78	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
79	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
80	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
81	816	1228	0,00	50,49	32	1228	0,00	50,49	26	0,00
82	4879	5583	0,19	14,43	41	5583	0,39	14,43	30	0,00
83	973	1196	0,10	22,92	33	1196	0,00	22,92	26	0,00
84	508	633	0,70	24,61	32	508	14,76	0,00	3	24,61
85	3780	3796	10,80	0,42	46	3796	6,16	0,42	28	0,00
86	20751	20823	3,52	0,35	65	20809	0,68	0,28	37	0,07
87	36053	36060	0,46	0,02	85	36053	0,50	0,00	29	0,02
88	28268	28504	1,59	0,83	53	28504	1,11	0,83	51	0,00
89	28846	29291	0,68	1,54	65	29485	0,16	2,22	42	-0,66
90	15451	16226	0,22	5,02	56	16052	0,87	3,89	37	1,08
91	89298	89412	0,03	0,13	105	89445	0,01	0,16	53	-0,04
92	66340	67080	0,99	1,12	66	67080	0,27	1,12	59	0,00
93	61060	61125	0,06	0,11	114	61125	0,08	0,11	67	0,00
94	42453	42547	0,59	0,22	92	42547	0,22	0,22	71	0,00

Tabela 59 - wt50: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
95	56522	57260	1,69	1,31	65	56549	1,43	0,05	49	1,26
96	177909	177975	0,00	0,04	71	177909	0,02	0,00	20	0,04
97	139591	139771	0,02	0,13	84	139591	0,09	0,00	57	0,13
98	148906	149036	0,01	0,09	141	148998	0,03	0,06	71	0,03
99	179264	179274	0,04	0,01	80	179275	0,07	0,01	22	0,00
100	120108	120108	0,01	0,00	38	120108	0,01	0,00	11	0,00
101	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
102	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
103	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
104	0	0	0,00	0,00	0	0	0,00	0,00	0	0,00
105	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
106	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
107	1717	1902	0,45	10,77	35	1902	0,50	10,77	29	0,00
108	0	0	0,00	0,00	1	0	0,00	0,00	0	0,00
109	6185	6204	0,80	0,31	42	6216	0,72	0,50	29	-0,19
110	1295	1440	0,00	11,20	32	1440	0,00	11,20	26	0,00
111	27310	27310	1,99	0,00	54	28081	0,28	2,82	36	-2,75
112	15867	16167	5,46	1,89	45	16055	5,26	1,18	35	0,70
113	35106	35263	2,49	0,45	66	36095	0,38	2,82	39	-2,31
114	15467	15579	1,92	0,72	53	15719	1,20	1,63	35	-0,89
115	10574	12127	6,29	14,69	49	12127	6,31	14,69	38	0,00
116	35727	35730	0,61	0,01	73	35749	1,39	0,06	43	-0,05
117	71922	72057	0,05	0,19	125	71957	0,11	0,05	104	0,14
118	65433	65556	0,00	0,19	76	65433	0,12	0,00	22	0,19
119	106043	106051	0,05	0,01	59	106043	0,06	0,00	9	0,01
120	101665	101783	0,12	0,12	130	101783	0,10	0,12	57	0,00
121	78315	78390	0,45	0,10	86	78390	0,46	0,10	47	0,00
122	119925	120578	0,25	0,54	88	120619	0,00	0,58	62	-0,03
123	101157	101170	0,14	0,01	120	101170	0,14	0,01	69	0,00
124	139488	139596	0,00	0,08	135	139521	0,04	0,02	69	0,05
125	110392	110392	0,07	0,00	88	110405	0,13	0,01	44	-0,01
Média			0,71	1,50	44		0,74	1,32	25	0,18

Tabela 60 - wt100: Resultados computacionais da meta-heurística VNS.

