Ana Rita Tiradentes Terra Argoud

PROCEDIMENTO PARA PROJETO DE ARRANJO
FÍSICO MODULAR EM MANUFATURA ATRAVÉS DE
ALGORITMO GENÉTICO DE AGRUPAMENTO

São Carlos

2007

Ana Rita Tiradentes Terra Argoud

PROCEDIMENTO PARA PROJETO DE ARRANJO
FÍSICO MODULAR EM MANUFATURA ATRAVÉS DE
ALGORITMO GENÉTICO DE AGRUPAMENTO

Tese apresentada à Escola de Engenharia de
São Carlos da Universidade de São Paulo,
como parte dos requisitos para a obtenção do
Título de Doutora em Engenharia Mecânica,
área de concentração Manufatura.

Orientador: Prof. Dr. Eduardo Vila Gonçalves Filho

São Carlos

2007

Ao Daniel, meu marido, com admiração e gratidão por sua compreensão e apoio durante a elaboração deste trabalho. Aos meus pais Evalda e Fábio pelo incentivo constante, e à minha amada filhinha Luana.

AGRADECIMENTOS

Ao Prof. Dr. Eduardo Vila Gonçalves Filho, por sua atenção e dedicação durante o processo de orientação deste trabalho.

À Escola de Engenharia de São Carlos, e ao Departamento de Engenharia Mecânica pela oportunidade de realização do curso de doutorado.

Em especial ao colega Alexandre José Tiberti pela ajuda e enriquecimento deste trabalho.

Aos professores com os quais tive a oportunidade de aprofundar meu conhecimento através das disciplinas cursadas: Eduardo Vila Gonçalves Filho, Arthur José Vieira Porto, Ruth de Gouvêa Duarte, Antônio Freitas Rentes, Antônio Francisco do Prado.

Aos funcionários do Departamento de Engenharia Mecânica e da Biblioteca Central e em especial à Elizabeth Maria A. Alexandre e Ana Paula Bueno Bardelotte.

A todos os colegas do Laboratório de Simulação com os quais tive a oportunidade de conviver e em especial à Mariella Florenzano Souza, Mamoro José Yamada, José Hamilton Chaves Gorgulho Júnior, Marcos Roberto Fortulan, Edwin Cardoza, Shih Yung Chin, João Gilberto Zalla Filho, Anselmo Ramalho Pitombeira Neto, Anna Cristina Barbosa Dias de Carvalho, Creusa Sayuri Tahara, Felipe Alves Cavani, Hilano José Rocha de Carvalho, e Robson Rogério Dutra Pereira.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPq, pela concessão da bolsa de estudo para realização desta pesquisa.

RESUMO

ARGOUD, A.R.T.T. (2007). **Procedimento para projeto de arranjo físico modular em manufatura através de algoritmo genético de agrupamento**. Tese (Doutorado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2007.

O objetivo desta tese foi desenvolver um procedimento para projeto de arranjo físico modular. O procedimento usa um algoritmo genético de agrupamento para gerar módulos de arranjo físico em manufatura. Um módulo é um grupo de máquinas com um fluxo característico. O algoritmo genético de agrupamento (AGA) permite ao usuário especificar *a priori* o número desejado de módulos (problema de K-agrupamentos), e também trabalhar de forma que o número e formação dos módulos seja uma otimização do problema (problema de agrupamento automático). Uma característica importante do AGA é sua flexibilidade, pois fornece possibilidade ao usuário de dirigir seu raciocínio através da escolha de diferentes codificações do cromossomo, de funções de aptidão para avaliar a qualidade da solução, de medidas de similaridade para comparação de seqüências de operações e de métodos de geração da nova população. Para tal foram desenvolvidos três operadores de cruzamento e quatro operadores de mutação. Os experimentos permitiram avaliar o comportamento do AGA com diferentes operadores e configurações de parâmetros, através do número de vezes em que a melhor solução foi encontrada. Os módulos de arranjo físico gerados pelo AGA e o arranjo físico final da fábrica foram comparados aqueles da principal referência encontrada na bibliografia em projeto de arranjo físico modular. No arranjo físico final da fábrica, elaborado a partir das soluções do AGA, houve importante redução da distância total percorrida pelo conjunto de peças, demonstrando a eficácia do algoritmo genético de agrupamento. Dessa forma, os resultados apontaram o algoritmo genético de agrupamento como uma alternativa à geração de módulos de arranjo físico em projeto de arranjo físico modular.

Palavras-chave: Projeto de arranjo físico. Arranjo físico modular. Algoritmos genéticos. Agrupamento. Manufatura.

ABSTRACT

ARGOUD, A.R.T.T. (2007). **Procedure to the design of modular facility layouts through clustering genetic algorithm**. Ph.D. Thesis - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2007.

The objective of this thesis was the development of a procedure to the design of modular facility layouts. The procedure uses a clustering genetic algorithm to generate layout modules. A module is a group of machines with a specific patterned flow. The clustering genetic algorithm (CGA) allows the specification of the desired number of modules (K-grouping problem) and also the optimization of the number of modules (automatic clustering problem). The genetic algorithm is very flexible and the user can guide his reasoning. It supplies different options of chromosome encoding, generation of the initial population, fitness function to evaluate the solution quality, different similarity measures for comparison of operation sequences and methods for generation of the new population. Three crossover operators and four mutation operators have been developed. The experiments have allowed evaluating the behavior of the CGA with different operators and parameters configurations, through the number of times in which the best solution was found. The layout modules generated by the CGA and the block layout were compared with the main reference which was found in literature in the design of modular facility layouts. In the block layout elaborated from the solutions of the CGA there was an important reduction of the total distance traveled by the set of parts, demonstrating the effectiveness of the clustering genetic algorithm. The results indicated that the clustering genetic algorithm is an alternative for generation of layout modules in the design of modular facility layouts.

Keywords: Facility layout design. Modular layout. Manufacturing. Genetic algorithms. Clustering.

LISTA DE FIGURAS

Figura 1.1 - Fluxograma de desenvolvimento da pesquisa	11
Figura 2.1 - Arranjo físico posicional	15
Figura 2.2 - Arranjo físico por produto.....	15
Figura 2.3 - Arranjo físico funcional.....	16
Figura 2.4 - Arranjo físico celular.....	17
Figura 2.5 - Classificação dos arranjos físicos tradicionais segundo volume-variedade..	20
Figura 2.6 - Exemplos de arranjo físico distribuído.....	24
Figura 2.7 - Arranjo físico celular com três células.....	25
Figura 2.8 - Arranjo físico celular com reorientação das células.....	26
Figura 2.9 - Arranjo físico celular com reorientação e mudança na forma das células....	26
Figura 2.10 - Arranjo físico celular com linhas em forma de S, em paralelo.....	27
Figura 2.11 - Arranjo físico <i>flowshop</i> híbrido.....	27
Figura 2.12 - Arranjo físico celular virtual.....	28
Figura 2.13 - Arranjo físico celular com uma célula remanescente.....	29
Figura 2.14 - Células em cascata.....	29
Figura 2.15 - Exemplo de arranjo físico fractal.....	30
Figura 2.16 - Tipos de módulos.....	33
Figura 2.17 - Algoritmo para determinar as subsequências comuns.....	39
Figura 2.18 - Classificação dos arranjos físicos tradicionais e não tradicionais.....	45
Figura 3.1 - Princípio de trabalho de um algoritmo genético.....	48
Figura 3.2 – Principais elementos de um algoritmo genético.....	49

Figura 3.3 - Ordenação no <i>grid</i>	52
Figura 3.4 - Ordenação no <i>grid</i> com separadores.....	52
Figura 3.5 - Cruzamento de ponto único.....	55
Figura 3.6 - Cruzamento de dois pontos.....	55
Figura 3.7 - Cruzamento uniforme.....	56
Figura 3.8 - Mutação.....	56
Figura 3.9 - Roleta.....	57
Figura 3.10 - Arranjo físico de bloco com baias flexíveis.....	71
Figura 4.1 - Fluxograma do algoritmo genético de agrupamento (AGA).....	80
Figura 5.1 - Tela de entrada com seqüência de operações.....	111
Figura 5.2 - Confirmação para encontrar subsequências comuns únicas.....	112
Figura 5.3 - Subseqüências comuns únicas.....	113
Figura 5.4 - Seleção do coeficiente de similaridade.....	114
Figura 5.5 - Matriz de similaridade.....	115
Figura 5.6 - Seleção dos parâmetros.....	116
Figura 5.7 - Estatísticas: valores de aptidão.....	118
Figura 5.8 - Estatísticas: posições selecionadas.....	120
Figura 5.9 - População final: grupos formados.....	121
Figura 5.10 - População final: cromossomo codificado.....	122
Figura 5.11 - População inicial.....	123
Figura 5.12 - Teste 1: subsequências comuns únicas.....	125
Figura 5.13 - Teste 1: matriz de similaridade.....	125
Figura 5.14 - Estatísticas: teste 1.....	127

Figura 5.15 - População inicial: teste 1, codificação R2.....	128
Figura 5.16 - População final: teste 1.....	129
Figura 5.17 - População inicial: teste 1, codificação R3.....	130
Figura 5.18 - População final: teste 2, largura da silhueta média.....	131
Figura 5.19 - Parâmetros para teste 2: $4 \leq K \leq 17$, soma das similaridades médias.....	132
Figura 5.20 - Parâmetros para teste 2: $4 \leq K \leq 17$, taxa do critério da variância.....	132
Figura 5.21 - População final para teste 2: $4 \leq K \leq 17$	133
Figura 5.22 - Matriz de similaridade: teste 3.....	135
Figura 5.23 - População final: teste 3, largura da silhueta média.....	136
Figura 5.24 - Cromossomo na 45ª posição com S7, S8, S9 e S10 agrupadas.....	137
Figura 5.25 - População final: teste 3, soma das similaridades médias, S5, S10 e S11 isoladas.....	138
Figura 5.26 - População final: teste 3, soma das similaridades médias, S5 e S11 agrupadas.....	139
Figura 5.27 - População final: teste 3, soma das similaridades médias, S10 e S11 agrupadas.....	140
Figura 5.28 - Cromossomo com maior aptidão: teste 3, taxa do critério da variância...	141
Figura 5.29 - Cromossomo com maior aptidão e agrupamentos diferentes: teste 3, taxa do critério da variância.....	142
Figura 5.30 - Matriz de similaridade: teste 4.....	145
Figura 5.31 - População final: teste 4, largura da silhueta média.....	146
Figura 5.32 – Teste de Validação 1: cromossomo na 200ª posição.....	153
Figura 5.33 - Teste de Validação 2: solução idêntica a de Huang (2003).....	155
Figura 6.1 - Valores de aptidão: experimento 1.....	161
Figura 6.2 - Valores de aptidão: experimento 2.....	163
Figura 6.3 - Valores de aptidão: experimento 3.....	166
Figura 6.4 - Valores de aptidão: experimento 4.....	168
Figura 6.5 - Valores de aptidão: experimento 5.....	170

Figura 6.6 - Experimentos combinados, número de grupos fixo, método probabilístico: cruzamento Recombinação com Correção Aleatória e mutação <i>Move</i> Aleatório.....	174
Figura 6.7 - Experimentos combinados, número de grupos fixo, método probabilístico: cruzamento Ponto Único de Grupo e mutação <i>Move</i> Aleatório.....	176
Figura 6.8 - Experimentos combinados, número de grupos fixo, método faixa percentual: cruzamento Recombinação com Correção Aleatória e mutação <i>Move</i> Aleatório.....	179
Figura 6.9 - Experimentos combinados, número de grupos fixo, método faixa percentual: cruzamento Ponto Único de Grupo e mutação <i>Move</i> Aleatório.....	180
Figura 6.10 - Experimentos combinados, número de grupos variável, método probabilístico: cruzamento Recombinação com Correção Aleatória e mutação <i>Move</i> Aleatório.....	185
Figura 6.11 - Experimentos combinados, número de grupos variável, método probabilístico: cruzamento Ponto Único de Grupo e mutação <i>Move</i> Aleatório.....	187
Figura 6.12 - Experimentos combinados, número de grupos variável, método probabilístico: 0,85 de cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3.....	192
Figura 6.13 - Experimentos combinados, número de grupos variável, método probabilístico: 0,50 de cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3.....	194
Figura 6.14 - Experimentos combinados, número de grupos variável, método probabilístico: 0,85 de cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3.....	196
Figura 6.15 - Experimentos combinados, número de grupos variável, método probabilístico: 0,50 de cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3.....	198
Figura 6.16 - Experimentos combinados, número de grupos variável, método faixa percentual: cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3.....	202
Figura 6.17 - Experimentos combinados, número de grupos variável, método faixa percentual: cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3.	204
Figura 7.1 - Diagrama de fluxo do arranjo físico modular: número de grupos fixo.....	215
Figura 7.2 - Arranjo físico de bloco: número de grupos fixo.....	218
Figura 7.3 - <i>Grid</i> de arranjo físico: número de grupos fixo.....	220
Figura 7.4 - Arranjo físico de bloco.....	222
Figura 7.5 - <i>Grid</i> de arranjo físico para a solução de Huang (2003).....	223
Figura 7.6 - Arranjo físico de bloco sem duplicação de máquinas.....	227

Figura 7.7 - Diagrama de fluxo do arranjo físico modular: número de grupos variável	237
Figura 7.8 - Arranjo físico de bloco: número de grupos variável.....	240
Figura 7.9 - <i>Grid</i> de arranjo físico: número de grupos variável.....	241
Figura 7.10 - Arranjo físico: número de grupos variável com alternativas à duplicação.....	244
Figura 7.11 - <i>Grid</i> de arranjo físico: número de grupos variável com alternativas à duplicação.....	245
Figura 8.1 - Matriz de similaridade.....	254
Figura 8.2 - Parâmetros do AGA.....	255
Figura 8.3 - Estatísticas: valores de aptidão.....	256
Figura 8.4 - População final.....	257
Figura 8.5 - Diagrama de fluxo do arranjo físico modular: número de grupos fixo.....	263
Figura 8.6 - Solução de Huang (2003): cromossomo na 109ª posição.....	265
Figura 8.7 - Parâmetros do AGA: número de grupos variável.....	267
Figura 8.8 - Estatísticas: valores de aptidão, número de grupos variável.....	268
Figura 8.9 - População final, número de grupos variável	269
Figura 8.10 - Diagrama de fluxo do arranjo físico modular: número de grupos variável.....	276
Figura C.1 - Fluxo de informações.....	302
Figura C.2 - Algoritmo MST aplicado ao conjunto de dados.....	303

LISTA DE TABELAS

Tabela 2.1 - Exemplo para identificação de subsequência comum.....	40
Tabela 3.1 - População inicial.....	58
Tabela 4.1 - Interpretação da silhueta média $\overline{s(i)}$	89
Tabela 5.1 - Sequência de operações: teste 1.....	124
Tabela 5.2 - Parâmetros: teste 1.....	126
Tabela 5.3 - Sequência de operações: teste 3.....	134
Tabela 5.4 - Subseqüências comuns únicas: teste 3.....	134
Tabela 5.5 - Subseqüências comuns únicas: teste 4.....	144
Tabela 5.6 - Sequência de operações: teste de validação 1.....	148
Tabela 5.7 - Subseqüências comuns únicas.....	149
Tabela 5.8 - Módulos básicos de <i>layout</i>	150
Tabela 5.9 - Teste de Validação 1: módulos gerados.....	151
Tabela 5.10 - Teste de Validação 2: módulos gerados.....	154
 Tabela 6.1 - Experimentos preliminares: método probabilístico.....	 158
Tabela 6.2 - Experimentos preliminares: método faixa percentual.....	158
Tabela 6.3 - Parâmetros: experimento 1.....	160
Tabela 6.4 - Parâmetros: experimento 2.....	162
Tabela 6.5 - Parâmetros: experimento 3.....	165
Tabela 6.6 - Parâmetros: experimento 4.....	167
Tabela 6.7 - Parâmetros: experimento 5.....	169
Tabela 6.8 - Experimentos combinados: método probabilístico, número de grupos fixo.....	172