Exemplar	melhor	VNS _H				VNS _C				delta
		sol	desvio	gap	t	sol	desvio	gap	t	
1	5988	6066	1,50	1,30	203	6066	0,00	1,30	172	0,00
2	6170	6182	0,00	0,19	172	6182	0,00	0,19	136	0,00
3	4267	4267	0,00	0,00	8	4267	0,00	0,00	4	0,00
4	5011	5011	0,00	0,00	4	5011	0,00	0,00	2	0,00
5	5283	5283	4,24	0,00	152	5283	3,18	0,00	78	0,00
6	58258	58258	0,45	0,00	175	58258	0,60	0,00	121	0,00
7	50972	51328	0,30	0,70	253	51318	0,31	0,68	172	0,02
8	59434	59624	0,12	0,32	268	59624	0,09	0,32	257	0,00
9	40978	41005	0,00	0,07	250	41005	0,00	0,07	192	0,00
10	53208	53251	0,64	0,08	263	53251	0,01	0,08	203	0,00
11	181649	182894	0,50	0,69	288	183749	0,51	1,16	210	-0,47
12	234179	234315	0,53	0,06	281	234718	0,46	0,23	226	-0,17
13	178840	179010	0,32	0,10	280	178974	0,35	0,07	228	0,02
14	157476	157651	0,00	0,11	276	157585	0,12	0,07	225	0,04
15	172995	173554	0,08	0,32	279	173075	0,23	0,05	215	0,28
16	407703	408266	0,16	0,14	300	408597	0,10	0,22	299	-0,08
17	332804	332810	0,32	0,00	251	333768	0,16	0,29	291	-0,29
18	544838	545186	0,12	0,06	300	545764	0,12	0,17	278	-0,11
19	477684	478482	0,22	0,17	294	478548	0,15	0,18	299	-0,01
20	406094	406424	0,01	0,08	297	406499	0,03	0,10	281	-0,02
21	898925	898998	0,00	0,01	112	898994	0,00	0,01	68	0,00
22	556873	556973	0,01	0,02	300	556987	0,01	0,02	299	0,00
23	539716	539886	0,01	0,03	301	539940	0,01	0,04	300	-0,01
24	744287	744322	0,00	0,00	176	744287	0,00	0,00	22	0,00
25	585306	585367	0,01	0,01	300	585375	0,01	0,01	299	0,00
26	8	8	0,00	0,00	2	8	0,00	0,00	1	0,00
27	718	823	0,00	14,62	170	718	8,77	0,00	16	14,62
28	27	27	0,00	0,00	1	27	0,00	0,00	1	0,00
29	480	480	11,96	0,00	158	480	0,08	0,00	20	0,00
30	50	50	0,00	0,00	2	50	0,00	0,00	1	0,00
31	24202	24243	0,26	0,17	255	24243	0,22	0,17	251	0,00
32	25469	25895	0,99	1,67	257	26045	1,06	2,26	258	-0,58
33	32964	33612	0,31	1,97	230	33612	0,97	1,97	174	0,00
34	22215	22546	2,75	1,49	231	22393	2,05	0,80	163	0,68
35	19114	19821	2,78	3,70	254	19821	0,40	3,70	213	0,00
36	108293	114348	0,34	5,59	277	112745	1,14	4,11	214	1,42
37	181850	182751	1,36	0,50	300	183687	1,08	1,01	257	-0,51
38	90440	90470	0,45	0,03	283	90644	0,57	0,23	225	-0,19
39	151701	154006	0,39	1,52	286	153082	1,59	0,91	299	0,60
40	129728	129747	0,85	0,01	291	129747	0,89	0,01	266	0,00
41	462324	463808	0,08	0,32	300	463304	0,18	0,21	299	0,11
42	425875	426311	0,19	0,10	300	426311	0,19	0,10	299	0,00
43	320537	321510	0,24	0,30	299	321510	0,41	0,30	299	0,00
44	360193	361830	0,26	0,45	299	361789	0,52	0,44	299	0,01
45	306040	307514	0,09	0,48	300	306971	0,20	0,30	299	0,18
46	829828	830074	0,01	0,03	300	830074	0,00	0,03	299	0,00
47	623356	623477	0,03	0,02	300	623693	0,01	0,05	299	-0,03

Tabela 60 - wt100: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	VNS _H					VNS _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
48	748988	749326	0,05	0,05	300	749762	0,00	0,10	300	-0,06
49	656693	656725	0,01	0,00	213	656746	0,02	0,01	76	0,00
50	599269	599416	0,01	0,02	300	599416	0,01	0,02	299	0,00
51	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
52	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
53	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
54	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
55	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
56	9046	9478	2,51	4,78	220	9730	6,15	7,56	164	-2,59
57	11539	11539	8,49	0,00	185	11539	2,39	0,00	52	0,00
58	16313	16896	2,87	3,57	272	16996	4,35	4,19	200	-0,59
59	7965	8156	0,00	2,40	180	8156	0,00	2,40	145	0,00
60	19912	21031	11,83	5,62	230	21031	12,17	5,62	162	0,00
61	86793	88057	0,83	1,46	281	87592	0,59	0,92	299	0,53
62	87067	89884	0,31	3,24	300	89884	0,68	3,24	299	0,00
63	96563	98126	0,43	1,62	290	98347	1,51	1,85	241	-0,22
64	100788	102593	0,00	1,79	249	102593	0,65	1,79	190	0,00
65	56510	60189	0,00	6,51	286	59677	3,77	5,60	196	0,86
66	243872	244583	0,18	0,29	299	244613	0,14	0,30	299	-0,01
67	401023	401858	0,10	0,21	299	401748	0,13	0,18	299	0,03
68	399085	400684	0,09	0,40	300	400847	0,09	0,44	299	-0,04
69	309232	311359	0,11	0,69	294	311431	0,11	0,71	299	-0,02
70	222684	224339	0,69	0,74	293	224359	0,99	0,75	252	-0,01
71	640816	641603	0,05	0,12	300	641521	0,06	0,11	299	0,01
72	611362	611704	0,00	0,06	300	611826	0,03	0,08	299	-0,02
73	623429	623491	0,03	0,01	251	623479	0,01	0,01	114	0,00
74	584628	585019	0,04	0,07	300	585050	0,05	0,07	299	-0,01
75	575274	575456	0,05	0,03	300	575449	0,07	0,03	299	0,00
76	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
77	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
78	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
79	0	0	0,00	0,00	2	0	0,00	0,00	1	0,00
80	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
81	1400	2167	18,52	54,79	226	2167	5,33	54,79	141	0,00
82	317	317	1,26	0,00	102	317	2,71	0,00	3	0,00
83	1146	2272	0,93	98,25	193	2272	1,85	98,25	138	0,00
84	136	136	97,06	0,00	105	136	129,41	0,00	27	0,00
85	284	341	15,31	20,07	166	341	0,00	20,07	136	0,00
86	66850	71104	2,95	6,36	267	71326	1,79	6,70	199	-0,31
87	84229	86449	0,60	2,64	300	87282	0,54	3,62	299	-0,95
88	55544	57185	2,61	2,95	291	58012	1,88	4,44	203	-1,43
89	54612	55238	2,05	1,15	286	56372	0,90	3,22	284	-2,01
90	75061	76319	0,14	1,68	288	76366	0,21	1,74	299	-0,06
91	248699	249045	0,26	0,14	300	249074	0,36	0,15	299	-0,01
92	311022	313174	0,53	0,69	300	313434	0,51	0,78	299	-0,08
93	326258	327497	0,21	0,38	299	328297	0,16	0,62	299	-0,24
94	273993	275214	0,10	0,45	300	275118	0,10	0,41	299	0,03

Tabela 60 - wt100: Resultados computacionais da meta-heurística VNS. (cont.)