Tabela 6.9 - Experimentos combinados: método faixa percentual, número de grupos fixo.....	178
Tabela 6.10 - Experimentos combinados: número de grupos variável.....	183
Tabela 6.11 - Experimentos combinados: número de grupos variável, método probabilístico.....	190
Tabela 6.12 - Experimentos combinados: número de grupos variável, método faixa percentual.....	200
Tabela 7.1 - Seqüência de operações e tempos de processamento.....	208
Tabela 7.2 - Subseqüências comuns únicas.....	209
Tabela 7.3 - Módulos gerados: número de grupos fixo.....	210
Tabela 7.4 - Índice de semelhança: número de grupos fixo.....	210
Tabela 7.5 - Seqüência de operações designadas aos módulos: número de grupos fixo.....	212
Tabela 7.6 - Módulos <i>layout</i> ajustados: número de grupos fixo.....	213
Tabela 7.7 - Seqüência de operações ajustadas aos módulos: número de grupos fixo..	214
Tabela 7.8 - Número necessário de máquinas: número de grupos fixo.....	217
Tabela 7.9 - Distância percorrida por cada peça: número de grupos fixo.....	221
Tabela 7.10 - Distância percorrida por cada peça: solução de Huang (2003).....	224
Tabela 7.11 - Distância percorrida por cada peça: sem duplicação de máquinas.....	228
Tabela 7.12 - Número necessário de máquinas: número de grupos fixo, sem duplicação.....	229
Tabela 7.13 - Alteração nos módulos de <i>layout</i> : número de grupos fixo.....	230
Tabela 7.14 - Módulos gerados: número de grupos variável.....	232
Tabela 7.15 - Índice de semelhança: número de grupos variável.....	233
Tabela 7.16 - Módulos básicos de <i>layout</i> agregados.....	233
Tabela 7.17 - Seqüência de operações designadas aos módulos: número de grupos variável.....	234
Tabela 7.18 - Módulos <i>layout</i> ajustados: número de grupos variável.....	235

Tabela 7.19 - Seqüência de operações ajustadas aos módulos: número de grupos variável.....	236
Tabela 7.20 - Número necessário de máquinas: número de grupos variável.....	238
Tabela 7.21 - Distância percorrida por cada peça: número de grupos variável.....	242
Tabela 7.22 - Distância percorrida: número de grupos variável com alternativa à duplicação.....	246
Tabela 7.23 - Número necessário de máquinas: número de grupos variável com alternativas à duplicação.....	247
Tabela 7.24 - Alteração nos módulos de <i>layout</i> : número de grupos variável.....	249
Tabela 8.1 - Seqüência de operações.....	252
Tabela 8.2 - Subseqüências comuns únicas.....	253
Tabela 8.3 - Módulos básicos: número de grupos fixo.....	258
Tabela 8.4 - Índice de semelhança: número de grupos fixo.....	258
Tabela 8.5 - Módulos agregados: número de grupos fixo.....	259
Tabela 8.6 - Seqüência de operações designadas aos módulos: número de grupos fixo.....	260
Tabela 8.7 - Módulos <i>layout</i> ajustados: número de grupos fixo.....	261
Tabela 8.8 - Seqüência de operações ajustadas aos módulos: número de grupos fixo..	262
Tabela 8.9 - Módulos básicos de arranjo físico.....	264
Tabela 8.10 - Módulos básicos: número de grupos variável.....	270
Tabela 8.11 - Índice de semelhança: número de grupos variável.....	271
Tabela 8.12 - Módulos agregados: número de grupos variável.....	272
Tabela 8.13 - Seqüência de operações designadas aos módulos: número de grupos variável.....	273
Tabela 8.14 - Módulos <i>layout</i> ajustados: número de grupos variável.....	274
Tabela 8.15 - Seqüência de operações ajustadas aos módulos: número de grupos variável.....	275
Tabela C.1 - Seqüência de operações e demanda.....	300

LISTA DE QUADROS

Quadro 2.1 - Vantagens e limitações dos tipos tradicionais de arranjo físico.....	21
Quadro 2.2 - Vantagens e limitações dos tipos não tradicionais de arranjo físico.....	44
Quadro 3.1 - Analogia de termos entre algoritmos genéticos e a biologia.....	48
Quadro 3.2 - Principais características dos artigos revisados.....	73
Quadro 4.1 - Pseudo-código de um algoritmo genético típico.....	77
Quadro 4.2 - Algoritmo genético de agrupamento (AGA).....	78
Quadro C.1 - Algoritmo MST.....	299
Quadro C.2 - Matriz De-Para.....	301
Quadro C.3 - Matriz De-Para diagonalizada.....	301

LISTA DE ABREVIATURAS

ACCA	<i>Adjacent Cluster Checking Algorithm</i> (Algoritmo para verificação de agrupamentos adjacentes)
AG	Algoritmo Genético
AGA	Algoritmo Genético de Agrupamento
CDA	<i>Cluster Decomposition Algorithm</i> (Algoritmo para decomposição de agrupamentos)
DOM	<i>Dominant common subsequence</i> (Subseqüência comum dominante)
FA	Função de Aptidão
FMS	<i>Flexible Manufacturing System</i> (Sistema Flexível de Manufatura)
HCMA	<i>Hierarchical Cluster Merging Algorithm</i> (Algoritmo para fusão de agrupamentos hierárquicos)
HCL	<i>Hybrid Cellular Layout</i> (Arranjo Físico Celular Híbrido)
LCS	<i>Longest Common Subsequence</i> (Mais longa subseqüência comum)
MST	<i>Maximum Spanning Tree</i> (Árvore de Medida Máxima)
VRC	<i>Variance Ratio Criterion</i> (Taxa do Critério da Variância)
USA	<i>United States of América</i> (Estados Unidos da América)
WIP	<i>Working In Process</i> (Estoque em Processo)

SUMÁRIO

1 INTRODUÇÃO	1
1.1 Considerações iniciais.....	1
1.2 Motivação e justificativa.....	2
1.3 Objetivo.....	7
1.4 Contribuições.....	7
1.5 Metodologia de pesquisa.....	9
1.5.1 Metodologia de pesquisa adotada.....	10
1.6 Estrutura.....	12
 2 ARRANJO FÍSICO DA FÁBRICA	 14
2.1 Considerações iniciais.....	14
2.2 Tipos tradicionais de arranjo físico.....	14
2.2.1 Arranjo físico posicional ou de posições fixas.....	14
2.2.2 Arranjo físico por produto.....	15
2.2.3 Arranjo físico funcional ou por processo.....	16
2.2.4 Arranjo físico celular.....	16
2.2.4.1 Medidas de similaridade.....	17
2.2.5 Comparação entre os tipos tradicionais de arranjo físico.....	20
2.3 Tipos não tradicionais de arranjos físicos.....	22
2.3.1 Arranjo físico ágil.....	22
2.3.2 Arranjo físico distribuído.....	23
2.3.3 Arranjo físico celular híbrido.....	25

2.3.4 Arranjo físico fractal.....	30
2.3.5 Arranjo físico modular.....	31
2.3.5.1 Similaridade entre seqüências de operações.....	34
2.3.5.2 Procedimento para projeto de arranjo físico modular.....	39
2.3.6 Comparação entre os tipos não tradicionais de arranjo físico.....	44
2.4 Considerações finais.....	45
 3 ALGORITMOS GENÉTICOS	 47
3.1 Considerações iniciais.....	47
3.2 Codificação do cromossomo.....	50
3.3 Geração da população inicial.....	52
3.4 Função de aptidão.....	53
3.5 Operadores genéticos.....	54
3.5.1 Operador de cruzamento.....	54
3.5.2 Operador de mutação.....	56
3.6 Processo de seleção.....	56
3.7 Geração da nova população.....	59
3.8 Critérios de parada.....	60
3.9 Parâmetros.....	60
3.10 Algoritmos genéticos aplicados ao projeto de arranjo físico.....	61
3.11 Considerações finais.....	76
 4 O MODELO DESENVOLVIDO	 77
4.1 Considerações iniciais.....	77

4.2 O algoritmo genético de agrupamento (AGA) desenvolvido.....	78
4.3 Codificação do cromossomo.....	81
4.4 Geração da população inicial.....	82
4.5 Função de aptidão.....	85
4.5.1 Soma das similaridades médias.....	85
4.5.2 Largura da silhueta média.....	87
4.5.3 Taxa do critério da variância.....	89
4.6 Operadores dos cromossomos.....	91
4.6.1 Operadores de cruzamento.....	92
4.6.1.1 Recombinação.....	92
4.6.1.2 Recombinação com correção aleatória.....	96
4.6.1.3 Ponto único de grupo.....	97
4.6.2 Operadores de mutação.....	101
4.7 Processo de seleção e geração da nova população.....	103
4.8 Critério de parada.....	108
4.9 Considerações finais.....	109
 5 IMPLEMENTAÇÃO DO ALGORITMO GENÉTICO DE AGRUPAMENTO	110
5.1 Considerações iniciais.....	110
5.2 Dados de entrada.....	110
5.3 Geração das subsequências comuns únicas.....	112
5.4 Geração da matriz de similaridade.....	113
5.5 Definição dos parâmetros.....	115
5.6 Estatísticas geradas.....	117
5.7 Visualização das populações final e inicial.....	120

5.8 Verificação do algoritmo genético de agrupamento.....	123
5.9 Validação do algoritmo genético de agrupamento.....	147
5.10 Considerações finais.....	155
6 EXPERIMENTAÇÃO	156
6.1 Considerações iniciais.....	156
6.2 Experimentos preliminares.....	157
6.3 Novos operadores de cruzamento e mutação 3.....	163
6.4 Experimentos combinados: número de grupos fixo.....	170
6.4.1 Método probabilístico.....	171
6.4.2 Método faixa percentual.....	178
6.4.3 Comentários: número de grupos fixo.....	181
6.5 Experimentos combinados: número de grupos variável.....	182
6.5.1 Número de grupos variável: sem operadores de mutação 1 e 2.....	183
6.5.2 Número de grupos variável: todos os operadores.....	189
6.5.2.1 Método probabilístico.....	189
6.5.2.2 Método faixa percentual.....	199
6.5.3 Comentários: número de grupos variável.....	205
7 ARRANJO FÍSICO FINAL	207
7.1 Considerações iniciais.....	207
7.2 Arranjo físico final: número de grupos fixo.....	209
7.2.1 Análise da distância percorrida: número de grupos fixo.....	219
7.2.2 Caminhos para reduzir a duplicação de máquinas.....	225
7.3 Arranjo físico final: número de grupos variável.....	232

7.3.1 Análise da distância percorrida: número de grupos variável.....	240
7.3.2 Caminhos para reduzir a duplicação de máquinas: número de grupos variável	243
8 ESTUDO DE CASO	251
8.1 Considerações iniciais.....	251
8.2 Número de grupos fixo.....	254
8.2.1 Comparação dos módulos de arranjo físico.....	264
8.3 Número de grupos variável.....	266
8.3.1 Comentários: número de grupos variável.....	277
9 CONCLUSÃO	278
9.1 Sugestões para trabalhos futuros.....	285
REFERÊNCIAS	288
APÊNDICE A – Exemplo de arquivo de entrada do AGA	296
APÊNDICE B – Exemplo de arquivo de saída do AGA	297
APÊNDICE C – <i>Maximum spanning tree</i>	299

1 INTRODUÇÃO

1.1 Considerações iniciais

O ambiente de manufatura tem se transformado constantemente nas últimas décadas principalmente em relação às constantes alterações no *mix* de produtos e na demanda. Alguns fatores foram determinantes nessa transformação, como o aumento da competição global, consumidores mais conscientes e exigentes em relação a custos, qualidade, prazos, lançamentos de novos produtos, e a rápida mudança tecnológica (KOCHHAR; HERAGU, 1999). Dessa forma, a manufatura precisou se tornar mais dinâmica e fornecer respostas rápidas a tais transformações.

O projeto do arranjo físico em manufatura trata da disposição física das máquinas, equipamentos, áreas de suporte à produção e áreas para pessoal (TOMPKINS et al., 2003). O correto projeto do arranjo físico tem papel decisivo para a sobrevivência e sucesso de uma empresa nesse ambiente dinâmico, pois permite a racionalização do espaço, minimização da movimentação de materiais e de pessoas, levando à redução de custos e aumento da eficiência do sistema produtivo.

Segundo Benjaafar, Heragu e Irani (2002), alguns setores da indústria têm demonstrado que as configurações de arranjos físicos tradicionais como funcional, celular e em linha, não atendem às necessidades de empresas que trabalham com vários tipos de produtos com demanda variável e curto ciclo de vida. As empresas precisam de arranjos flexíveis, modulares e de fácil configuração. Como consequência desta necessidade surgiram novos tipos de arranjos físicos como o ágil, distribuído, fractal e modular.

Esses novos tipos de arranjo físico têm despertado grande interesse entre os pesquisadores, seja em metodologia de projeto, em formas de avaliação de desempenho ou em

comparações com arranjos físicos tradicionais, entretanto ainda é uma área recente e um campo vasto para pesquisa.

1.2 Motivação e justificativa

Dentre os novos tipos de arranjos físicos, o arranjo modular despertou especial interesse, pois surge como uma nova forma de se pensar o projeto de arranjo físico.

A prática tradicional da indústria tem sido usar uma amostra de produtos para projetar o tipo de arranjo físico da fábrica. Por exemplo, para produtos padronizados e com grande volume de demanda usa-se arranjo físico em linha, para produtos com características similares de projeto e processo constrói-se o arranjo físico celular, e produtos com baixo volume de demanda e sem similaridade aparente são processados em arranjo físico funcional. Entretanto, uma amostra de peças não é suficiente para representar uma fábrica que possua *mix* variável de produção.

O arranjo físico modular não assume *a priori* padrões de fluxo e critérios para agrupamento de máquinas, como nos arranjos celular, funcional e em linha. Os padrões de fluxo e agrupamentos de máquinas são conseqüências dos fluxos implícitos nas seqüências de operações. Assim, o arranjo físico modular identifica os padrões de fluxo existentes na fábrica e gera um arranjo, chamado módulo de *layout*, para cada fluxo de materiais. Uma das vantagens do arranjo físico modular é possibilitar o encadeamento de operações e com isso diminuir a movimentação de peças na fábrica.

O primeiro trabalho encontrado sobre arranjo físico modular foi o de Shahrukh A. Irani e Heng Huang apresentado em 1998 (IRANI; HUANG, 1998). O grupo liderado por estes dois pesquisadores foi o único encontrado na bibliografia que trabalha com projeto de arranjo físico modular. Em continuidade ao primeiro trabalho, diversos artigos foram

publicados, dentre eles, Irani e Huang (2000a e 2000b), Huang (2003) e Irani e Huang (2006). A tese de doutoramento de Huang (2003), na qual o tema é bem detalhado, recebeu especial atenção desta autora e foi fonte de motivação para o presente trabalho.

Huang (2003) propôs um procedimento para projeto de arranjo físico modular baseado em análise de agrupamento (*cluster analysis*) e comparação de seqüências de operações. Para tal usou um método de agrupamento hierárquico aglomerativo com método da média das distâncias para regra de ligação (*unweighted pair-group average linkage*).

Métodos de agrupamento hierárquico consistem em uma série de sucessivas fusões ou divisões do conjunto de dados. Os métodos hierárquicos aglomerativos iniciam com cada elemento em um grupo, separadamente. Baseado em uma medida de similaridade, os elementos são agrupados, de forma que os grupos mais similares são agrupados a cada estágio, até que se forme apenas um grupo. Os métodos hierárquicos divisivos trabalham de modo inverso. Iniciam com todos os elementos em um único grupo, e a cada passo são formados subgrupos, baseado na dissimilaridade entre os elementos, até que cada objeto tenha seu próprio grupo. O resultado dos métodos hierárquicos é mostrado em um dendrograma (KAUFMAN; ROUSSEEUW, 1990; COLE, 1998). O dendrograma é um gráfico em forma de árvore que mostra o agrupamento de objetos em níveis conforme o grau de semelhança entre eles. Bussab, Miazaki e Andrade (1990) explicam detalhadamente como construir um dendrograma.

Os métodos de agrupamento hierárquico aglomerativos mais conhecidos são: *single-linkage*, *complete-linkage*, *average-linkage* e método da variância mínima de Ward. Esses métodos são detalhados em Everitt, Landau e Leese (2001).

O método aglomerativo *average linkage*, usado por Huang (2003), define a distância entre dois grupos como a distância média de todos os elementos de um grupo para com todos os elementos do outro grupo. Esta abordagem tende a combinar grupos com pequena

variância, e favorece a produção de grupos com variância aproximadamente igual, ou seja, agrupamentos esféricos (COLE, 1998; EVERITT; LANDAU; LEESE, 2001).

Segundo Cole (1998), métodos de agrupamento hierárquico possuem algumas desvantagens:

- Eles são indicados para pequenos conjuntos de dados, devido à necessidade de armazenar e atualizar constantemente matrizes de similaridade;
- Se um elemento for alocado incorretamente a um grupo em um estágio inicial, ele permanecerá no mesmo grupo até o término do processo. Não há chance para que o elemento procure pelo grupo correto;
- Os resultados refletem o grau com que os dados se ajustam à estrutura formal do algoritmo.

De acordo com Everitt, Landau e Leese (2001), o mesmo conjunto de dados pode produzir resultados muito diferentes, conforme o método de agrupamento hierárquico usado.

Para aplicação do método proposto por Huang (2003), o usuário deve especificar o número desejado de módulos, ou seja, o número de grupos (K) é conhecido, e é usado como parâmetro para se encontrar a solução. Entretanto, quando se define o número de grupos, podem ser gerados agrupamentos artificiais. Geralmente, o algoritmo é rodado várias vezes, para depois escolher a solução que pareça melhor, ou que tenha a interpretação mais significativa. Para isso deve-se ter conhecimento do domínio do problema. Dessa forma, esses algoritmos realizam busca pela solução em um espaço relativamente pequeno, e, portanto, suas possibilidades de sucesso são reduzidas (HRUSCHKA, 2001).

No projeto do arranjo físico modular são inúmeras as possibilidades de realizar o agrupamento de máquinas em módulos a partir das seqüências de operações. Este é um problema de agrupamento de natureza combinatorial.

Resolver um problema de agrupamento consiste em agrupar elementos de um conjunto de dados de maneira tal que cada elemento de um grupo possua maior similaridade com qualquer elemento de seu grupo do que com elementos de outros grupos. O objetivo é encontrar estruturas “naturais” dentro do conjunto de dados. O problema de agrupamento, de modo formal, é definido como (COLE, 1998):

Seja um conjunto de n elementos $X = \{ X_1, X_2, \dots, X_n \}$. Os elementos devem ser agrupados em grupos não sobrepostos $G = \{ G_1, G_2, \dots, G_k \}$, onde K é o número de grupos, tal que $G_1 \cup G_2 \cup \dots \cup G_k = X$, $G_i \neq \emptyset$, e $G_i \cap G_j = \emptyset$ para $i \neq j$.

O número de grupos, K , pode ser conhecido ou não. Quando o valor de K é conhecido, ou seja, é fornecido pelo usuário como parâmetro para se encontrar a solução, o problema é chamado de problema de K -agrupamentos. Quando o número de grupos é desconhecido, e a obtenção do valor de K faz parte da solução do problema, o mesmo é chamado de problema de agrupamento automático.

No primeiro caso, o número de possibilidades de se classificar n elementos em K grupos é dado por (LIU apud HRUSCKHA, 2001):

$$N(n, k) = \frac{1}{k!} \sum_{i=0}^k (-1)^i \binom{k}{i} (k-i)^n \quad (1.1)$$

Considerando a equação 1.1, para agrupar 25 elementos em 5 grupos existem $N(25,5) = 2.436.684.974.110.751$ maneiras. A partir deste número pode-se ver a complexidade em se agrupar, por exemplo, 25 máquinas em 5 grupos, ou módulos, problema considerado relativamente pequeno para projeto de arranjo físico.

O tamanho do problema cresce consideravelmente quando o número de grupos, K , é desconhecido. Determinar o melhor número de grupos, é uma das tarefas mais difíceis em problemas de agrupamento. O número de combinações possíveis para se classificar n elementos é dado por (LIU apud HRUSCKHA, 2001):

$$N(n) = \sum_{i=1}^n \frac{1}{k!} \sum_{i=0}^k (-1)^i \binom{k}{i} (k-i)^n \quad (1.2)$$

Portanto, existem 4×10^{18} possibilidades de se agrupar 25 máquinas com número de módulos variante, entre 1 e 25. Este problema pertence à classe NP-completo e é inviável do ponto de vista computacional para um algoritmo realizar busca exaustiva visando encontrar solução ótima global (COLE, 1998).

Vários algoritmos e técnicas têm sido propostos como ferramentas de auxílio ao projeto de arranjo físico, visto que é um problema de otimização combinatória (SINGH; RAJAMANI, 1996). Dentre eles destacam-se aplicações com algoritmos genéticos, *simulated annealing*, redes neurais artificiais e busca tabu.

Algoritmos genéticos apresentam algumas vantagens para este tipo de problema, pois (KOCHHAR; FOSTER; HERAGU, 1998):

- Trabalham com várias possíveis soluções em paralelo, o que permite explorar vasto campo de soluções;
- Fornecem a possibilidade de escapar de ótimos locais devido à atuação do operador de mutação;
- Permitem a redução rápida do espaço de busca em virtude do mecanismo de evolução a partir das melhores soluções.

Além dessas vantagens, algoritmos genéticos fornecem flexibilidade à criação de codificações de cromossomos e operadores genéticos sensíveis ao contexto do problema.

1.3 Objetivo

O objetivo deste trabalho foi desenvolver um procedimento para projeto de arranjo físico modular. O procedimento usa um algoritmo genético de agrupamento (AGA) para gerar os módulos de arranjo físico. O termo arranjo físico refere-se à disposição das máquinas no chão-de-fábrica.

O algoritmo genético de agrupamento (AGA) é altamente flexível pois possibilita ao usuário dirigir seu próprio raciocínio através da escolha de diferentes codificações do cromossomo, de funções de aptidão, de medidas de similaridade e de métodos de geração da nova população.

1.4 Contribuições

Na literatura pesquisada não foi encontrado nenhum trabalho em que se fez a aplicação de algoritmos genéticos ao projeto de arranjo físico modular. A seguir são apresentadas, por ordem de relevância, conforme o entendimento desta autora, as principais contribuições desta tese. Em seguida à contribuição está indicado entre colchetes a seção em que o leitor poderá ler a respeito.

1. Flexibilidade do algoritmo genético de agrupamento (AGA) pois permite a seleção de uma entre três medidas de similaridade para comparação de seqüências de operações [seções 2.3.5.1 e 5.4], e possibilita a escolha da forma da geração da população inicial, conforme três diferentes tipos de codificações do cromossomo [seções 4.3 e 5.5]. O usuário também deve escolher uma entre três funções de aptidão para avaliar a qualidade da solução [seções 4.5 e 5.5] e um entre dois métodos de geração da nova população [seções 4.7 e 5.5].

2. O AGA resolve problemas tanto de K-agrupamentos como de agrupamento automático. No primeiro, o usuário especifica a priori o número desejado de módulos [seções 6.4, 7.2 e 8.2]. No segundo, o número e formação dos módulos é uma otimização do problema [seções 6.5, 7.3 e 8.3].
3. Modelagem do algoritmo genético através de subsequência de operações comuns a duas ou mais peças [seção 4.3].
4. Desenvolvimento do operador de cruzamento Ponto Único de Grupo [seção 4.6.1.3].
5. Criação do operador de mutação *Move* Direcionado [seção 4.6.2].
6. Aplicação e validação da função largura da silhueta média para agrupamento de máquinas em projeto de arranjo físico modular [seções 4.5.2, 5.8 e 5.9].
7. A introdução no operador Recombinação do mecanismo de correção dos descendentes para evitar cromossomos inválidos [seção 4.6.1.1].
8. Desenvolvimento do operador de cruzamento Recombinação com Correção Aleatória [seção 4.6.1.2].
9. Modificação do operador de cruzamento Recombinação proposto por Hruschka (2001) para adaptação ao problema de agrupamento de subsequências de operações de peças [seção 4.6.1.1].
10. Modificação dos operadores de mutação *Merge* e *Break* propostos por Hruschka (2001) através de diferente forma de inserção dos objetos e de escolha das sementes, respectivamente [seção 4.6.2].
11. Modificação do operador de mutação *Move* proposto por Cole (1998) [seção 4.6.2].

1.5 Metodologia de pesquisa

Segundo Marconi e Lakatos (1985), a pesquisa é um procedimento formal que requer tratamento científico e é o caminho para se conhecer a realidade ou descobrir verdades parciais.

Para Trujillo apud Marconi e Lakatos (1985), a pesquisa tem duas finalidades, pura ou prática. A pesquisa pura gera novos conhecimentos através do desenvolvimento da metodologia, obtenção de diagnósticos e estudos aprimorados dos fenômenos. Cita como exemplo a teoria da relatividade. A pesquisa prática é aplicada com determinado objetivo prático, como a aplicação da energia nuclear.

Esta mesma classificação de pesquisa é apresentada por Ander-Egg apud Marconi e Lakatos (1985). A pesquisa pura ou fundamental visa ampliar o conhecimento teórico sem preocupação de utilização prática. Procura formular princípios, leis e generalizações. Na pesquisa aplicada os resultados são usados imediatamente na solução de problemas que ocorrem na realidade.

Além das classificações em pura e aplicada, Best apud Marconi e Lakatos (1985) acrescentam: pesquisa história, descritiva e experimental. A pesquisa histórica procura a investigação, registro, análise e interpretação de fatos ocorridos no passado para, através de generalizações, compreender o presente e prever o futuro. A pesquisa descritiva aborda a descrição, registro, análise e interpretação de fenômenos atuais, visando o seu funcionamento no presente. Na pesquisa experimental há controle sobre determinados fatores e sua importância está nas relações de causa e efeito.

De acordo com Cervo e Bervian (1996), as pesquisas podem ser classificadas em bibliográfica, experimental e descritiva. A pesquisa bibliográfica procura conhecer e analisar as contribuições culturais ou científicas já existentes sobre determinado assunto ou problema.

Pode ser realizada independentemente ou como parte da pesquisa descritiva ou experimental. A pesquisa experimental manipula diretamente as variáveis relacionadas ao objeto de estudo. A partir dessa manipulação possibilita o estudo das relações entre causas e efeitos de determinado fenômeno. A pesquisa descritiva observa, registra, analisa e correlaciona fenômenos sem manipulá-los. Segundo aquele autor, a pesquisa descritiva, muito usada em Ciências Humanas e Sociais pode assumir várias formas, entre elas: estudos exploratórios, descritivos, pesquisa de opinião, pesquisa de motivação, estudo de caso e pesquisa documental.

1.5.1 Metodologia de pesquisa adotada

Segundo a classificação dos tipos de pesquisa apresentada no item anterior, esta tese é de natureza aplicada com caráter experimental. Para seu desenvolvimento faz uso de pesquisa bibliográfica.

A figura 1.1 a seguir apresenta um fluxograma com as etapas do desenvolvimento desta pesquisa.

Figura 1.1 – Fluxograma de desenvolvimento da pesquisa

A etapa de implementação computacional do algoritmo genético ocorreu à medida que houve o refinamento do procedimento para geração dos módulos de arranjo físico com a incorporação de novas funcionalidades no *software*, como o desenvolvimento de novos operadores genéticos e funções para avaliação da solução. A redação foi realizada durante todo o desenvolvimento do trabalho. A etapa de redação final refere-se à verificação e fechamento do texto.

1.6 Estrutura

Esta tese está estruturada em nove capítulos, da seguinte maneira:

Capítulo 2 – Arranjo físico da fábrica. Este capítulo apresenta os principais tipos de arranjos físicos presentes no ambiente de manufatura. São abordados os tipos tradicionais de arranjos e novas tendências, conhecidas como arranjos físicos não tradicionais. Este capítulo também detalha os principais conceitos sobre projeto de arranjo físico modular.

Capítulo 3 – Algoritmos genéticos. Este capítulo faz uma introdução à teoria sobre algoritmos genéticos e apresenta a revisão bibliográfica sobre aplicação dos mesmos ao projeto de arranjo físico em manufatura.

Capítulo 4 – O modelo desenvolvido. Este capítulo apresenta o modelo de algoritmo genético desenvolvido para geração de módulos de *layout* em projeto de arranjo físico modular. Os principais elementos do algoritmo genético são detalhados como a codificação do cromossomo, função de aptidão, população inicial, processo de seleção, operadores genéticos e critério de parada.

Capítulo 5 – Implementação do algoritmo genético de agrupamento. Este capítulo aborda os detalhes da implementação do algoritmo genético, a interface com o usuário e os testes de verificação e de validação do mesmo.

Capítulo 6 – Experimentação. Este capítulo apresenta as várias rodadas de simulação realizadas sobre determinado conjunto de dados com o objetivo de avaliar o comportamento do algoritmo genético de agrupamento. Também é analisado o desempenho dos operadores genéticos de cruzamento e mutação desenvolvidos.

Capítulo 7 – Arranjo físico final. Este capítulo apresenta o arranjo físico final da fábrica, elaborado a partir dos módulos de *layout* gerados pelo algoritmo genético de agrupamento. Também é feita a análise do resultado obtido e comparação com uma referência da literatura.

Capítulo 8 – Estudo de caso. Este capítulo apresenta a aplicação do procedimento para geração de módulos através do algoritmo genético de agrupamento em um segundo conjunto de dados.

Capítulo 9 – Conclusões. Este capítulo apresenta as conclusões relativas ao objetivo inicialmente traçado, sumariza as principais contribuições e sugere futuros trabalhos sobre o assunto.

2 ARRANJO FÍSICO DA FÁBRICA

2.1 Considerações iniciais

Um arranjo físico de manufatura trata da disposição das máquinas e equipamentos no chão de fábrica (TOMPKINS et al., 2003). Este capítulo apresenta os tipos tradicionais de arranjos físicos e novas tendências de arranjos que têm surgido no ambiente industrial. Dentro dos novos tipos de arranjos físicos, conhecidos como não tradicionais, será tratado o arranjo físico modular, tema deste trabalho.

2.2 Tipos tradicionais de arranjo físico

A literatura apresenta diversos tipos de arranjos físicos, entretanto, quatro deles estão consolidados no ambiente industrial. São eles: arranjo físico posicional, por produto, funcional e celular (TOMPKINS et al., 2003; SINGH; RAJAMANI, 1996). Todos apresentam vantagens e limitações.

2.2.1 Arranjo físico posicional ou de posições fixas

O produto permanece fixo, enquanto os recursos como mão-de-obra, máquinas, e ferramentas se movem até ele para realizar o processamento (figura 2.1). O arranjo físico é flexível, pois pode acomodar mudanças no projeto. As habilidades requeridas são altas e o trabalho é feito em times, o que fornece oportunidade de enriquecimento do trabalho. A demanda é baixa ou esporádica e o produto é difícil de ser movimentado. Este é o caso da construção de edifícios, navios, aviões e equipamentos de grande porte.

Figura 2.1 – Arranjo físico posicional

2.2.2 Arranjo físico por produto

As máquinas são arranjadas de acordo com a sequência de operações do produto (figura 2.2). Isto facilita o controle do processo e o manuseio de materiais. O arranjo físico por produto requer equipamentos específicos de alto investimento. Alterações no produto implicam em mudança na disposição das máquinas. Esta é uma das razões da sua baixa flexibilidade. Este arranjo não necessita de mão-de-obra altamente qualificada e a maioria das tarefas é simples e repetitiva. A movimentação de materiais é reduzida, os *leadtimes* são curtos e o inventário baixo. Entretanto, os equipamentos requerem alta confiabilidade, pois a falha em uma estação pode causar parada de toda a linha. A linha de montagem é o exemplo mais típico deste tipo de arranjo físico.

Figura 2.2 – Arranjo físico por produto

2.2.3 Arranjo físico funcional ou por processo

As máquinas são arranjadas de acordo com a sua função (figura 2.3). Por exemplo, em uma usinagem há o setor de tornos, o de fresas, o de furadeiras, entre outros. A maioria dos equipamentos é de propósito geral. Requer grande habilidade da mão-de-obra devido à variedade de tarefas a serem desempenhadas. A flexibilidade é alta, mas a eficiência baixa. Os investimentos são menores e a utilização é maior. Uma peça visita diversos setores da fábrica. O fluxo de materiais não é suave, como no arranjo físico por produto. Tipicamente há muito fluxo entre os setores e pouco fluxo dentro de cada setor. O sistema de produção é mais complicado e decisões operacionais como programação e carga são importantes e difíceis. Os níveis de estoque e os *leadtimes* são maiores.

Figura 2.3 – Arranjo físico funcional

2.2.4 Arranjo físico celular

Baseia-se no agrupamento de peças em famílias. Uma família possui peças com características de projeto e/ou processo similares como forma, composição do material, ferramental, controles. No arranjo físico celular as máquinas agrupadas são capazes de realizar um conjunto de operações diferentes (figura 2.4), mas suficiente para produzir uma gama de peças completas na saída. Tipicamente há muito fluxo dentro da célula e pouco fluxo entre as células.

Figura 2.4 – Arranjo físico celular

Para realizar um agrupamento, seja de peças em famílias, de máquinas, ou mesmo de outros conjuntos de dados, alguns algoritmos definem uma medida de similaridade para gerar a matriz de similaridade/dissimilaridade. A partir da matriz é que se torna evidente a existência ou não dos grupos. A seguir são apresentadas algumas medidas de similaridade usadas em problemas de agrupamento e mais especificamente em projeto de arranjo físico.

2.2.4.1 Medidas de similaridade

A medida de similaridade avalia quão perto ou longe está um elemento de outro em um conjunto de dados. A literatura (KAUFMAN; ROUSSEEUW, 1990; COLE, 1998; EVERITT; LANDAU; LEESE, 2001; HUANG, 2003) apresenta diversas medidas de similaridade e de distância (dissimilaridade), definidas a partir da natureza dos dados e do objetivo do problema. Quando se trabalha com similaridade, quanto maior o índice, mais próximos estão os elementos, já com a distância ocorre o inverso.

Para dados contínuos, as medidas mais usadas são a distância Euclidiana, a distância *city-block* e a distância Minkowski (KAUFMAN; ROUSSEEUW, 1990; COLE, 1998). A distância Euclidiana é dada por:

$$d(i, j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \dots + (x_{in} - x_{jn})^2} \quad (2.1)$$

ou seja, a distância entre dois objetos é o comprimento da hipotenusa de um triângulo retângulo.