Exemplar	VNS _H					VNS _C				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
95	316870	318008	0,20	0,36	300	318025	0,15	0,36	299	-0,01
96	495516	495908	0,07	0,08	299	495751	0,19	0,05	299	0,03
97	636903	636963	0,02	0,01	270	637141	0,03	0,04	299	-0,03
98	680082	680286	0,01	0,03	300	680265	0,02	0,03	299	0,00
99	622464	622669	0,03	0,03	300	622852	0,03	0,06	299	-0,03
100	449545	449842	0,03	0,07	300	449889	0,09	0,08	300	-0,01
101	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
102	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
103	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
104	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
105	0	0	0,00	0,00	1	0	0,00	0,00	1	0,00
106	0	0	0,00	0,00	3	0	0,00	0,00	2	0,00
107	1193	1198	6,18	0,42	203	1193	4,61	0,00	155	0,42
108	0	0	0,00	0,00	4	0	0,00	0,00	2	0,00
109	232	232	97,93	0,00	148	232	97,93	0,00	34	0,00
110	0	0	0,00	0,00	11	0	0,00	0,00	5	0,00
111	159123	160197	0,74	0,67	299	161217	0,62	1,32	299	-0,63
112	174367	176538	0,83	1,25	299	177791	0,28	1,96	300	-0,70
113	91169	92788	1,63	1,78	287	93000	1,77	2,01	245	-0,23
114	168266	169403	0,20	0,68	295	168891	0,43	0,37	299	0,30
115	70190	71296	1,43	1,58	300	71953	1,37	2,51	299	-0,91
116	370614	371426	0,04	0,22	300	371228	0,06	0,17	299	0,05
117	324437	326009	0,32	0,48	300	325883	0,27	0,45	300	0,04
118	246237	246594	0,09	0,14	299	246498	0,08	0,11	299	0,04
119	293571	294257	0,01	0,23	299	294305	0,16	0,25	299	-0,02
120	267316	267913	0,07	0,22	300	267895	0,04	0,22	299	0,01
121	471214	472103	0,05	0,19	300	472221	0,06	0,21	299	-0,02
122	570459	570588	0,02	0,02	300	570680	0,03	0,04	299	-0,02
123	397029	397956	0,07	0,23	300	398015	0,06	0,25	299	-0,01
124	431115	432438	0,02	0,31	300	432501	0,15	0,32	299	-0,01
125	560754	561729	0,15	0,17	300	562442	0,19	0,30	299	-0,13
Média			2,51	2,16	217		2,52	2,11	193	0,05

APÊNCICE C - Resultados Computacionais - Problema do Caixeiro Viajante com Coleta de Prêmios

C.1 META-HEURÍSTICA CS-SA

Tabela 61 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-SA.

Exemplar	CS-SA _H					CS-CS _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>burma14</i>	797	797	0,00	0,00	0,00	797	0,00	0,00	0,07	0,00
<i>ulysses22</i>	1699	1699	0,00	0,00	1,94	1699	0,00	0,00	0,46	0,00
<i>berlin52</i>	2189	2189	0,00	0,00	1,41	2189	0,00	0,00	1,25	0,00
<i>st70</i>	665	665	0,00	0,00	1,60	665	0,00	0,00	1,64	0,00
<i>pr76</i>	20390	21246	1,34	4,20	3,31	21226	0,31	4,10	1,83	0,09
<i>kroA100</i>	8002	8012	0,07	0,12	1,86	8002	0,13	0,00	3,12	0,12
<i>kroB100</i> 0,2	7300	7300	0,33	0,00	1,82	7300	0,65	0,00	2,49	0,00
<i>bier127</i>	12395	12395	0,30	0,00	1,94	12395	0,08	0,00	3,02	0,00
<i>pr152</i>	22511	26421	0,00	17,37	2,25	26421	0,00	17,37	4,21	0,00
<i>gr202</i>	14724	14784	0,12	0,41	2,67	14735	0,34	0,07	5,71	0,33
<i>tsp225</i>	3732	3763	0,36	0,83	2,71	3763	0,33	0,83	7,55	0,00
<i>a280</i>	2562	2586	0,16	0,94	3,54	2591	0,40	1,13	11,65	-0,19
<i>lin318</i>	18155	20790	0,33	14,51	32,55	19763	1,21	8,86	13,23	5,20
<i>gr431</i>	27817	28031	0,21	0,77	6,09	28067	0,54	0,90	24,08	-0,13
<i>burma14</i>	1438	1438	0,00	0,00	1,12	1438	0,00	0,00	0,59	0,00
<i>ulysses22</i>	2144	2144	0,00	0,00	1,06	2144	0,00	0,00	1,09	0,00
<i>berlin52</i>	3147	3147	0,17	0,00	2,00	3147	0,17	0,00	3,79	0,00
<i>st70</i>	665	665	0,00	0,00	9,37	665	0,00	0,00	5,11	0,00
<i>pr76</i>	39144	39614	1,65	1,20	8,24	39144	2,45	0,00	6,8	1,20
<i>kroA100</i>	10813	10842	2,55	0,27	6,58	10931	2,53	1,09	9,19	-0,81
<i>kroB100</i> 0,5	11020	11276	1,63	2,32	3,10	11252	1,27	2,11	11,8	0,21
<i>bier127</i>	26688	27621	1,28	3,50	4,63	27572	1,00	3,31	15,56	0,18
<i>pr152</i>	38350	39727	13,06	3,59	5,22	39859	1,90	3,93	34,21	-0,33
<i>gr202</i>	18386	18610	0,39	1,22	18,62	18770	0,36	2,09	33,8	-0,85
<i>tsp225</i>	3719	3743	0,59	0,65	9,34	3739	0,89	0,54	42,02	0,11
<i>a280</i>	2566	2560	0,79	-0,23	14,01	2574	0,71	0,31	66,08	-0,54
<i>lin318</i>	23381	24042	0,30	2,83	17,72	23726	0,97	1,48	82,57	1,33
<i>gr431</i>	44515	46175	3,47	3,73	69,06	47088	3,26	5,78	484,08	-1,94
<i>burma14</i>	2026	2026	0,00	0,00	1,22	2026	0,00	0,00	1,07	0,00
<i>ulysses22</i>	3436	3436	0,00	0,00	1,61	3436	0,00	0,00	1,7	0,00
<i>berlin52</i>	5322	5322	1,31	0,00	2,74	5322	1,08	0,00	13,6	0,00
<i>st70</i>	665	665	0,00	0,00	11,04	665	0,00	0,00	11,32	0,00
<i>pr76</i>	64205	67069	1,51	4,46	8,55	64880	3,22	1,05	13,36	3,37
<i>kroA100</i>	15242	15518	0,36	1,81	5,56	15415	0,69	1,14	21,72	0,67
<i>kroB100</i> 0,8	15400	15479	0,43	0,51	5,12	15537	0,18	0,89	21,64	-0,37
<i>bier127</i>	55731	57524	1,86	3,22	11,56	57351	1,20	2,91	38,13	0,30
<i>pr152</i>	60445	62197	0,70	2,90	9,89	61858	0,80	2,34	58,96	0,55