A distância *city-block*, ou distância Manhattan, é bastante usada em projeto de arranjo físico, pois considera a distância possível de ser percorrida entre dois objetos, por exemplo, através de corredores perpendiculares. Esta distância é definida por:

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{in} - x_{jn}| \quad (2.2)$$

A distância Minkowski é uma generalização das distâncias Euclidiana e *city-block*, e é dada por:

$$d(i, j) = \sqrt[q]{|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \dots + |x_{in} - x_{jn}|^q} \quad (2.3)$$

onde $q \geq 1$.

Entretanto, em alguns problemas de agrupamento, a distância não pode ser usada como medida de dissimilaridade, dado que os valores dos elementos não são escalares. Por exemplo, em um problema que envolve atributos como endereço e nível de escolaridade são necessárias outras medidas que mostrem o grau de proximidade entre os elementos do conjunto de dados.

Para problemas relacionados à célula de fabricação, que envolvem a formação de famílias de peças e agrupamento de máquinas a partir de matriz binária peça-máquina, o coeficiente de Jaccard é um dos mais adotados (GONÇALVES FILHO, 2004a). O coeficiente de Jaccard para similaridade de peças é definido como:

$$S_{ij}^P = \frac{N_{ij}^M}{N_{ii}^M + N_{jj}^M - N_{ij}^M}, \text{ onde} \quad (2.4)$$

S_{ij}^P = Similaridade entre as peças i e j;

N_{ij}^M = Número de máquinas requeridas por ambas as peças;

N_{ii}^M = Número de máquinas requeridas pela peça i somente;

N_{jj}^M = Número de máquinas requeridas pela peça j somente.

O coeficiente de Jaccard para similaridade de máquinas é definido como:

$$S_{ij}^M = \frac{N_{ij}^P}{N_{ii}^P + N_{jj}^P - N_{ij}^P}, \text{ onde} \quad (2.5)$$

S_{ij}^M = Similaridade entre as máquinas i e j;

N_{ij}^P = Número de peças processadas por ambas as máquinas;

N_{ii}^P = Número de peças processadas pela máquina i somente;

N_{jj}^P = Número de peças processadas pela máquina j somente.

GONÇALVES FILHO (1988) propôs um coeficiente de similaridade usado para projeto de células de fabricação que indica como dois conjuntos de máquinas, representando duas peças, se interceptam. Considere uma peça representada por um conjunto de máquinas requeridas para seu processamento. O coeficiente de similaridade S entre duas peças X e Y é definido por:

$$S(X, Y) = \frac{n(X - Y)}{n(X \cap Y)} \quad \text{onde,} \quad (2.6)$$

$n(X - Y)$ representa o número de máquinas requeridas pela peça X e não requeridas pela peça Y e $n(X \cap Y)$, o número de máquinas comuns requeridas pelas duas peças.

Outra classe de medida de similaridade usada em projeto de arranjo físico é a que trabalha com comparação de seqüência de operações das peças. Essas medidas identificam diferenças e relações entre duas rotas. Em geral trabalham com operações de substituição, inserção e remoção de operações. Entre as medidas baseadas em seqüência de operações estão a distância de Levenshtein (TAM, 1990), coeficiente LCS (mais longa subsequência comum)

(ASKIN; ZHOU, 1998) e coeficiente *Merger* (HUANG, 2003). Essas medidas de similaridade estão detalhadas na seção 2.3.5.1.

2.2.5 Comparação entre os tipos tradicionais de arranjo físico

Tompkins et al. (2003) recomendam o uso de um dos tipos tradicionais de arranjo físico baseado na classificação segundo o volume de produção e a variedade de produtos (figura 2.5).

Adaptado de Tompkins et al. (2003)

Figura 2.5 – Classificação dos arranjos físicos tradicionais segundo volume-variedade

Como ilustra a figura 2.5, de acordo com Tompkins et al. (2003), o arranjo físico por produto é justificado quando a demanda é grande, estável e o produto padronizado. No arranjo celular, há um nível médio tanto de demanda como de variedade de peças. O arranjo físico funcional é usado quando a variedade de peças é grande, mas o volume de produção não é suficiente para justificar um arranjo físico por produto ou celular. No arranjo posicional a demanda é muito baixa e às vezes unitária.

Algumas vantagens e limitações dos tipos tradicionais de arranjo físico são apresentadas no quadro 2.1.

	Vantagens	Limitações
Posicional	<p>Reduzida movimentação de material.</p> <p>Trabalhador conhece o trabalho como um todo, o que resulta em maior qualidade e responsabilidade pelo produto.</p> <p>Alta flexibilidade, pode acomodar mudanças no projeto do produto e no volume de produção.</p>	<p>Maior movimentação de mão-de-obra e equipamentos.</p> <p>Pode resultar em duplicação de equipamentos.</p> <p>Requer maior habilidade da mão-de-obra.</p> <p>Pode resultar em aumento de espaço e maior estoque em processo.</p> <p>Programação da produção mais complexa.</p>
Produto	<p>Fluxo suave, simples e direto.</p> <p>Baixo estoque em processo.</p> <p>Menor tempo de produção unitário.</p> <p>Controle de produção simples.</p> <p>Menor movimentação de material.</p> <p>Requer menor habilidade do pessoal.</p> <p>Podem ser usados equipamentos especializados.</p>	<p>Parada de máquina pára a linha.</p> <p>Mudança no projeto do produto causa obsolescência no arranjo físico.</p> <p>Trabalho pode ser repetitivo.</p> <p>Alto investimento em equipamentos.</p> <p>Requer supervisão geral.</p>
Celular	<p>Maior utilização da máquina que o arranjo por processo devido à formação de famílias de peças.</p> <p>Fluxo mais suave e menor distância percorrida que o arranjo físico por processo.</p> <p>Permite uso de equipamentos de propósito geral.</p> <p>Trabalhador pode operar mais de uma máquina.</p> <p>Trabalho em equipe com aumento da comunicação e enriquecimento do trabalho.</p>	<p>Requer supervisão geral.</p> <p>Requer maior capacitação da mão-de-obra.</p> <p>Exige balanceamento do fluxo entre as células e dentro delas.</p> <p>Menor oportunidade de uso de equipamentos especializados.</p> <p>Pode necessitar de duplicação de equipamentos.</p>
Processo	<p>Permite uso de equipamentos de propósito geral.</p> <p>Alta flexibilidade na alocação do <i>mix</i> de produtos, pessoal e equipamentos.</p> <p>Pouca duplicação de máquinas.</p> <p>Supervisão relativamente fácil.</p>	<p>Maior movimentação de material.</p> <p>Controle da produção complexo.</p> <p>Aumento de estoque em processo.</p> <p>Requer maior capacitação do pessoal devido à diversidade de tarefas.</p> <p>Baixa utilização de recursos.</p>

Adaptado de Tompkins et al. (2003)

Quadro 2.1 – Vantagens e limitações dos tipos tradicionais de arranjo físico

2.3 Tipos não tradicionais de arranjos físicos

Segundo Benjaafar, Heragu e Irani (2002), as configurações de arranjos físicos tradicionais não atendem às necessidades de empresas que trabalham com vários tipos de produtos com demanda variável e curto ciclo de vida. Essas empresas precisam de arranjos flexíveis, modulares e de fácil configuração. Dentro desta necessidade surgiram novos tipos de arranjos físicos como o ágil, distribuído e modular. Outros tipos não tradicionais também aparecem na literatura como o arranjo físico celular híbrido (IRANI; HUANG, 1998) e o fractal (VENKATADRI; RARDIN; MONTREUIL, 1997).

2.3.1 Arranjo físico ágil

O arranjo físico ágil não se trata de uma nova forma de configuração do chão-de-fábrica. Benjaafar, Heragu e Irani (2002) e Benjaafar (2002) aplicam este termo para designar instalações que permitem reconfigurações frequentes do chão-de-fábrica e são projetadas para maximizar o desempenho operacional ao invés de minimizar o custo de movimentação de materiais. Segundo aqueles autores, um critério de projeto de arranjo físico baseado em distância média percorrida é um indicador pobre do desempenho operacional, pois pode levar à acumulação de estoque em processo. Os autores abordam a necessidade de considerar no projeto do arranjo físico tanto a distância cheia como a distância vazia percorrida pelos transportadores e também a variância da distância. Nos arranjos físicos ágeis outras medidas de desempenho passam a ser importantes como o tempo de ciclo, estoque em processo (WIP) e tempo de passagem das peças pelo sistema. Os autores ressaltam também a importância da integração do arranjo físico com o sistema de movimentação de materiais. Como exemplo de empresas que têm experimentado arranjos físicos ágeis no sentido de maximizar o

desempenho operacional, Benjaafar, Heragu e Irani (2002) citam a Volvo, a Lucent, a Volkswagen e a General Motors. A fábrica da Volvo na cidade de Kalmar, Suécia, é composta por módulos hexagonais onde o fluxo de materiais é formado por linhas concêntricas dentro de cada módulo. A Lucent têm experimentado arranjos físicos com processadores compartilhados, centralmente localizados em departamentos funcionais e eqüidistantes de várias células dedicadas na fábrica. No Brasil, a fábrica de caminhões da Volkswagen em Resende (RJ) adota o arranjo físico em espinha. O produto se move ao longo de uma artéria principal e ligado a esta estão mini-fábricas de fornecedores independentes que acoplam seus módulos ao produto principal. Na fábrica da General Motors em Gravataí (RS), 16 fornecedores são conectados à linha de montagem principal, através de um sistema compartilhado de movimentação de materiais com empilhadeiras, caminhões e esteiras transportadoras.

2.3.2 Arranjo físico distribuído

A idéia do arranjo físico distribuído é, segundo Benjaafar, Heragu e Irani (2002), desagregar os grandes departamentos funcionais em sub-departamentos e distribuí-los pela fábrica. O mesmo ocorre com os recursos, como máquinas e equipamentos. A duplicação estratégica de departamentos pela fábrica permite lidar com flutuações no padrão de fluxo e no volume de produção. Este arranjo físico é especialmente indicado quando a flutuação da demanda é grande e é inviável do ponto de vista de custos reconfigurar a fábrica conforme a demanda. Embora a duplicação dos departamentos aumente a flexibilidade, outros recursos como pessoal, sistemas computacionais, áreas de carga/descarga, também devem ser duplicados. Portanto, há um *trade-off* entre os benefícios da desagregação/duplicação e suas implicações, que deve ser analisado. Alguns benefícios do arranjo físico distribuído são:

- Menor tempo de passagem das peças pelo sistema;
- Redução significativa no custo de movimentação de materiais;
- Permite o uso de células virtuais temporárias formadas para um produto particular ou ordem de produção. Após completar a ordem ou produto, a célula é desfeita.

A figura 2.6 apresenta dois exemplos de arranjo físico distribuído. Em 2.6a o arranjo físico é parcialmente distribuído. Agrupamentos do mesmo tipo de máquina são distribuídos pela fábrica. No arranjo físico da figura 2.6b, as máquinas são espalhadas individualmente pelo chão-de-fábrica, o que caracteriza o arranjo físico maximamente distribuído.

Benjaafar, Heragu e Irani (2002)

(a) Arranjo físico parcialmente distribuído

(b) Arranjo físico maximamente distribuído

Figura 2.6 – Exemplos de arranjo físico distribuído

Gorgulho Júnior (2006) apresenta uma revisão detalhada sobre arranjo físico distribuído e em toda a literatura analisada pelo autor apenas um trabalho usou dados reais, mas sem que o arranjo físico fosse implementado. Os demais trabalhos são baseados em dados gerados por seus autores. Portanto, Gorgulho Júnior afirma que o arranjo físico distribuído ainda encontra-se em fase de pesquisa.

2.3.3 Arranjo físico celular híbrido

O arranjo físico celular híbrido (HCL – *Hybrid Cellular Layout*) possui propriedades intermediárias entre o celular e o funcional. Este arranjo tenta evitar a separação física de máquinas idênticas em várias células sem destruir a composição da célula original. O HCL é desenvolvido a partir da matriz peça-máquina usada para projetar células independentes. Entretanto, são usadas estratégias para colocar máquinas compartilhadas como se permanecessem em seções funcionais. Irani e Huang (1998) apresentam várias configurações de arranjo físico celular híbrido, apresentadas a seguir.

Celular com Reorientação das Células

A célula é rotacionada 90° (figura 2.8), assim algumas máquinas ficam adjacentes a máquinas do mesmo tipo de outra célula, como se fosse um grupo funcional. No caso de quebra de máquina ou mudança na demanda, a peça poderia ser transferida entre as máquinas do mesmo tipo.

Para melhor compreensão e visualização dos tipos de arranjo celular híbrido, é apresentado primeiramente na figura 2.7 um arranjo físico celular tradicional com três células.

Irani e Huang (1998)

Figura 2.7 – Arranjo físico celular com três células

Em relação à figura 2.7, na figura 2.8 a célula 2 foi rotacionada 90°, permitindo a formação de grupos funcionais entre as células.

Irani e Huang (1998)

Figura 2.8 – Arranjo físico celular com reorientação das células

Celular com Reorientação e Mudança na Forma das Células

Ao invés de permanecerem, por exemplo, em forma de U ou retangular, a forma das células passa para L ou S. Isso permite que mais máquinas de células diferentes formem grupos funcionais. Em relação à figura 2.7, a figura 2.9 mostra a reorientação e mudança da forma das células 2 e 3 e a mudança da forma da célula 1 com inversão (espelhamento) das máquinas.

Irani e Huang (1998)

Figura 2.9 – Arranjo físico celular com reorientação e mudança na forma das células

Celular com Linhas em Forma de S

Para cada célula é feito um arranjo físico em linha (*flowline*). A seguir, as células são colocadas em paralelo para reduzir o fluxo intercelular. Então, a forma linear da célula é mudada para S (figura 2.10), para agrupar máquinas idênticas em seções funcionais.

Irani e Huang (1998)

Figura 2.10 – Arranjo físico celular com linhas em forma de S, em paralelo

Flowshop Híbrido

As máquinas são arranjadas em grupos, e estes em seqüência, como mostra a figura 2.11. Cada grupo não processa uma família de peças e sim uma ou mais operações consecutivas para a maioria das peças.

Irani e Huang (1998)

Figura 2.11– Arranjo físico *flowshop* híbrido

Celular Virtual

Células são colocadas adjacentes de forma que máquinas idênticas formem seções funcionais. A adjacência das células permite que grupos de máquinas baseados em famílias de peças sejam virtuais. Isso é possível devido a um sistema de movimentação que liga máquinas em células adjacentes (figura 2.12) sem necessitar de uma localização rígida da máquina na célula.

Irani e Huang (1998)

Figura 2.12 – Arranjo físico celular virtual

Celular com uma Célula Remanescente

Elimina o problema de duplicação de máquinas do arranjo físico celular. As máquinas compartilhadas permanecem em uma célula remanescente entre as células originais. Dentro da célula remanescente as máquinas são arranjadas conforme arranjo físico funcional (figura 2.13).

Irani e Huang (1998)

Figura 2.13 – Arranjo físico celular com uma célula remanescente

Na figura 2.13, a composição original das células 1 e 2 foi desfeita, para facilitar o fluxo de peças através da célula remanescente.

Células em Cascata

Este arranjo físico depende da similaridade das rotas. Peças simples são alocadas em pequenas células e peças complexas em células maiores (figura 2.14). Não há movimento intercelular. É preciso uma significativa duplicação de máquinas neste arranjo físico.

Irani e Huang (1998)

Figura 2.14 – Células em cascata

Segundo Irani e Huang (2006), um exemplo onde o arranjo físico celular híbrido do tipo células em cascata é aplicado é o setor de emergência de hospitais. Para cada nível de

cuidado necessário ao paciente, é visitada uma seqüência de departamentos onde são feitos os atendimentos. Fazendo uma analogia a um sistema de manufatura, o paciente é análogo à peça, e a seqüência de departamentos à seqüência de operações.

2.3.4 Arranjo físico fractal

No tipo de arranjo físico denominado fractal (VENKATADRI; RARDIN; MONTREUIL, 1997) são criadas mini-fábricas dentro da fábrica. A idéia central é fazer uma dispersão das máquinas na fábrica através de zonas, que possuem rotas de produtos dominantes.

A unidade básica de organização é a célula fractal, um conjunto de máquinas (ou estações) capaz de processar a maioria dos produtos do sistema. A quantidade de máquinas em cada célula é proporcional à quantidade da mesma máquina na fábrica. Cada célula tem aproximadamente a mesma capacidade de processamento. Cada célula funciona como uma mini-fábrica multifuncional. A figura 2.15 ilustra um arranjo físico fractal constituído por três células fractais.

Venkatadri, Rardin e Montreuil (1997)

Figura 2.15 – Exemplo de arranjo físico fractal

Uma das vantagens do arranjo físico fractal está no fato de uma peça poder ser processada em qualquer uma das células. Isso faz com que o arranjo físico fractal tenha

flexibilidade, o que não ocorre na manufatura celular. Além disso, o arranjo fractal não exige que as células sejam independentes. Uma peça pode passar por duas ou mais células fractais para completar seu processamento.