Tabela 61 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-SA. (cont.)

Exemplar	CS-SA _H					CS-CS _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>gr202</i>	24119	24291	0,93	0,71	17,15	24187	2,06	0,28	161,2	0,43
<i>tsp225</i>	3723	3738	0,58	0,40	20,67	3731	0,87	0,21	103,96	0,19
<i>a280</i>	2558	2578	0,20	0,78	32,43	2593	0,15	1,37	167,16	-0,58
<i>lin318</i>	31166	31687	0,96	1,67	39,60	31869	0,80	2,26	221,95	-0,57
<i>gr431</i>	82458	84804	0,69	2,85	469,77	84375	1,47	2,32	1261,21	0,51
Média			0,92	1,85	20,73		0,76	1,63	70,81	0,20

Tabela 62 - PCTSP: $\gamma_i \in [1, 1000]$ - Resultados computacionais do CS-SA.

Exemplar	CS-SA _H					CS-SA _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>burma14</i>	2313	2313	0,00	0,00	0,88	2313	0,00	0,00	0,68	0,00
<i>ulysses22</i>	4866	4866	0,00	0,00	1,26	4866	0,00	0,00	0,75	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	1,65	7374	0,00	0,00	1,30	0,00
<i>st70</i>	675	675	0,00	0,00	1,83	675	0,00	0,00	1,59	0,00
<i>pr76</i>	45853	45853	0,00	0,00	1,73	45853	0,00	0,00	1,99	0,00
<i>kroA100</i>	20079	20079	0,00	0,00	1,81	20079	0,00	0,00	2,30	0,00
<i>kroB100</i>	0,2 20442	20442	0,07	0,00	1,68	20449	0,12	0,03	7,56	-0,03
<i>bier127</i>	52763	52763	0,00	0,00	2,04	52763	0,00	0,00	4,95	0,00
<i>pr152</i>	62448	62498	0,01	0,08	7,76	62460	0,05	0,02	4,08	0,06
<i>gr202</i>	32913	33135	0,26	0,67	2,61	33185	0,34	0,83	38,15	-0,15
<i>tsp225</i>	3939	4007	0,20	1,73	2,97	3988	0,78	1,24	7,82	0,48
<i>a280</i>	2586	2604	0,84	0,70	157,37	2595	1,24	0,35	12,48	0,35
<i>lin318</i>	40543	40593	0,31	0,12	4,61	40747	0,41	0,50	16,08	-0,38
<i>gr431</i>	113316	113823	0,25	0,45	6,50	114373	0,17	0,93	27,66	-0,48
<i>burma14</i>	2313	2313	0,00	0,00	1,53	2313	0,00	0,00	0,86	0,00
<i>ulysses22</i>	4866	4866	0,00	0,00	1,67	4866	0,00	0,00	1,11	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	2,05	7374	0,00	0,00	3,17	0,00
<i>st70</i>	675	675	0,00	0,00	2,48	675	0,00	0,00	5,08	0,00
<i>pr76</i>	57252	57252	0,20	0,00	2,62	57252	0,06	0,00	6	0,00
<i>kroA100</i>	20079	20079	0,00	0,00	3,12	20079	0,00	0,00	9,32	0,00
<i>kroB100</i>	0,5 20442	20442	0,09	0,00	22,82	20442	0,12	0,00	20,73	0,00
<i>bier127</i>	54325	54523	0,11	0,36	19,21	54480	0,06	0,29	14,2	0,08
<i>pr152</i>	65328	67124	2,06	2,75	23,13	65328	4,55	0,00	44,47	2,75
<i>gr202</i>	32813	33153	0,10	1,04	8,01	33119	0,41	0,93	34,34	0,10
<i>tsp225</i>	3925	3972	0,62	1,20	9,72	3974	0,93	1,25	41,78	-0,05
<i>a280</i>	2587	2616	0,21	1,12	13,24	2627	0,56	1,55	66,45	-0,42
<i>lin318</i>	40410	40635	0,19	0,56	16,33	40687	0,55	0,69	84,71	-0,13
<i>gr431</i>	113549	114077	0,10	0,46	32,13	114113	0,12	0,50	161,91	-0,03
<i>burma14</i>	2321	2321	0,00	0,00	1,59	2321	0,00	0,00	1,13	0,00
<i>ulysses22</i>	4887	4887	0,00	0,00	1,73	4887	0,00	0,00	1,72	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	2,79	7374	0,00	0,00	6,44	0,00
<i>st70</i>	675	675	0,00	0,00	3,55	675	0,00	0,00	11,28	0,00
<i>pr76</i>	73272	73272	1,26	0,00	3,91	73809	0,62	0,73	18,13	-0,73
<i>kroA100</i>	20079	20079	0,00	0,00	5,17	20079	0,00	0,00	21,7	0,00
<i>kroB100</i>	0,8 20442	20442	0,06	0,00	5,51	20442	0,09	0,00	24,8	0,00
<i>bier127</i>	68902	68902	1,46	0,00	7,32	69123	1,89	0,32	41,64	-0,32
<i>pr152</i>	69516	69554	0,11	0,05	9,54	69554	0,14	0,05	51,22	0,00
<i>gr202</i>	32867	32967	0,18	0,30	31,23	33012	0,53	0,44	83,46	-0,14
<i>tsp225</i>	3952	3981	0,32	0,73	51,07	3976	0,76	0,61	181,52	0,13
<i>a280</i>	2588	2600	0,65	0,46	30,35	2611	0,72	0,89	170,28	-0,42
<i>lin318</i>	40265	40754	0,27	1,21	42,24	40686	0,43	1,05	215,42	0,17
<i>gr431</i>	115874	116330	0,30	0,39	75,29	116165	0,40	0,25	439,67	0,14
Média			0,24	0,34	14,86		0,38	0,32	45,00	0,02

Tabela 63 - PCTSP: $\gamma_i \in [1, 10000]$ - Resultados computacionais do CS-SA.

Exemplar	CS-SA _H					CS-SA _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>burma14</i>	3050	3050	0,00	0,00	1,50	3050	0,00	0,00	0,71	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	1,68	7013	0,00	0,00	0,78	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	1,78	7542	0,00	0,00	1,32	0,00
<i>st70</i>	675	675	0,00	0,00	1,57	675	0,00	0,00	1,61	0,00
<i>pr76</i>	97269	97269	0,00	0,00	8,41	97269	0,00	0,00	3,8	0,00
<i>kroA100</i>	21282	21282	0,00	0,00	2,14	21282	0,01	0,00	2,39	0,00
<i>kroB100</i>	0,2 21697	21697	0,15	0,00	1,84	21697	0,07	0,00	7,73	0,00
<i>bier127</i>	108605	108691	0,14	0,08	1,93	108617	0,20	0,01	17,7	0,07
<i>pr152</i>	73682	73682	0,08	0,00	7,97	73682	0,13	0,00	4,09	0,00
<i>gr202</i>	38743	38965	0,49	0,57	2,73	39252	0,13	1,31	6,48	-0,73
<i>tsp225</i>	3942	3978	0,58	0,91	2,96	3984	0,55	1,07	8,03	-0,15
<i>a280</i>	2592	2606	0,25	0,54	3,99	2614	0,34	0,85	469,07	-0,31
<i>lin318</i>	42361	42855	0,37	1,17	4,04	42717	0,66	0,84	15,93	0,32
<i>gr431</i>	168632	169628	0,40	0,59	222,90	170061	0,43	0,85	30,22	-0,25
<i>burma14</i>	3050	3050	0,00	0,00	1,37	3050	0,00	0,00	0,83	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	1,88	7013	0,00	0,00	1,14	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	2,12	7542	0,00	0,00	3,24	0,00
<i>st70</i>	675	675	0,00	0,00	2,44	675	0,00	0,00	5	0,00
<i>pr76</i>	97269	97269	0,03	0,00	2,57	97269	0,00	0,00	7,73	0,00
<i>kroA100</i>	21282	21282	0,01	0,00	3,11	21282	0,01	0,00	9,29	0,00
<i>kroB100</i>	0,5 21697	21697	0,13	0,00	3,06	21697	0,12	0,00	9,28	0,00
<i>bier127</i>	108605	108605	0,14	0,00	3,80	108605	0,11	0,00	14,36	0,00
<i>pr152</i>	73682	73682	0,16	0,00	5,15	73682	0,17	0,00	27,59	0,00
<i>gr202</i>	38882	38974	0,30	0,24	42,40	39018	0,48	0,35	423,93	-0,11
<i>tsp225</i>	3952	3972	0,68	0,51	10,12	4018	0,29	1,67	43,1	-1,14
<i>a280</i>	2593	2598	0,71	0,19	13,07	2631	0,18	1,47	66,68	-1,25
<i>lin318</i>	42484	42721	0,15	0,56	17,24	42682	0,37	0,47	85,15	0,09
<i>gr431</i>	168221	169729	0,33	0,90	231,06	169928	0,56	1,01	161,58	-0,12
<i>burma14</i>	3050	3050	0,00	0,00	1,56	3050	0,00	0,00	8,92	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	1,79	7013	0,00	0,00	14,37	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	2,83	7542	0,00	0,00	63	0,00
<i>st70</i>	675	675	0,00	0,00	2,87	675	0,00	0,00	116,13	0,00
<i>pr76</i>	97269	97269	0,00	0,00	3,75	97269	0,00	0,00	116,53	0,00
<i>kroA100</i>	21282	21282	0,00	0,00	5,31	21282	0,01	0,00	249,36	0,00
<i>kroB100</i>	0,8 21697	21697	0,05	0,00	5,02	21697	0,13	0,00	258,97	0,00
<i>bier127</i>	108605	108605	0,18	0,00	7,40	108649	0,09	0,04	394,04	-0,04
<i>pr152</i>	73682	73682	0,05	0,00	57,14	73682	0,07	0,00	620,3	0,00
<i>gr202</i>	38855	39100	0,17	0,63	15,71	38938	0,50	0,21	1116,35	0,42
<i>tsp225</i>	3955	3981	0,50	0,66	20,36	3973	1,02	0,46	1493,97	0,20
<i>a280</i>	2593	2614	0,35	0,81	32,49	2621	0,47	1,08	2615,69	-0,27
<i>lin318</i>	42373	42719	0,28	0,82	140,13	42721	0,57	0,82	3709,51	0,00
<i>gr431</i>	168765	169773	0,50	0,60	72,41	170165	0,47	0,83	7739,28	-0,23
Média			0,17	0,23	23,18		0,19	0,32	474,89	-0,08