Segundo Venkatadri, Rardin e Montreuil (1997), os principais desafios do projeto fractal são: maior dificuldade de gerenciamento e designação do fluxo, uma vez que as peças processadas em cada célula não são tão similares; a determinação do arranjo físico intra-celular devido à variedade de processos dentro da célula e a determinação do arranjo físico inter-celular devido ao fato das células não serem independentes.

Segundo Gonçalves Filho (2004b), o arranjo físico fractal parece ser adequado para manufatura com alta variedade de produtos e lotes pequenos de fabricação. Outra situação em que o uso de células fractais com basicamente a mesma composição parece ser viável ocorre quando a fabricação do produto requer mão de obra intensiva e equipamentos simples e baratos. Um exemplo desta situação é a fabricação de cestas de madeira.

2.3.5 Arranjo físico modular

O arranjo físico modular proposto por Irani e Huang (1998), não assume *a priori* padrões de fluxo ou critérios para agrupamento de máquinas, como nos arranjos físicos celular e funcional tradicionais. Os padrões de fluxo e agrupamentos de máquinas são determinados pelos fluxos implícitos nas seqüências de operações. Dessa forma, o arranjo físico modular é mais flexível que os tradicionais (em linha, funcional e celular), pois identifica os vários padrões de fluxo existentes na fábrica e gera um arranjo físico para cada fluxo específico de materiais.

O conceito do arranjo físico modular partiu da observação de que roteiros de produção de diferentes produtos possuem subsequências de operações comuns, que poderiam ser

agregadas em módulos. Um módulo é um grupo de máquinas conectadas por um fluxo de materiais bem definido. O objetivo do arranjo físico modular é maximizar o número de operações consecutivas de um conjunto de produtos feitas no mesmo módulo. Os autores classificaram seis tipos de módulos, mostrados a seguir.

Módulo *Flowline* (Linha): Arranjo linear tal que todo movimento entre duas operações consecutivas é sempre para frente, em sequência ou saltando máquinas (*bypass*) se necessário. Caso haja contra-fluxo, uma das decisões deve ser tomada: modificar da forma linear para segmentos lineares com *loop* circular entre os lineares; permanecer com a forma linear mas usar um sistema de movimentação de materiais bidirecional; ou duplicar a máquina para que não haja contra-fluxo (figura 2.16a).

Módulo *Flowline* Ramificado (convergente/divergente): Em um ponto a linha é dividida em ramos em paralelo, cada um específico para um tipo de produto. Caso as operações subsequentes forem comuns, os ramos voltam a convergir para uma única linha (figura 2.16b).

Módulo Célula: Conjunto de máquinas diferentes que juntas podem processar um produto completo, ou uma família de produtos. Não é necessário que sejam feitas operações fora do módulo célula (figura 2.16c). Semelhante ao arranjo físico celular tradicional.

Módulo Centro de Usinagem: Constituído por apenas uma máquina com alto grau de automação, capaz de produzir uma família de peças. Este módulo pode substituir várias máquinas, eliminando a movimentação entre as mesmas e o tempo de *setup*, características das células tradicionais (figura 2.16d).

Módulo Funcional: Diferentes máquinas são usadas por diferentes peças. O padrão de fluxo é aleatório. Semelhante ao arranjo físico funcional tradicional (figura 2.16e).

Módulo Fluxo Padronizado: Há um fluxo dominante e precedência entre as operações, definindo uma hierarquia entre elas. O módulo pode ser decomposto em módulos *flowline* ou *flowline* ramificado (figura 2.16f).

Irani e Huang (1998)

Figura 2.16 – Tipos de módulos

Como exemplo de arranjo físico modular, Irani e Huang (2000b) citam uma fábrica da Motorola. A empresa desejava mudar o arranjo de uma fábrica de semicondutores de funcional para celular. Entretanto, os autores avaliaram que o arranjo físico celular não seria

viável devido à necessidade de duplicação de equipamentos e processos. A análise e comparação dos roteiros de produção revelaram que havia muitas subsequências de operações em comum. Dessa forma, os autores propuseram o arranjo físico modular, que combina características dos três tipos de arranjos tradicionais, por produto, funcional e celular.

Outra implantação de arranjo físico modular foi realizada por Irani e Huang (2004) na fábrica da Ulven Forging Inc. localizada na cidade de Hubbard, OR, USA. Segundo os autores, com a implantação do arranjo físico recomendado houve uma economia anual de U\$137,000 (cento e trinta e sete mil dólares).

O procedimento para projeto de arranjo físico modular proposto por Huang (2003), fundamenta-se na comparação de seqüências de produção e análise de agrupamentos (*cluster analysis*). Para comparação de seqüências de produção, Huang (2003) desenvolveu um novo coeficiente de similaridade, o coeficiente *Merger*. A seguir são apresentados três coeficientes para comparação de seqüências, abordados por Huang (2003) e usados no presente trabalho. O procedimento para projeto de arranjo físico modular é apresentado na seção 2.3.5.2.

2.3.5.1 Similaridade entre seqüências de operações

Coeficiente baseado na Distância de Levenshtein (TAM, 1990)

A distância de Levenshtein (TAM, 1990) entre duas seqüências é definida como o número mínimo de transformações requeridas para obter uma seqüência a partir da outra. As transformações são: substituição, cancelamento e inserção.

Exemplo: Sejam $x = ADHI$ e $y = ABFDHG$.

A seqüência de transformações para passar de x para y é:

portanto, $d(x,y) = 3$.

Neste trabalho será adotada a distância de Levenshtein normalizada (equação 2.7), já que os demais coeficientes usados, *Merger* e LCS variam no intervalo $[0,1]$.

$$S_i = 1 - \frac{d_i}{\max d_i} \quad (2.7)$$

onde d_i é a distância de Levenshtein.

Coeficiente LCS (ASKIN; ZHOU, 1998)

Askin e Zhou (1998) propuseram um coeficiente de similaridade baseado na mais longa subsequência comum (LCS – *longest common subsequence*) entre peças. O coeficiente de similaridade S_{ij} entre as seqüências O_i e O_j é dado pela equação:

$$S_{ij} = \max \left\{ \frac{|LCS_{ij}|}{|O_i|}, \frac{|LCS_{ij}|}{|O_j|} \right\} \quad \text{onde,} \quad (2.8)$$

LCS_{ij} é a mais longa subsequência comum entre O_i e O_j ; $|O_i|$ e $|O_j|$ representam o número de operações nas seqüências i e j respectivamente.

Coeficiente *Merger* (HUANG, 2003)

Segundo Huang (2003), a distância de Levenshtein considera somente a diferença entre duas seqüências, ignorando suas subsequências comuns. Isso não é eficiente em situações onde o número de operações em duas seqüências difere significativamente. Nestas situações, seqüências com aparente diferença podem possuir alta similaridade. Já o coeficiente LCS está focado na ordem das operações comuns em duas seqüências, mas ignora a distância entre conjuntos de operações comuns. Isto pode gerar agrupamentos onde, para executar duas

operações consecutivas a peças necessite saltar várias máquinas (*bypass*) aumentando significativamente a distância percorrida.

Para evitar tais efeitos indesejáveis, Huang (2003) propôs o coeficiente *Merger*, que avalia a viabilidade de absorção de uma seqüência completamente em outra. Este coeficiente também usa as operações de substituição, inserção e remoção. Além disso, considera as subsequências de operações comuns e a distância existente entre elas.

A equação 2.9 mostra o cálculo do coeficiente *Merger*, $mc(x,y)$:

$$mc(x,y) = \begin{cases} \max \left\{ 1 - \frac{md(y,x) + \frac{id(y,x)}{N_{\max}} + \frac{N_x - N_y}{N_{\max}^2}}{N_y}, 0 \right\} & se \quad N_x > N_y \\ \max \left\{ 1 - \frac{md(x,y) + \frac{id(x,y)}{N_{\max}} + \frac{N_y - N_x}{N_{\max}^2}}{N_x}, 0 \right\} & se \quad N_x < N_y \\ \max \left\{ 1 - \frac{md(x,y) + \frac{id(x,y)}{N_{\max}}}{N_x}, 1 - \frac{md(y,x) + \frac{id(y,x)}{N_{\max}}}{N_y}, 0 \right\} & se \quad N_x = N_y \end{cases} \quad (2.9)$$

onde $N_{\max}$ é o número de operações da seqüência mais longa; e N_x N_y representam o número de operações da seqüência x e y respectivamente; $md(x,y)$ é a distância *Merger*; e $id(x,y)$ é a distância de interrupção, detalhadas a seguir. Propriedades do coeficiente *Merger*:

- 1) $0 \leq mc(x,y) \leq 1$;
- 2) $mc(x,y) = mc(y,x)$;
- 3) $mc(x,y) = 1$ se e somente se as seqüências x e y forem idênticas;
- 4) $mc(x,y) = 0$ se e somente se as seqüências x e y não possuírem operações comuns.

Distância *Merger* $md(x,y)$

Distância para absorção da sequência x na sequência y . É o menor número de substituições e inserções de operações na sequência y , tal qual a sequência x fique contida na sequência y , preservando a sequência de elementos de x .

Distância de Interrupção $id(x,y)$

Mantendo a Distância *Merger*, $md(x,y)$, fixa, é o menor número de remoções não finais requeridas, ou seja, nem no início e nem no final da sequência y . Uma remoção não final aparece entre duas inserções e/ou substituições consecutivas, ou entre duas operações consecutivas comuns (*matching*), ou entre uma inserção (ou substituição) e uma operação comum (*matching*).

Tomemos o exemplo apresentado por Narayanan (2002), para cálculo do coeficiente *Merger*. Sejam X e Y duas seqüências de operações: $X = (1,2,3)$ e $Y = (1,4,5,3)$.

Passo1: Cálculo da Distância *Merger*, $md(x,y)$.

A partir de Y , as seguintes possibilidades podem ocorrer para que y contenha x :

1. $(1, 4, 5, 3)$ -----→ $(1, \mathbf{2}, 5, 3)$ - substituição do 4 pelo 2
2. $(1, 4, 5, 3)$ -----→ $(1, 4, \mathbf{2}, 3)$ – substituição do 5 pelo 2
3. $(1, 4, 5, 3)$ -----→ $(1, \mathbf{2}, 4, 5, 3)$ – inserção do 2 entre 1 e 4
4. $(1, 4, 5, 3)$ -----→ $(1, 4, \mathbf{2}, 5, 3)$ – inserção do 2 entre 4 e 5
5. $(1, 4, 5, 3)$ -----→ $(1, 4, 5, \mathbf{2}, 3)$ – inserção do 2 entre 5 e 3

Portanto $md(x,y) = 1$, ou seja, é necessário apenas uma substituição ou inserção para que y contenha x .

Passo 2: Cálculo da Distância de Interrupção, $id(x,y)$.

A partir da sequência intermediária obtida no passo 1, as seguintes possibilidades de remoção podem ocorrer para converter a sequência intermediária em x:

1. Sequência intermediária (1, 2, 5, 3)

(1, 2, 5, 3)-----→ (1, 2, 3) = sequência x - remoção do 5 - $id(x,y) = 1$

2. Sequência intermediária (1, 4, 2, 3)

(1, 4, 2, 3) -----→ (1, 2, 3) = sequência x - remoção do 4 - $id(x,y) = 1$

3. Sequência intermediária (1, 2, 4, 5, 3)

(1, 2, 4, 5, 3)-----→ (1, 2, 3) = sequência x - remoção do 4 e do 5 - $id(x,y) = 2$

4. Sequência intermediária (1, 4, 2, 5, 3)

(1, 4, 2, 5, 3)-----→ (1, 2, 3) = sequência x - remoção do 4 e do 5 - $id(x,y) = 2$

5. Sequência intermediária (1, 4, 5, 2, 3)

(1, 4, 5, 2, 3)-----→ (1, 2, 3) = sequência x - remoção do 4 e do 5 - $id(x,y) = 2$

Portanto, o menor valor da distância de interrupção é $id(x,y) = 1$.

Passo 3: Cálculo do coeficiente *Merger*.

Com $md(x,y) = 1$, $id(x,y) = 1$, $N_x = 3$, $N_y = 4$ e $N_{\max} = 4$ e substituindo na equação 2.9, temos:

$$\begin{aligned}
 mc(x,y) &= \max \left\{ 1 - \frac{md(x,y) + \frac{id(x,y)}{N_{\max}} + \frac{N_y - N_x}{N_{\max}^2}}{N_x}, 0 \right\} \\
 &= \max \left\{ 1 - \frac{1 + \frac{1}{4} + \frac{4-3}{4^2}}{3}, 0 \right\} = \max \left\{ 1 - \frac{1 + \frac{1}{4} + \frac{1}{16}}{3}, 0 \right\} = \max \{0,5625, 0\}
 \end{aligned}$$

Portanto, o coeficiente *Merger*, é $mc(x,y) = 0,5625$. Quanto mais próximo do valor 1 (um), mais similares são as seqüências de operações.

2.3.5.2 Procedimento para projeto de arranjo físico modular

Os dados de entrada necessários para o projeto de arranjo físico modular são o conjunto de peças com a seqüência de operações e a quantidade produzida de cada peça, e os tempos de processamento de cada operação. O procedimento é composto das etapas (HUANG, 2003):

Etapla 1 – Identificação das subseqüências comuns entre todos os pares de seqüências de operações. Uma subseqüência comum é definida como uma seqüência de operações consecutivas comuns a duas ou mais seqüências de operações. O algoritmo da figura 2.17 mostra como encontrar as subseqüências comuns.

Huang (2003)

Figura 2.17 – Algoritmo para determinar as subseqüências comuns

Na figura 2.17 as variáveis significam:

X_i = A i-ésima operação da sequência X;

Y_j = A j-ésima operação da sequência Y;

len = número de operações consecutivas comuns às sequências X e Y;

m = tamanho da sequência X;

n = tamanho da sequência Y;

Subseq_Com = subsequência de operações consecutivas comum às sequências X e Y.

A seguir é apresentado um exemplo que ilustra a aplicação do algoritmo da figura

2.17. Sejam as peças X e Y com as seguintes sequências de operações:

Peça X = {1, 4, 8, 9} : m = 4 (tamanho da sequência X)

Peça Y = {1, 4, 7, 4, 8, 7} : n = 6 (tamanho da sequência Y)

A tabela 2.1 mostra um exemplo para identificação de subsequência comum.

Tabela 2.1 – Exemplo para identificação de subsequência comum

i	X_i	j	Y_j	len	Subseq_Com
1	1	1	1	0	-
2	4	2	4	1	-
3	8	3	7	2	{1,4}
3	8	4	4	0	-
3	8	5	8	0	-
4	9	6	7	1	-
4	9	7	1	0	-
5	1	1	1	0	FIM

Portanto, o algoritmo identificou {1, 4} como subsequência comum através da comparação da sequência X com a sequência Y. Para que todas as subsequências comuns entre o par X-Y sejam identificadas, também deve ser feita a comparação entre Y e X.

Após a aplicação do algoritmo, identifique as subsequências comuns únicas, ou seja, que não se repetem, e estabeleça a sua frequência de ocorrência nas rotas originais.

Etapa 2 – Defina as subsequências comuns dominantes. Subsequências comuns dominantes (DOM) são aquelas cuja frequência de ocorrência nas rotas originais é maior que um limitante (*threshold*) estabelecido pelo usuário. Em seu trabalho Huang (2003) usou limitante DOM = 1, ou seja, considerou todas as subsequências comuns únicas como dominantes.

Etapa 3 – Gere a matriz de similaridade. A matriz contém o valor da similaridade entre cada par de subsequência comum dominante. A similaridade é calculada de acordo com o coeficiente *Merger*, detalhado na seção 2.3.5.1

Etapa 4 – Agrupe as subsequências comuns dominantes usando análise de agrupamento (*cluster analysis*). Nesta etapa Huang (2003) propôs um procedimento heurístico que usa método de agrupamento hierárquico aglomerativo com método da média das distâncias para regra de ligação (*unweighted pair-group average linkage*). Ao final desta etapa são gerados os módulos básicos de arranjo físico. Os módulos básicos definidos por Huang (2003) são *flowline*, *flowline* ramificado e fluxo padronizado, ou seja, aqueles que possuem estrutura com fluxo dominante.

Etapa 5 – Gere módulos funcionais se necessário. Se dois módulos possuem muitas máquinas em comum, eles são agrupados em um módulo funcional para reduzir a duplicação de máquinas. Inicialmente calcule o índice de semelhança (*commonality*) entre todos os módulos gerados no passo 4. Sejam dois módulos M_i e M_j , o índice de semelhança é dado por:

$$\text{Índice de Semelhança} = \frac{n_{ij}}{\min(n_i, n_j)} \quad (2.10)$$

onde n_{ij} é o número de operações comuns a ambos os módulos; n_i e n_j é o número de operações dos módulos M_i e M_j respectivamente. Caso haja operação repetida no módulo, ela é contada apenas uma vez.

Estabeleça um limitante V (*threshold*) no intervalo $0 \leq V \leq 1$. Se o índice de semelhança for maior que o limitante V , agregue os dois módulos.

Etapa 6 – Expresse as seqüências de operações originais em termos dos módulos de arranjo físico. As seqüências de operações originais são aplicadas aos módulos resultantes da etapa anterior. Caso para determinada peça, haja máquinas residuais (que não se encontram nos módulos), analise a possibilidade de absorvê-las em módulos adjacentes conforme procedimento heurístico (HUANG, 2003).

Etapa 7 – Analise a viabilidade de duplicação de máquinas. Determine o número de máquinas de cada tipo para cada módulo de arranjo físico conforme análise de capacidade:

$$N_{jk} = \sum_i \frac{T_{ijk}}{A_j} \quad \text{onde,} \quad (2.11)$$

N_{jk} = número de máquinas do tipo j requerido no módulo k ;

T_{ijk} = capacidade requerida pela peça i na máquina j no módulo k ;

A_j = capacidade disponível por máquina do tipo j .

Ainda,

$$A_j = \Delta U_j \cdot C \quad \text{onde,} \quad (2.12)$$

ΔU_j = porcentagem de utilização da máquina j ;

C = duração do período de produção (horas).

Caso a carga de determinado tipo de máquina seja muito baixa em um ou mais módulos, Huang (2003) recomenda a análise de compra de máquina flexível multifuncional.

Etapa 8 – Gere o arranjo físico como uma rede de módulos. Represente os módulos graficamente mostrando o fluxo intra e inter-módulos. Quando um módulo é estaque (não há fluxo inter-módulos), ele é chamado módulo célula. Gere o arranjo físico de bloco de acordo com heurística (*maximum weigh planar graph*) proposta por Foulds e Giffin apud Huang (2003).

De acordo com Huang (2003) e Irani e Huang (2006), este procedimento para projeto de arranjo físico modular apresenta as seguintes contribuições:

- Ao invés de usar a abordagem tradicional de agrupamento peça-máquina ou de sequência de operações, introduz uma nova abordagem de detecção *a priori* de subsequências comuns para realizar o agrupamento, com o objetivo de projetar módulos de arranjo físico.
- Admite a formação de “células parciais” (módulos) que agrupam produtos dissimilares em famílias conforme as rotas dos produtos específicas para uma unidade fabril em particular. Este conceito é extensivo aos departamentos, formados por uma combinação de múltiplos processos compatíveis, ao invés de um processo único.
- Apresenta uma abordagem lógica para duplicação de máquinas baseado na ocorrência das mesmas em diferentes combinações de operações requeridas por diversas famílias de rotas e não apenas baseada na carga da máquina;
- Mais de um tipo de *layout* pode ser usado para arranjar diferentes máquinas e recursos na fábrica;
- Melhora o fluxo de materiais que passa a ser um facilitador do controle do chão-de-fábrica, e não trata apenas da otimização de um tipo específico de *layout* que não se ajusta aos padrões de fluxo existentes na fábrica.

2.3.6 Comparação entre os tipos não-tradicionais de arranjo físico

Algumas vantagens e limitações dos tipos não tradicionais de arranjo físico são sumarizadas no quadro 2.2. Essas características foram extraídas das referências citadas ao longo do texto do item 2.3.

	Vantagens	Limitações
Ágil	Adaptável às mudanças no projeto do produto, <i>mix</i> de produtos e volume de demanda.	Alto investimento em equipamentos, ferramentas e sistemas de movimentação flexíveis.
Distribuído	Permite lidar com variações na demanda, tanto em volume como em variabilidade. Flexibilidade devido à duplicação e/ou desagregação de departamentos. Menor tempo de passagem das peças pelo sistema. Redução no custo de movimentação de materiais.	Necessidade de duplicação de recursos como pessoal, sistemas computacionais e áreas de carga e descarga.
Fractal	Flexibilidade: uma peça pode ser processada em qualquer uma das células fractais. Uma peça pode passar por mais de uma célula para completar seu processamento.	Exige balanceamento do fluxo entre as células e dentro delas. Necessidade de duplicação e máquinas e equipamentos. Gerenciamento complexo.
Celular Híbrido	No caso de quebra de máquina ou mudança na demanda, permite a transferência de peça entre máquinas do mesmo tipo, devido a criação de grupos funcionais.	Possui as mesmas limitações do arranjo físico celular tradicional.
Modular	Fluxo suave. Não assume critérios <i>a priori</i> (como por exemplo baseados em volume e variedade) para o agrupamento de máquinas. Mais de um tipo de arranjo físico pode ser usado na fábrica. Uma peça pode passar por mais de um módulo para completar seu processamento.	Pode haver necessidade de duplicação de máquinas e equipamentos.

Quadro 2.2 – Vantagens e limitações dos tipos não tradicionais de arranjo físico

2.4 Considerações finais

Este capítulo abordou os tipos tradicionais de arranjos físicos como o posicional, por produto, funcional e celular, assim como as vantagens e desvantagens de cada um deles. Entre as novas configurações de arranjos físicos destacam-se o distribuído, ágil, celular híbrido, fractal e modular. Huang (2003) apresenta uma classificação dos arranjos físicos tradicionais e não tradicionais de acordo com o foco no produto ou no processo como base para o agrupamento e disposição das máquinas no chão de fábrica. Esta classificação é mostrada na figura 2.18.

Adaptado de Huang (2003)

Figura 2.18 – Classificação dos arranjos físicos tradicionais e não tradicionais

A figura 2.18 mostra que o arranjo físico modular, alvo deste trabalho, encontra-se em uma posição intermediária entre os arranjos celular e funcional. O foco do arranjo celular é o produto através da formação de famílias de peças, já o arranjo físico funcional está voltado para o processo. O modular possui tanto foco no produto como no processo, pois permite diferentes configurações de arranjos físicos, além disso, é mais flexível que o celular, visto que permite que uma peça visite mais de um módulo para completar o seu processamento.

O capítulo seguinte apresenta os principais conceitos sobre algoritmos genéticos e aplicações destes ao projeto de arranjo físico.

3 ALGORITMOS GENÉTICOS

3.1 Considerações iniciais

Um algoritmo genético (AG) é um método de busca e otimização inspirado no princípio de seleção natural e sobrevivência dos indivíduos mais aptos descoberto pelo fisiologista Charles Darwin e descrito em seu livro *A Origem das Espécies* de 1859. Algoritmos genéticos foram introduzidos por John Holland em 1975 e popularizados por David Goldberg em 1989.

“Um algoritmo genético é um método genérico adaptativo de busca que imita o processo de evolução natural através da combinação da sobrevivência dos mais aptos entre estruturas de *strings* com uma troca estruturada, ainda que aleatória, de informações” (GOLDBERG, 1989).

Segundo Goldberg (1989), as principais características de um algoritmo genético (AG) são:

- Trabalha com a codificação do conjunto de parâmetros em uma *string*;
- Realiza busca a partir de uma população de pontos (possíveis soluções), e não de um ponto único;
- Usa informação da função objetivo (função de aptidão) para efetuar busca por melhores soluções;
- Usa regras de transição probabilísticas (escolha aleatória) para guiar a busca em regiões do espaço com maior chance de melhoria.

Muitos termos usados em algoritmos genéticos são originados da biologia. O quadro 3.1 a seguir, sumariza os principais termos segundo Lacerda, Carvalho e Ludermir (1999).

Termo	Biologia	Algoritmos Genéticos
Cromossomo ou Indivíduo	Conjunto de genes de um organismo.	<i>String</i> que representa uma solução para o problema.
Gene	Unidade de hereditariedade transmitida pelo cromossomo e que controla as características do organismo.	Parâmetro codificado no cromossomo, ou seja, um elemento da <i>string</i> .
Genótipo	Composição genética contida no cromossomo.	Informação contida no cromossomo.
Fenótipo	Manifestação física de um gene no organismo, por exemplo, pele lisa ou rugosa.	Estrutura construída a partir do genótipo.
Alelo	Uma das formas alternativas de um gene.	Valores que o gene pode assumir.

Adaptado de Lacerda, Carvalho e Ludermir (1999)

Quadro 3.1 – Analogia de termos entre algoritmos genéticos e a biologia

Um AG inicia com uma população de indivíduos (cromossomos) que representam possíveis soluções do problema a ser resolvido. A população é avaliada e cada cromossomo recebe uma nota, denominada aptidão. Os melhores cromossomos (mais aptos) são selecionados e os piores descartados. Os selecionados podem sofrer cruzamento e mutação, gerando descendentes para a próxima geração. O processo continua até que uma solução satisfatória seja encontrada (LACERDA; CARVALHO; LUDERMIR, 1999). Este processo é ilustrado na figura 3.1.

Gonçalves Filho, Gorgulho Júnior e Argoud (2004)

Figura 3.1 – Princípio de trabalho de um algoritmo genético

Os principais elementos de um algoritmo genético são ilustrados na figura 3.2.

Figura 3.2 – Principais elementos de um algoritmo genético

Na figura 3.2, a codificação refere-se à forma pela qual o cromossomo representa uma solução. A população inicial é formada por um conjunto de cromossomos. A função de aptidão irá avaliar a qualidade da solução. A seleção é o processo de escolha dos cromossomos mais aptos que posteriormente irão sofrer a ação dos operadores genéticos (cruzamento e mutação). Após a ação dos operadores genéticos são gerados os cromossomos descendentes. A geração da nova população refere-se ao modo de inserção dos descendentes na população. O critério de parada trata da forma de término do algoritmo. Os parâmetros devem ter seus valores definidos pelo usuário para execução do AG. Cada um dos elementos da figura 3.2 é detalhado a seguir.

3.2 Codificação do cromossomo

Existem diversas formas de codificação ou representação do cromossomo. A codificação deve ser capaz de conter todas as possíveis soluções do problema. Algumas formas encontradas na literatura para representar o problema de projeto de arranjo físico são:

Grupo-Número

A codificação grupo-número é uma das mais usadas em problemas de projeto de arranjo físico. Os genes do cromossomo representam os objetos a serem agrupados, e o valor do gene indica o subconjunto ao qual o objeto pertence. Portanto, o tamanho do cromossomo equivale ao número de objetos a serem agrupados. Exemplos:

a) Célula-Máquina e Peça-Rota (GRAVEL; NSAKANDA; PRICE, 1998):

Seja o cromossomo célula-máquina: (3 3 3 1 3 2 2 1).

Este cromossomo possui 8 genes, que representam 8 máquinas a serem agrupadas. O primeiro gene representa a máquina M1, o segundo gene, a máquina M2, e assim por diante. O valor do gene indica a célula a qual a máquina está designada. Dessa forma, neste exemplo, a célula 1 é composta pelas máquinas M4 e M8, a célula 2 pelas máquinas M6 e M7, e a célula 3 por M1, M2, M3 e M5. A forma de gerar o cromossomo é explicada adiante, na seção 3.3.

Seja o cromossomo peça-rota: (3 2 2 1 3)

A peça 1 usa a rota 3, a peça 2 usa a rota 2, a peça 3, a rota 2, e assim por diante.

b) Grupo-número com separador (SIGH; RAJAMANI, 1996):

Nesta codificação a primeira parte do cromossomo refere-se às máquinas, e a segunda parte às peças. Seja o cromossomo:

(2 1 2 1 3 | 2 2 3 3 1)
máquinas peças

Célula 1: M2 M4 P5; Célula 2: M1 M3 P1 P2; Célula 3: M5 P3 P4

Matriz Binária (NALDI; CARVALHO, 2005)

As colunas representam os objetos a serem agrupados e as linhas os subconjuntos aos quais pertencem. Exemplo:

$$\begin{matrix} \text{Grupo 1} \\ \text{Grupo 2} \end{matrix} \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

O grupo 1 é formado pelas máquinas A, C e F, e o grupo 2 pelas máquinas B, D e E.

Permutação

O valor do gene representa o objeto a ser agrupado e a posição no cromossomo indica o subconjunto ao qual o objeto pertence. Exemplos:

a) Departamento-local-período (BALAKRISHNAN; CHENG, 2000):

Seja o cromossomo: (243156 342615 342615 342561).

No primeiro período (sublinhado), o departamento 2 está no local 1, e o departamento 4 no local 2. Já no quarto período o departamento 2 encontra-se no local 3, e o departamento 4 no local 2.

b) Ordenação no *grid* (HAMAMOTO; YIH; SALVENDY, 1999; ADEL EL-BAZ, 2004):

O valor do gene representa o departamento, e a posição representa a ordem em que os departamentos são colocados no *grid*, segundo um caminho especificado. Seja o cromossomo:

(9 1 5 6 4 3 8 7 2). Uma possível ordenação no *grid* é mostrada na figura 3.3.

El-Baz (2004)

Figura 3.3 – Ordenação no *grid*

c) Ordenação no *grid* com separadores (AZADIVAR; WANG, 2000):

O valor do gene representa a máquina, e os símbolos * e + representam cortes horizontal e vertical respectivamente. Seja o cromossomo: (1 2 * 3 4 5 * 6 + + *). O arranjo físico correspondente é mostrado na figura 3.4.

Azadivar e Wang (2000)

Figura 3.4 – Ordenação no *grid* com separadores

3.3 Geração da população inicial

A população inicial deve conter cromossomos representativos do espaço de soluções. A literatura (LACERDA; CARVALHO; LUDERMIR, 1999; COLE, 1998) apresenta várias formas de geração da população inicial. A mais comum é a geração aleatória. A população inicial também pode ser gerada uniformemente, com pontos igualmente espaçados no espaço de busca. Outra forma é gerar a primeira metade da população aleatoriamente e o restante da população a partir da primeira metade, pela troca dos valores dos *bits* dos cromossomos (GORGULHO JÚNIOR, 2006; LACERDA; CARVALHO; LUDERMIR, 1999). Também há

a geração chamada *seeding*. Esta é uma geração tendenciosa para regiões promissoras do espaço de busca, que consiste em colocar na população inicial soluções encontradas por outros métodos de otimização.

3.4 Função de aptidão

Em um algoritmo genético, o papel da função de aptidão é atribuir uma nota a cada cromossomo e assim avaliar a qualidade da solução. A função de aptidão também é usada para selecionar os cromossomos para reprodução, ou seja, cromossomos mais aptos têm maior chance de passar o material genético para as próximas gerações. Como pondera Cole (1998), o sucesso do algoritmo genético é dependente da escolha da função de aptidão.

Algoritmos genéticos trabalham com problemas de maximização, portanto, os cromossomos que representam as melhores soluções possuem maiores valores da função de aptidão. Como o processo de evolução tende a reter material genético de cromossomos com maior aptidão, a escolha apropriada da função de aptidão irá aumentar a probabilidade de reter características associadas a soluções ótimas ou quase ótimas (COLE, 1998).

Um ponto a se considerar quando da escolha da função de aptidão é a escala (GOLDBERG, 1989; COLE, 1998). Poucos indivíduos podem apresentar valores excessivamente altos de aptidão (mas não ótimos), enquanto muitos indivíduos podem possuir valores baixos. O algoritmo genético selecionará para reprodução os indivíduos com os valores muito altos de aptidão, que irão dominar a população. Isso pode ocasionar a convergência prematura, ou seja, os demais indivíduos tendem a desaparecer e o algoritmo achará rapidamente uma solução que representa um máximo ou mínimo local, sem ter explorado suficientemente o espaço de busca. Este problema pode ser contornado de algumas formas: limitar o número de filhos por cromossomo; aumentar a taxa de mutação para

melhorar a diversidade ou evitar a inserção de filhos duplicados na população (LACERDA; CARVALHO; LUDERMIR, 1999).

Outro problema ocorre quando os indivíduos possuem valores de aptidão muito próximos, a chamada supercompressão do intervalo da aptidão. Isto faz com que qualquer cromossomo tenha a mesma probabilidade de seleção, tornando o processo praticamente aleatório. Assim, o algoritmo tem dificuldade de melhorar a solução e converge lentamente. Este problema pode ser resolvido expandindo-se o intervalo da função de aptidão, através de uma forma de mapeamento da função objetivo. Entre os métodos de mapeamento da função objetivo destacam-se o *ranking* linear, o *ranking* exponencial e o escalonamento linear (LACERDA; CARVALHO; LUDERMIR, 1999; COLE, 1998).

3.5 Operadores genéticos

A função dos operadores genéticos é fazer com que a população inicial sofra sucessivas transformações e o algoritmo genético melhore no tempo (GOLDBERG, 1989). Os operadores de cruzamento e mutação são os principais mecanismos de busca de um algoritmo genético em regiões desconhecidas do espaço de soluções (LACERDA; CARVALHO; LUDERMIR, 1999). Esses dois operadores são apresentados a seguir.