C.2 META-HEURÍSTICA CS-GA

Tabela 64 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-GA.

Exemplar	CS-GA _H					CS-GA _C (CHAVES, 2009)					
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta	
<i>burma14</i>	797	797	0,00	0,00	0,00	797	0,00	0,00	0,00	0,00	
<i>ulysses22</i>	1699	1699	0,00	0,00	0,45	1699	0,00	0,00	0,57	0,00	
<i>berlin52</i>	2189	2189	0,00	0,00	0,56	2189	0,00	0,00	1,19	0,00	
<i>st70</i>	665	665	0,00	0,00	4,02	665	0,00	0,00	24,03	0,00	
<i>pr76</i>	20390	20390	2,52	0,00	5,25	20390	3,59	0,00	6,76	0,00	
<i>kroA100</i>	8002	8002	0,02	0,00	7,96	8002	0,11	0,00	7,87	0,00	
<i>kroB100</i>	0,2	7300	7300	0,00	0,00	10,66	7300	0,52	0,00	12,28	0,00
<i>bier127</i>	12395	12395	0,00	0,00	14,02	12395	1,79	0,00	21,68	0,00	
<i>pr152</i>	22511	26078	0,42	15,85	16,75	26421	1,85	17,37	41,51	-1,30	
<i>gr202</i>	14724	14724	0,05	0,00	31,88	14759	0,11	0,24	72,75	-0,24	
<i>tsp225</i>	3732	3720	0,18	-0,32	449,57	3736	0,47	0,11	518,52	-0,43	
<i>a280</i>	2562	2552	0,25	-0,39	888,10	2567	0,37	0,20	1106,11	-0,58	
<i>lin318</i>	18155	19245	2,91	6,00	114,26	20232	1,98	11,44	248,68	-4,88	
<i>gr431</i>	27817	27881	0,14	0,23	178,13	28082	0,85	0,95	335,81	-0,72	
<i>burma14</i>	1438	1438	0,00	0,00	0,98	1438	0,10	0,00	1,90	0,00	
<i>ulysses22</i>	2144	2144	0,00	0,00	0,50	2144	0,00	0,00	1,04	0,00	
<i>berlin52</i>	3147	3147	0,00	0,00	7,86	3147	0,23	0,00	10,42	0,00	
<i>st70</i>	665	665	0,00	0,00	5,79	665	0,00	0,00	16,19	0,00	
<i>pr76</i>	39144	39207	0,99	0,16	16,38	39551	2,50	1,04	22,49	-0,87	
<i>kroA100</i>	10813	10813	0,25	0,00	27,01	10842	2,20	0,27	46,57	-0,27	
<i>kroB100</i>	0,5	11020	11080	0,56	0,54	31,36	11206	0,93	1,69	39,38	-1,12
<i>bier127</i>	26688	27181	0,72	1,85	43,93	27583	1,29	3,35	76,82	-1,46	
<i>pr152</i>	38350	38917	11,46	1,48	63,02	46181	1,86	20,42	126,14	-15,73	
<i>gr202</i>	18386	18327	0,07	-0,32	127,00	18590	0,79	1,11	249,26	-1,41	
<i>tsp225</i>	3719	3719	0,15	0,00	494,18	3726	0,96	0,19	608,47	-0,19	
<i>a280</i>	2566	2548	0,31	-0,70	944,91	2578	0,43	0,47	1470,72	-1,16	
<i>lin318</i>	23381	23214	0,71	-0,71	412,47	23601	1,57	0,94	763,47	-1,64	
<i>gr431</i>	44515	46309	0,90	4,03	749,14	49136	5,29	10,38	1457,02	-5,75	
<i>burma14</i>	2026	2026	0,00	0,00	0,45	2026	0,00	0,00	1,04	0,00	
<i>ulysses22</i>	3436	3436	0,00	0,00	0,72	3436	0,00	0,00	1,70	0,00	
<i>berlin52</i>	5322	5322	0,00	0,00	13,89	5380	1,51	1,09	21,09	-1,08	
<i>st70</i>	665	665	0,00	0,00	7,10	665	0,00	0,00	10,88	0,00	
<i>pr76</i>	64205	64880	1,56	1,05	24,18	66149	1,30	3,03	41,09	-1,92	
<i>kroA100</i>	15242	15294	0,29	0,34	48,87	15385	0,80	0,94	94,14	-0,59	
<i>kroB100</i>	0,8	15400	15479	0,05	0,51	40,54	15505	0,35	0,68	82,91	-0,17
<i>bier127</i>	55731	55720	0,46	-0,02	88,30	56962	1,59	2,21	185,99	-2,18	
<i>pr152</i>	60445	61122	0,53	1,12	122,81	61389	1,31	1,56	247,17	-0,43	
<i>gr202</i>	24119	24043	0,60	-0,32	273,45	24586	0,96	1,94	554,72	-2,21	
<i>tsp225</i>	3723	3715	0,27	-0,21	524,44	3737	0,53	0,38	785,55	-0,59	
<i>a280</i>	2558	2557	0,13	-0,04	967,47	2579	0,33	0,82	1886,86	-0,85	
<i>lin318</i>	31166	31081	0,65	-0,27	933,06	31696	0,66	1,70	1255,60	-1,94	

Tabela 64 - PCTSP: $\gamma_i \in [1, 100]$ - Resultados computacionais do CS-GA. (cont.)

Exemplar	CS-GA _H				CS-GA _C (CHAVES, 2009)					
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>gr431</i>	82458	82734	0,27	0,33	1988,00	84558	1,38	2,55	3542,18	-2,16
Média			0,65	0,72	230,42		0,96	2,07	380,92	-1,23

Tabela 65 - PCTSP: $\gamma_i \in [1, 1000]$ - Resultados computacionais do CS-GA.

Exemplar	CS-GA _H					CS-GA _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>burma14</i>	2313	2313	0,00	0,00	0,34	2313	0,00	0,00	0,63	0,00
<i>ulysses22</i>	4866	4866	0,00	0,00	0,44	4866	0,00	0,00	0,72	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	1,48	7374	0,00	0,00	3,59	0,00
<i>st70</i>	675	675	0,00	0,00	3,79	675	0,00	0,00	8,15	0,00
<i>pr76</i>	45853	45853	0,00	0,00	3,35	45853	0,00	0,00	6,04	0,00
<i>kroA100</i>	20079	20079	0,00	0,00	2,26	20079	0,00	0,00	5,32	0,00
<i>kroB100</i>	0,2 20442	20442	0,00	0,00	35,65	20442	0,04	0,00	43,71	0,00
<i>bier127</i>	52763	52763	0,00	0,00	19,06	52763	0,04	0,00	34,50	0,00
<i>pr152</i>	62448	62460	0,01	0,02	129,02	62460	0,12	0,02	135,41	0,00
<i>gr202</i>	32913	32840	0,08	-0,22	352,29	32931	0,44	0,05	437,83	-0,28
<i>tsp225</i>	3939	3935	0,13	-0,10	531,56	3958	0,24	0,48	678,82	-0,58
<i>a280</i>	2586	2586	0,12	0,00	929,13	2606	0,40	0,77	1448,89	-0,77
<i>lin318</i>	40543	40205	0,33	-0,83	1318,27	40543	0,26	0,00	2623,29	-0,83
<i>gr431</i>	113316	113447	0,15	0,12	1783,51	113316	0,72	0,00	139,04	0,12
<i>burma14</i>	2313	2313	0,00	0,00	0,38	2313	0,00	0,00	0,78	0,00
<i>ulysses22</i>	4866	4866	0,00	0,00	0,92	4866	0,00	0,00	1,30	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	2,20	7374	0,00	0,00	5,84	0,00
<i>st70</i>	675	675	0,00	0,00	5,25	675	0,00	0,00	19,58	0,00
<i>pr76</i>	57252	57252	0,00	0,00	15,75	57252	0,06	0,00	19,66	0,00
<i>kroA100</i>	20079	20079	0,00	0,00	3,91	20079	0,00	0,00	15,43	0,00
<i>kroB100</i>	0,5 20442	20442	0,00	0,00	49,69	20442	0,01	0,00	92,54	0,00
<i>bier127</i>	54325	54480	0,00	0,29	45,18	54505	0,22	0,33	98,44	-0,05
<i>pr152</i>	65328	68213	0,03	4,42	138,53	68227	1,09	4,44	267,04	-0,02
<i>gr202</i>	32813	32857	0,08	0,13	361,30	32914	0,79	0,31	498,02	-0,17
<i>tsp225</i>	3925	3925	0,51	0,00	601,01	3964	0,67	0,99	762,94	-0,98
<i>a280</i>	2587	2589	0,09	0,08	1052,12	2605	0,60	0,70	1096,65	-0,61
<i>lin318</i>	40410	40254	0,23	-0,39	1421,28	40635	0,41	0,56	2192,55	-0,94
<i>gr431</i>	113549	113412	0,06	-0,12	2086,90	114231	0,31	0,60	2885,57	-0,72
<i>burma14</i>	2321	2321	0,00	0,00	0,49	2321	0,00	0,00	1,02	0,00
<i>ulysses22</i>	4887	4887	0,00	0,00	0,59	4887	0,00	0,00	1,63	0,00
<i>berlin52</i>	7374	7374	0,00	0,00	2,39	7374	0,00	0,00	6,89	0,00
<i>st70</i>	675	675	0,00	0,00	9,88	675	0,00	0,00	31,83	0,00
<i>pr76</i>	73272	73272	0,00	0,00	30,22	73272	1,09	0,00	56,40	0,00
<i>kroA100</i>	20079	20079	0,00	0,00	6,83	20079	0,00	0,00	26,87	0,00
<i>kroB100</i>	0,8 20442	20442	0,00	0,00	50,78	20442	0,04	0,00	91,50	0,00
<i>bier127</i>	68902	68902	0,13	0,00	103,78	69472	1,52	0,83	169,26	-0,82
<i>pr152</i>	69516	69416	0,12	-0,14	128,37	69554	0,19	0,05	316,17	-0,20
<i>gr202</i>	32867	32805	0,09	-0,19	375,06	33145	0,10	0,85	546,40	-1,03
<i>tsp225</i>	3952	3932	0,17	-0,51	646,36	3990	0,81	0,96	865,55	-1,45
<i>a280</i>	2588	2588	0,12	0,00	1081,64	2607	0,31	0,73	1601,10	-0,73
<i>lin318</i>	40265	40319	0,13	0,13	1425,23	40466	0,86	0,50	2124,90	-0,36
<i>gr431</i>	115874	115059	0,18	-0,70	2547,57	115874	0,54	0,00	3993,92	-0,70
Média			0,07	0,05	411,98		0,28	0,31	556,09	-0,26

Tabela 66 - PCTSP: $\gamma_i \in [1; 10000]$ - Resultados computacionais do CS-GA.