3.5.1 Operador de cruzamento

O operador de cruzamento é aplicado a um par de cromossomos (cromossomos pais) para gerar descendentes, em geral, dois cromossomos filhos. O cruzamento pode ocorrer ou não, dependendo de uma taxa de probabilidade. Se o cruzamento não ocorrer, os cromossomos filhos serão cópias de seus pais. Existem diversos tipos de operadores de

cruzamento. Entre os mais usados estão o de ponto único, o de dois pontos e cruzamento uniforme.

No cruzamento de ponto único cada cromossomo pai é cortado em uma posição aleatória e os *bits* à direita são trocados, como na figura 3.5:

Figura 3.5 – Cruzamento de ponto único

No cruzamento de dois pontos, são selecionadas duas posições aleatoriamente nos cromossomos pai e os *bits* intermediários são trocados, como na figura 3.6:

Figura 3.6 – Cruzamento de dois pontos

No cruzamento uniforme é gerada uma máscara com valores 0 ou 1, aleatórios. Se a primeira posição da máscara possuir valor 1, então o gene da primeira posição do pai1 é copiado para o filho1. Caso contrário, o gene da primeira posição do pai2 é copiado para o filho1. O processo se repete para todos os genes do filho1. Para gerar o filho2, o processo é invertido, ou seja, se a primeira posição da máscara for 1 então é copiado o primeiro gene do pai2. Se a primeira posição da máscara for 0, então é copiado o primeiro gene do pai1. A figura 3.7 mostra o cruzamento uniforme.

Máscara:	1	1	0	0	1	0
Pai 1:	3	1	1	2	3	3
	↓	↓			↓	
Filho 1:	3	1	2	3	3	1
			↑	↑		↑
Pai 2:	1	1	2	3	1	1

Figura 3.7 – Cruzamento uniforme

3.5.2 Operador de mutação

A função do operador de mutação é introduzir novo material genético na população, de forma a possibilitar que o algoritmo realize busca em regiões ainda não exploradas no espaço de soluções. Com isso torna-se possível sair de ótimos locais indesejáveis e talvez alcançar o ótimo global.

A mutação, em geral realizada após o cruzamento, também é aplicada de acordo com uma taxa de probabilidade e consiste em alterar o valor de um ou mais genes do cromossomo, como mostra a figura 3.8.

(1 3 <u>1</u> 2 1 <u>3</u>)	→	(1 3 3 2 1 1)
-----------------------------------	---	-----------------------------------

Figura 3.8 - Mutação

3.6 Processo de seleção

O processo de seleção refere-se à escolha dos indivíduos (cromossomos pais) que irão gerar descendentes (cromossomos filhos) através dos operadores de cruzamento e mutação. Os métodos de seleção comumente usados são pela Roleta, por *Ranking* e por Torneio.

Seleção pela Roleta

O método de seleção dos indivíduos pela Roleta (*Roulette Wheel*) foi proposto por Goldberg (1989). Um modo fácil de entender o método da Roleta é pensar em uma circunferência dividida em n faixas, onde n é o número de cromossomos. Cada faixa é rotulada de 1 a n , e o comprimento do arco é proporcional à aptidão do cromossomo. Isto é mostrado na figura 3.9 para quatro cromossomos.

Figura 3.9 - Roleta

Se girarmos a roleta e jogarmos uma pequena bola, esta irá parar em um ponto aleatório, determinando o cromossomo selecionado. Cromossomos com faixa mais larga (maior aptidão) têm maior chance de serem selecionados para posteriormente sofrerem cruzamento. Geralmente, os cromossomos pais são selecionados com probabilidade p_i dada por (LACERDA; CARVALHO; LUDERMIR, 1999):

$$p_i = \frac{f_i}{\sum_{i=1}^N f_i} \quad \text{onde} \quad (3.1)$$

f_i é a aptidão do cromossomo e N é o número de elementos da população.

Um modo prático de fazer a seleção é ordenar os cromossomos em ordem decrescente de aptidão e calcular a aptidão acumulada. Seja o exemplo fictício da tabela 3.1 a seguir.

Tabela 3.1 – População inicial

Posição na Ordenação i	Cromossomo	Aptidão f_i	Aptidão Acumulada $\sum_{k=1}^i f_k$
1	11001000	2,00	2,00
2	00101011	1,75	3,75
3	00011101	1,50	5,25
4	10011011	1,00	6,25

Adaptado de Lacerda, Carvalho e Ludermir (1999)

Na tabela 3.1, o cromossomo da primeira posição ordenada possui aptidão igual a 2,00 (dois), o segundo cromossomo na ordenação, aptidão igual a 1,75, e assim por diante.

Em seguida gera-se um número aleatório r (de uma distribuição uniforme) entre $[0, \text{AptidãoAcumulada}]$. O cromossomo selecionado será aquele com aptidão acumulada maior que r . Neste exemplo suponha $r = 5,00$. Então, conforme a tabela 3.1, o cromossomo 3 é selecionado para reprodução. Este procedimento é repetido até se completar o número de indivíduos para reprodução.

Seleção por *Ranking*

Os cromossomos são colocados em ordem decrescente da função objetivo. O melhor cromossomo recebe uma aptidão arbitrária, por exemplo, 2,0 (dois), e o último 0 (zero). A aptidão (f_i) dos demais cromossomos é obtida interpolando esses extremos por uma reta, ou seja, $f_i = 2(N-i)/(N-1)$, onde N é o tamanho da população e i é a posição do cromossomo no *ranking*. São selecionados para reprodução os primeiros n (número necessário) indivíduos com maior f_i (LACERDA; CARVALHO; LUDERMIR, 1999).

Seleção por Torneio

Na seleção por torneio, cada cromossomo é selecionado para o grupo de reprodutores da seguinte forma: escolhe-se aleatoriamente (com probabilidades iguais) K indivíduos da

população (geralmente $K = 2$) e o com melhor aptidão é selecionado. O processo se repete até preencher o grupo de reprodutores. Também há a seleção por torneio com probabilidades diferenciadas. Nesta, em um torneio com n indivíduos, o primeiro ganha o torneio com probabilidade p (geralmente $0,5 < p < 1$), o segundo com probabilidade $p(1-p)$, o terceiro com probabilidade $p(1-p)^2$ e assim por diante. Aumentando-se a probabilidade p do primeiro vencer, ou o número de indivíduos (n) do torneio, indivíduos com aptidão acima da média terão mais chance de serem selecionados para reprodução (LACERDA; CARVALHO; LUDERMIR, 1999).

3.7 Geração da nova população

Consiste no processo de substituição dos cromossomos, ou seja, em como incorporar os cromossomos filhos à população. As formas de substituição geralmente usadas são (LACERDA; CARVALHO; LUDERMIR, 1999): substituição geracional, substituição geracional com elitismo e substituição de estado uniforme.

Na substituição geracional, toda a população é substituída a cada geração, ou seja, N cromossomos filhos substituem N cromossomos pais.

Na substituição geracional com elitismo, os K melhores pais são transferidos integralmente para a nova população. Em geral K é igual a 1 ou 2. Este método assegura que os melhores cromossomos passarão para a geração seguinte, mas por outro lado pode restringir o espaço de busca do algoritmo genético.

Na substituição de estado uniforme (*steady-state*) apenas 2 (ou 1) filhos são gerados e podem substituir os pais, ou os dois piores indivíduos da população, ou os dois indivíduos mais velhos (que estão a mais tempo na população e, portanto, já transmitiram seus genes).

3.8 Critérios de parada

O critério de parada refere-se ao momento em que se deve parar o processo de evolução do algoritmo genético e adotar o cromossomo com maior aptidão como a melhor solução. Os critérios comumente usados são número de gerações, perda da diversidade e convergência. No primeiro critério, o algoritmo genético termina quando se atingiu um número de gerações especificado pelo usuário. No critério perda da diversidade, o AG termina quando uma porcentagem alta da população, geralmente acima de 90%, representa a mesma solução. Pelo critério da convergência, o AG pára quando não há melhoramento no cromossomo de maior aptidão por determinado número de gerações. A chegada ao valor ótimo da função de aptidão também pode ser adotada como critério de parada quando o mesmo é conhecido.

3.9 Parâmetros

Os principais parâmetros de um algoritmo genético cujos valores devem ser definidos pelo usuário são: tamanho da população, taxa de aplicação dos operadores genéticos e valor do critério de parada. Pequenas populações (poucos indivíduos ou cromossomos) podem não ser representativas do espaço de soluções e levar à convergência rápida do algoritmo para ótimos locais. Grandes populações podem incorrer em maior tempo computacional. A taxa de aplicação de operadores genéticos refere-se à probabilidade de aplicação dos operadores de cruzamento e mutação. Baixa taxa de cruzamento faz com que a convergência seja mais lenta, pois é preciso mais tempo para que haja a melhora do material genético (soluções). Alta taxa de mutação pode destruir a informação contida no cromossomo e, portanto, destruir boas

soluções. Um valor pequeno do critério de parada pode não ser suficiente para garantir a convergência do algoritmo.

3.10 Algoritmos genéticos aplicados ao projeto de arranjo físico

Venugopol e Narendran (1992) foram uns dos primeiros autores a aplicarem algoritmos genéticos ao projeto de arranjo físico. Estes autores propuseram um AG para o problema de agrupamento peça-máquina com múltiplos objetivos. Este problema também é conhecido como problema de formação de células de manufatura. A função objetivo contempla a minimização do volume de movimento intercelular e a minimização da variação da carga de trabalho dentro da célula. O tamanho da célula e a designação única de máquinas às células são restrições do problema. Como os objetivos são conflitantes, não existe solução ótima única para o problema. Portanto, tenta-se alcançar uma solução satisfatória, ou seja, quando não existe outra solução que possa melhorar um objetivo sem detrimento ao outro objetivo.

A representação do cromossomo usada pelos autores é a grupo-número. No cromossomo chamado máquina-célula a posição do gene representa a máquina e seu valor, a célula a qual a máquina pertence. O tamanho do cromossomo equivale ao número de máquinas. Dois operadores de cruzamento são usados, de ponto único e de dois pontos. A mutação consiste na troca aleatória do valor do gene (alteração da célula a qual a máquina está alocada). Após a aplicação do cruzamento é adotada uma política de substituição da população: se todos os descendentes forem melhores que os pais, toda a população é substituída, ou senão os melhores descendentes da geração atual substituem em igual número os piores pais da geração anterior. Para os demais descendentes é estabelecida uma probabilidade de aceitação do cromossomo. Com o objetivo de prevenir a convergência

prematura, após a política de substituição é feito o monitoramento da diversidade da população. Caso a diversidade seja menor que um *threshold* é aplicado o operador de mutação em um cromossomo escolhido aleatoriamente.

Os autores apresentaram um exemplo numérico de 30 peças e 15 máquinas e o melhor cromossomo gerou um agrupamento de três células de manufatura. Este estudo considerou vários aspectos reais como tempo de processamento para cada peça em diferentes máquinas, demanda de cada peça e carga de trabalho nas máquinas. Os resultados mostraram um conjunto de soluções satisfatórias, que permitem ao projetista a escolha da melhor alternativa.

Morad e Zalzal (1996) desenvolveram um algoritmo genético aplicável aos problemas de formação de células de manufatura e programação da produção.

Para o problema de formação de células de manufatura o tamanho do cromossomo corresponde ao número de máquinas. A posição do gene representa o número da máquina e o valor do gene a célula a qual a máquina pertence. O número desejado de células é especificado *a priori*. A função objetivo é composta por três termos. O primeiro visa a minimização do movimento intercelular, o segundo visa a minimização da variação da carga de trabalho dentro das células e o terceiro termo objetiva a maximização da similaridade das máquinas dentro da célula. O número mínimo e máximo de máquinas em cada célula é definido como uma restrição do problema. Para testes foram usados três exemplos da literatura, com 15 máquinas e 10 peças, 14 máquinas e 24 peças e 16 máquinas e 43 peças. Os resultados mostraram que quando o tempo de processamento, a capacidade de máquina e a demanda da peça são constantes, os agrupamentos obtidos foram os mesmos da literatura. Quando esses fatores variam, os agrupamentos também mudam, mostrando sua influência na formação das células.

Para o problema de programação da produção a função objetivo visava minimizar o *tardiness* (atraso das tarefas) e minimizar o *makespan* (duração total da programação). O cromossomo é representado por duas partes. A primeira indica o número de tarefas a serem feitas em cada lote, e a segunda parte consiste das tarefas e da ordem de processamento. Para tanto, o número de lotes deve ser pré-estabelecido ou sorteado aleatoriamente. Os operadores agem em cada parte do cromossomo separadamente. Na primeira parte são feitas a troca do valor do gene e a troca da posição. Na segunda parte são aplicados dois tipos de cruzamento e dois de mutação, baseados na ordem e na posição dos genes. O algoritmo é capaz de encontrar a sequência ótima de tarefas e também determinar o número de tarefas em cada lote.

Os autores destacam a vantagem do uso do algoritmo em ambos os problemas, sendo possível fornecer mais de uma solução para o usuário.

Gravel, Nsakanda e Price (1998) apresentaram um algoritmo genético para o problema de formação de células de manufatura com rotas alternativas. O objetivo era minimizar o movimento intercelular e a variação da carga de trabalho dentro da célula. Foi utilizado um algoritmo genético de duplo *loop*. O *loop* externo trabalha com cromossomos que representam a alocação de máquinas às células, e o *loop* interno utiliza os cromossomos que representam a alocação de peças às rotas. Cada célula gerada pelo *loop* externo é avaliada de acordo com o melhor cromossomo peça-rota encontrado no *loop* interno. Para o cromossomo peça-rota também foi usado um método de melhoria local através da troca de alguns genes do cromossomo. Dois tipos de cruzamento foram implementados, de ponto único e tipo OX. Testes foram realizados com 15 máquinas e 30 peças e o algoritmo foi eficiente em encontrar boas soluções comparadas com otimização por um único critério.

Balakrishnan e Cheng (2000) aplicaram algoritmo genético ao problema de arranjo físico de bloco com período e demanda variáveis. O objetivo foi o de minimizar o custo de rearranjo e o custo do fluxo de materiais ao longo do horizonte planejamento. Para tanto aplicaram o AG em dois *loops*.

No *loop* interno é feito cruzamento ponto-a-ponto, realizado um teste de viabilidade dos indivíduos e aplicada a mutação. O cruzamento produz $2((n * t) - 1)$ descendentes, onde n é o número de departamentos dentro de um período e t é o número de períodos do horizonte de planejamento. O cromossomo pai com máximo custo é substituído pelo descendente com mínimo custo. A mutação é feita no descendente com o mínimo custo. Primeiro é escolhido aleatoriamente um período, e então, dois departamentos são escolhidos aleatoriamente para troca. O *loop* interno termina quando a diferença entre os melhores descendentes em duas gerações sucessivas é menor que um limitante.

O *loop* externo consiste na substituição dos piores indivíduos na população (que podem ser tanto da população antiga, como do *loop* interno ou gerados aleatoriamente). O *loop* externo termina quando é atingido certo número de gerações.

Os testes foram conduzidos com 5 e 10 períodos de planejamento e com três tamanhos de arranjo físico: 6, 15 e 30 departamentos. Os resultados mostraram que o algoritmo proposto foi melhor que o comparado da literatura principalmente para os problemas de maior porte.

Kumar e Shanker (2000) aplicaram algoritmo genético para seleção do tipo de peça e carga de máquina em um sistema flexível de manufatura (FMS). O problema é formulado através de programação inteira mista e o algoritmo genético é modelado em termos das variáveis de decisão e das restrições do problema. As hipóteses do ambiente FMS são: existe somente uma máquina de cada tipo; cada máquina pode fazer várias operações; não há

agrupamento de máquina; uma máquina possui por vez uma cópia de um tipo de ferramenta; cada operação pode ser feita em máquinas alternativas; não há divisão de tarefas.

A população inicial é gerada de modo a reduzir a inviabilidade de soluções. Isso é feito através da codificação do cromossomo que contempla a maioria das restrições do problema e possui genes dependentes do contexto. Um problema com 3 tipos de peça, cada uma com 3, 2 e 5 operações respectivamente pode ser representada pelo cromossomo a seguir:

(**1** 2 1 3 **0** 0 0 **1** 3 1 2 2 1)

Os genes em negrito são binários (um para cada operação). Recebem valor 0 (zero) se a peça não for selecionada e 1 caso contrário. Os genes seguintes ao em negrito representam a operação: a posição indica a máquina e o valor do gene o tipo de ferramenta.

Dois tipos de cruzamento foram testados, de dois pontos e uniforme. O cruzamento é feito em duas etapas. Na primeira etapa é trocada informação do tipo de peça e na segunda é feita a troca da informação das operações para aquele tipo de peça. A mutação consiste na troca aleatória do valor de um gene.