Exemplar	CS-GA _H					CS-GA _C (CHAVES, 2009)				
	melhor	sol	desvio	gap	t	sol	desvio	gap	t	delta
<i>burma14</i>	3050	3050	0,00	0,00	0,33	3050	0,00	0,00	0,65	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	0,41	7013	0,00	0,00	0,71	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	0,80	7542	0,00	0,00	1,43	0,00
<i>st70</i>	675	675	0,00	0,00	3,73	675	0,00	0,00	10,78	0,00
<i>pr76</i>	97269	97269	0,00	0,00	7,30	97269	0,00	0,00	9,05	0,00
<i>kroA100</i>	21282	21282	0,00	0,00	17,55	21282	0,00	0,00	35,91	0,00
<i>kroB100</i>	0,2 21697	21697	0,00	0,00	42,85	21697	0,05	0,00	69,91	0,00
<i>bier127</i>	108605	108605	0,00	0,00	78,50	108605	0,02	0,00	111,88	0,00
<i>pr152</i>	73682	73682	0,00	0,00	85,24	73682	0,02	0,00	161,44	0,00
<i>gr202</i>	38743	38733	0,05	-0,03	316,01	38777	0,65	0,09	424,03	-0,11
<i>tsp225</i>	3942	3933	0,35	-0,23	464,64	3966	0,56	0,61	690,95	-0,83
<i>a280</i>	2592	2585	0,26	-0,27	885,27	2593	0,54	0,04	1262,84	-0,31
<i>lin318</i>	42361	42300	0,12	-0,14	1440,70	42361	0,78	0,00	1723,13	-0,14
<i>gr431</i>	168632	168580	0,10	-0,03	3840,40	169204	0,58	0,34	3614,12	-0,37
<i>burma14</i>	3050	3050	0,00	0,00	0,39	3050	0,00	0,00	0,80	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	0,49	7013	0,00	0,00	1,07	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	1,14	7542	0,00	0,00	3,53	0,00
<i>st70</i>	675	675	0,00	0,00	6,04	675	0,00	0,00	19,21	0,00
<i>pr76</i>	97269	97269	0,00	0,00	10,42	97269	0,00	0,00	26,97	0,00
<i>kroA100</i>	21282	21282	0,00	0,00	22,37	21282	0,00	0,00	53,92	0,00
<i>kroB100</i>	0,5 21697	21697	0,00	0,00	48,96	21697	0,08	0,00	110,80	0,00
<i>bier127</i>	108605	108605	0,00	0,00	104,78	108605	0,10	0,00	135,03	0,00
<i>pr152</i>	73682	73682	0,00	0,00	119,64	73682	0,11	0,00	277,02	0,00
<i>gr202</i>	38882	38807	0,08	-0,19	403,38	39038	0,37	0,40	573,93	-0,59
<i>tsp225</i>	3952	3938	0,23	-0,35	557,81	3985	0,24	0,84	779,16	-1,18
<i>a280</i>	2593	2589	0,10	-0,15	1129,16	2602	0,87	0,35	1501,67	-0,50
<i>lin318</i>	42484	42293	0,23	-0,45	1437,81	42701	0,20	0,51	2010,47	-0,96
<i>gr431</i>	168221	168835	0,15	0,36	3902,90	169244	0,64	0,61	5525,47	-0,24
<i>burma14</i>	3050	3050	0,00	0,00	0,46	3050	0,00	0,00	1,03	0,00
<i>ulysses22</i>	7013	7013	0,00	0,00	0,58	7013	0,00	0,00	1,61	0,00
<i>berlin52</i>	7542	7542	0,00	0,00	1,59	7542	0,00	0,00	6,64	0,00
<i>st70</i>	675	675	0,00	0,00	8,86	675	0,00	0,00	30,52	0,00
<i>pr76</i>	97269	97269	0,00	0,00	16,74	97269	0,02	0,00	59,82	0,00
<i>kroA100</i>	21282	21282	0,00	0,00	30,73	21282	0,00	0,00	104,09	0,00
<i>kroB100</i>	0,8 21697	21697	0,00	0,00	64,65	21697	0,02	0,00	114,25	0,00
<i>bier127</i>	108605	108605	0,00	0,00	133,30	108605	0,14	0,00	215,44	0,00
<i>pr152</i>	73682	73682	0,00	0,00	137,75	73682	0,04	0,00	303,38	0,00
<i>gr202</i>	38855	38742	0,15	-0,29	430,30	39014	0,11	0,41	640,13	-0,70
<i>tsp225</i>	3955	3926	0,37	-0,73	596,11	3970	0,66	0,38	828,02	-1,11
<i>a280</i>	2593	2590	0,08	-0,12	1076,81	2621	0,12	1,08	1363,08	-1,18
<i>lin318</i>	42373	42217	0,32	-0,37	1529,14	42430	0,78	0,13	2429,68	-0,50
<i>gr431</i>	168765	168246	0,47	-0,31	3839,72	168879	0,99	0,07	4466,44	-0,37
Média			0,07	-0,08	542,76		0,21	0,14	707,14	-0,22