O algoritmo genético permite a seleção de uma entre quatro funções objetivo. São elas: maximizar o número de tipos de peças selecionadas; maximizar o número total de peças selecionadas; maximizar a utilização média das máquinas e maximizar a utilização e o balanceamento de carga. O critério de parada foi baseado na convergência da população, ou seja, a diferença entre o melhor *fitness* e o *fitness* médio deveria ser menor que 0.001. Para todas as funções objetivo o algoritmo genético foi capaz de encontrar a solução ótima e várias vezes encontrou soluções quase-ótimas. Um estudo dos operadores e parâmetros do algoritmo mostrou que a utilização de função de penalidade, o tamanho da população e a probabilidade de mutação influenciam de modo significativo no desempenho do algoritmo.

Tseng e Yang (2001) desenvolveram um algoritmo genético para o problema de agrupamento (*clustering problem*). O algoritmo pode ser usado de dois modos. No primeiro o usuário pode controlar o resultado do agrupamento através da variação do valor de um parâmetro (w). Pequenos valores do parâmetro resultam em vários agrupamentos compactos. Valores maiores de w geram poucos agrupamentos, mas com vários elementos cada. No segundo modo o algoritmo resolve o problema de agrupamento automático, ou seja, a melhor solução corresponde àquela que possui o melhor número de grupos.

Uma estratégia heurística é usada na busca pelos melhores agrupamentos. O algoritmo, chamado *Clustering*, é composto por dois estágios. No primeiro estágio é implementado o algoritmo do vizinho mais próximo (*single linkage*), com o objetivo de reduzir a dimensão dos dados de entrada e com isso reduzir o tempo computacional do segundo estágio. O primeiro estágio gera pequenos agrupamentos (subconjuntos iniciais), que servirão como entradas para o segundo estágio. Este, por sua vez, é o algoritmo genético propriamente dito. A população inicial do AG é gerada aleatoriamente. A codificação do cromossomo é binária e o tamanho do cromossomo equivale ao número de subconjuntos (agrupamentos iniciais) obtidos no primeiro estágio. Se o valor do gene for 1, o subconjunto correspondente continuará a existir. Se o valor do gene for 0, o agrupamento correspondente deixará de existir e seus elementos serão adicionados a outros agrupamentos cujo valor seja 1. O AG transforma os subconjuntos iniciais do primeiro estágio em agrupamentos maiores, de modo aglomerativo. A função objetivo considera a distância entre os elementos de um mesmo agrupamento (D_{intra}) e a distância entre os agrupamentos (D_{inter}). A influência de cada distância é controlada através do aumento ou diminuição do valor de uma constante. A seleção é feita pelo método da roleta.

No cruzamento é selecionado um par de cromossomos e depois são gerados dois números aleatórios no intervalo $[1, m]$ onde m é o tamanho dos cromossomos. Os

cromossomos são divididos nos pontos sorteados e o segmento entre eles é trocado. A mutação consiste na alteração do valor de um gene (0 para 1 ou vice-versa) de acordo com dada probabilidade. Esta mutação foi aplicada *bit-a-bit*, ou seja, a cada gene do cromossomo. Como critério de parada foi usado o número de gerações. A busca pelo número ideal de agrupamentos é feita por uma heurística que, para isso, altera constantemente alguns parâmetros do algoritmo genético.

O algoritmo foi comparado aos métodos de agrupamento *k-means*, *single linkage* e *complete linkage*. O número de agrupamentos obtidos foi o mesmo, mas a medida de comparação (distância média do centro) foi muito menor, o que mostra que o algoritmo gerou agrupamentos mais compactos.

Uddin e Shanker (2002) propuseram uma metodologia por algoritmos genéticos para o problema de agrupamento peça-máquina. O modelo designa rotas (peças) e máquinas às células simultaneamente, considerando múltiplas rotas para cada peça. O objetivo é minimizar o movimento intercelular através da minimização do número de células visitadas.

A representação do cromossomo é a grupo-número. Existem dois tipos de cromossomos, o cromossomo máquina-célula e o peça-rota. O cromossomo máquina-célula (2 3 1 1 1 2 1) é interpretado da forma: célula 1: M3 M4 M5 M7; célula 2: M1 M6; célula 3: M2. Já o cromossomo peça-rota (2 3 1 1 2 1) indica: peça1, rota2; peça 2, rota3, peça 3, rota 1, e assim por diante. O operador de cruzamento usado foi o de ponto único. A mutação consistia na alteração do valor de um gene. O algoritmo foi comparado com três problemas da literatura com 4, 6 e 20 máquinas e 5, 10 e 20 peças respectivamente. O algoritmo encontrou a solução ótima dentro de poucas iterações e com mesmo volume de movimento intercelular.

Garai e Chaudhuri (2004) também aplicaram algoritmos genéticos ao problema de agrupamento automático. O algoritmo é composto por três partes: CDA (*Cluster Decomposition Algorithm*), ACCA (*Adjacent Cluster Checking Algorithm*) e HCMA (*Hierarchical Cluster Merging Algorithm*).

O CDA divide o conjunto de dados em subconjuntos iniciais segundo o método do vizinho mais próximo, assim como no trabalho de Tseng e Yang (2001). Entretanto, segundo Garai e Chaudhuri (2004), o algoritmo de Tseng e Yang (2001) falha se os agrupamentos contém subconjuntos parcialmente ou totalmente confinados um dentro do outro. Para resolver esse problema os autores desenvolveram o algoritmo ACCA, que verifica a adjacência dos subconjuntos e evita que sejam gerados subconjuntos um dentro do outro. Em seguida é aplicado o HCMA.

O HCMA é o algoritmo genético propriamente dito, que combina os agrupamentos fragmentados até formar o número ideal de grupos. O algoritmo seleciona dois indivíduos da população pelo método da roleta e realiza cruzamento de ponto único entre eles, gerando dois descendentes. A mutação utilizada é a *bit-a-bit* com probabilidade dada em função de uma probabilidade de mutação inicial e do número de gerações que o algoritmo executou. A função objetivo é baseada nas distâncias inter e intra-agrupamento. Como critério de parada é especificado o número de agrupamentos desejado ou número de gerações (para agrupamento automático). O algoritmo foi comparado a três outros da literatura e obteve bons resultados.

Adel El-Baz (2004) implementou um algoritmo genético para o problema de arranjo físico determinístico com o objetivo de minimizar o custo de movimentação de materiais. O modelo pode ser aplicado a vários tipos de arranjos físicos como em linha, semi-circular, em *loop* e múltiplas linhas. Cada gene do cromossomo representa uma máquina e a ordenação dos genes no cromossomo representa a ordem em que as máquinas serão dispostas no *grid* (chão

de fábrica), conforme um caminho pré-especificado para cada tipo de arranjo físico. Espaços vazios no *grid* também podem ser representados no cromossomo. O tipo de cruzamento usado foi o de dois pontos e a mutação consistia na troca aleatória de dois genes no cromossomo.

Os resultados foram comparados com *benchmarks* da literatura para dois tamanhos de problemas, o primeiro com 9 máquinas em um *grid* 3x3 (não permitia espaços vazios) e o segundo com 24 máquinas e 40 peças em um *grid* 5x6 (30 posições). O modelo proposto obteve a solução ótima para ambos os problemas em maior número de vezes que os modelos da literatura.

Gonçalves Filho, Gorgulho Júnior e Argoud (2004) apresentaram um algoritmo genético para o problema de formação dos grupos de máquinas em manufatura celular. A codificação tradicional grupo-número (célula-máquina) é transformada em codificação por grupos de células com separadores. A função objetivo minimiza o volume de movimentos entre as células e a variação da carga total dentro da célula. A população inicial é gerada aleatoriamente de acordo com um procedimento para não violar as restrições do problema: cada indivíduo gerado é analisado para garantir que cada máquina pertença a somente uma célula e que cada célula seja composta pelo menos por duas máquinas. Os parâmetros de entrada são: o número de tipos de máquinas disponíveis (M) e o número de células a serem formadas (K). A taxa de elitismo usada foi de 5%. O processo de seleção adotado foi o da roleta. Antes do cruzamento é aplicado o operador de inversão, que consiste na inversão da ordem das células no cromossomo.

O operador de cruzamento implementado trabalha com grupos, trocando células completas dos cromossomos pais para gerar os filhos. Após o cruzamento é aplicado um mecanismo de correção dos descendentes se necessário. Dois operadores de mutação foram desenvolvidos. No primeiro sorteia-se uma célula, e em seguida uma máquina da célula

sorteada, para realizar a troca entre dois genes do cromossomo. No segundo sorteia-se uma fronteira entre duas células e a direção (direita ou esquerda). A mutação consiste em mover a fronteira uma posição na direção sorteada.

Vários testes foram realizados com diferentes números de máquinas e peças. Para 15 máquinas e 30 peças, o resultado foi o mesmo que o apresentado Venugopol e Narendran (1992).

Norman e Smith (2006) apresentaram uma formulação para o problema de arranjo físico de bloco de áreas desiguais e retangulares através de algoritmo genético. Os autores consideraram incerteza na demanda, representada pelo valor esperado e variância da previsão de vendas de cada produto. O roteiro de produção é fixo e somente foi considerado um período de produção. O objetivo é particionar uma região em n sub-regiões, cada uma representando um departamento, de modo a minimizar o custo de transporte entre os departamentos.

Para representar o arranjo físico de bloco foi usada uma formulação por baias flexíveis. Os autores propuseram uma codificação do cromossomo através de chaves aleatórias. O tamanho do cromossomo equivale ao número de departamentos. A cada gene, ou a cada departamento é designada uma chave aleatória segundo uma distribuição uniforme $U(0,1)$. Quando colocadas em ordem crescente, as chaves aleatórias determinam a seqüência dos departamentos através das baias. A divisão através das baias é codificada através da adição de um número inteiro a cada chave aleatória. O cromossomo a seguir representa o arranjo físico da figura 3.10.

(A	B	C	D	E	F	G	H	I	J	K	L	M)
(1.16	2.28	3.49	4.93	2.37	1.19	1.07	1.24	3.74	4.81	2.43	3.55	3.66)

G	B	C	J
A		L	
F	E	M	D
H		I	
	K		

Norman e Smith (2006)

Figura 3.10 – Arranjo físico de bloco com baias flexíveis

Os autores aplicaram o modelo em dois problemas da literatura sendo um com 10 e outro com 14 departamentos. Os resultados se mostraram similares ou melhores que os da literatura.

Jeon e Leep (2006) aplicaram algoritmos genéticos à formação de família de peças. Os autores desenvolveram um novo coeficiente de similaridade baseado no número de rotas alternativas devido à falha de máquina.

O modelo é composto por duas fases. A primeira utiliza o algoritmo genético para identificação de família de peças baseado no coeficiente de similaridade. A função objetivo visa maximizar a soma dos coeficientes de similaridade entre duas peças. A segunda fase emprega modelo de programação inteira mista para designar as máquinas às famílias de peças com objetivo de minimizar custos operacionais (estoque em processo, investimento em máquina, penalidade por adiantamento ou atraso de tarefas).

A representação do cromossomo utilizada foi a grupo-número onde cada gene representa uma peça e o valor do gene indica a família a qual a peça pertence. Foram utilizados operadores de cruzamento e mutação tradicionais como proposto por Goldberg

(1989). Foi apresentado um exemplo em que o modelo foi aplicado a uma amostra de 10 peças e 7 máquinas. O AG identificou 3 famílias de peças que serviram como entradas para a segunda fase de programação inteira mista.

O quadro 3.2 a seguir sumariza as principais características dos artigos revisados e apresentados nesta seção.

Característica	Venugopol e Narendran (1992)	Morad e Zalzala (1996)	Gravel, Nsakanda e Price (1998)	Balakrishnan e Cheng (2000)	Kumar e Shanker (2000)	Tseng e Yang (2001)
Tipo de Problema	Formação de células	1 - Formação de células 2- Programação da produção	Formação de células	Layout de bloco	Seleção do tipo de peça em um FMS	N.A.
Rotas	Fixas	Fixas	Alternativas	Fixas	Alternativas	N.A.
Demanda	Fixa	Fixa	Fixa	Variável	Fixa	N.A.
Período	-	-	-	Variável	-	N.A.
Tipo de agrupamento	K-agrupamentos	K-agrupamentos	K-agrupamentos	K-agrupamentos	K-agrupamentos	K-agrupamentos e automático
FO	Minimizar o movimento intercelular e a variação da carga de trabalho dentro da célula.	1 -Minimizar o movimento intercelular, a variação da carga de trabalho e maximizar a similaridade de máquinas dentro da célula. 2 - Minimizar o <i>tardiness</i> e <i>makespan</i> .	Minimizar movimento intercelular e balancear carga de trabalho dentro da célula.	Minimizar custo de rearranjo e do fluxo de materiais.	Maximizar o número de tipos de peças selecionadas, ou maximizar a utilização das máquinas ou maximizar a utilização e o balanceamento de carga.	Distância inter e intra-agrupamentos
Restrições	Tamanho da célula e designação única de máquinas às células.	Número mínimo e máximo de máquinas em cada célula.	No mínimo duas máquinas por célula	O mesmo departamento só pode ter uma ocorrência em cada período.	Peças do mesmo tipo possuem diferentes rotas; alocação de ferramental; capacidade de armazenagem de ferramentas; carga de máquina.	-
Representação Cromossomo	Grupo-número	Grupo-número	Grupo-número	Permutação	Inteira	Binária
Seleção	Aleatória	Ranking	Aleatória	Aleatória	Roleta	Roleta
Cruzamento	Ponto único e dois pontos	Baseado na ordem e posição dos genes	Ponto Único e OX	Ponto a ponto	Troca referente ao tipo de peça e à operação	Dois pontos
Mutação	Troca aleatória do valor de um gene	Baseado na ordem e posição dos genes	Troca aleatória do valor de um gene	No descendente com menor custo	Troca aleatória do valor de um gene	Troca aleatória do valor de um gene
Critério de Parada	Número de gerações	Número de gerações	Perda da diversidade	Número de gerações e perda da diversidade	Convergência	Número de gerações
Outros Parâmetros	Tamanho da população: 20 Prob. cruzamento: 0,60 Prob. mutação: 0,005	1-Tamanho da população: 40 Prob. Cruzamento: 0,70 Prob. Mutação: 0,05 2 -Tamanho da população: 100 Prob. Cruzamento: 0,80 Prob. Mutação: 0,50	Tamanho população: 10	Tamanho da população: 500 Prob. Mutação: 0,002	Tamanho da população: 20; 40 Prob. cruzamento: 0,70; 0,75; 0,90; 0,95; 0,99 Prob. mutação: 0,01; 0,05; 0,10; 0,15	Tamanho população: 50 No. de gerações: 100 Prob. cruzamento: 0,80 Prob. mutação: 0,05
Tamanho do teste	30 peças e 15 máquinas	16 máquinas e 43 peças	30 peças e 15 máquinas	10 períodos e 30 departamentos	5 a 9 máquinas 10 a 30 tipos ferramentas 5 a 15 tipos de peças 1 a 4 processos alternativos por operação	20000 vetores (Timit Database)

continua

Quadro 3.2 - Principais características dos artigos revisados

continuação

Característica	Uddin e Shanker (2002)	Garai e Chaudhuri (2004)	Adel El Baz (2004)	Gonçalves Filho, Gorgulho Júnior e Argoud (2006)	Norman e Smith (2006)	Jeon e Leep (2006)
Tipo de Problema	Formação de células	N.A.	Layout de Bloco	Formação de grupos de máquinas	Layout de Bloco	Identificação de família de peças
Rotas	Alternativas	N.A.	Fixas	Fixas	Fixas	Alternativas
Demanda	Fixa	N.A.	Fixa	Fixa	Considera incerteza	Variável
Período	-	N.A.	-	-	-	2
Tipo de agrupamento	K-agrupamentos	K-agrupamentos e automático	K-agrupamentos	K-agrupamentos	K-agrupamentos	automático
FO	Minimizar movimento intercelular	Distância inter e intra-agrupamentos	Minimizar custo de movimentação de materiais	Minimizar o movimento intercelular e a variação da carga de trabalho dentro da célula.	Minimizar custo de transporte entre departamentos	Maximização da soma do coeficiente de similaridade entre duas peças.
Restrições	Número mínimo e máximo de máquinas em cada célula.	-	-	Cada máquina pertence somente a uma célula e cada célula deve conter pelo menos duas máquinas.	-	O problema de formação de células mediante demanda e período variáveis é resolvido por programação inteira mista.
Representação Cromossomo	Grupo-número	Binária	Permutação	Grupo-número	Chave aleatória: seqüência e área dos departamentos	Grupo-número
Seleção	Aleatória	Roleta	Monte Carlo	Roleta	-	Roleta
Cruzamento	Ponto único	Ponto único	Dois pontos	Ponto único: troca de grupos	-	Ponto único
Mutação	Troca aleatória do valor de um gene	Troca aleatória do valor de um gene	Troca aleatória de dois genes	1 - Troca de dois genes 2 - Mudança da fronteira da célula	-	Troca aleatória do valor de um gene
Critério de parada	Número de gerações	Número desejado de agrupamentos ou número de gerações	Número de gerações ou convergência	Número de gerações	-	Número de gerações
Outros Parâmetros	Tamanho população: 20 Número gerações: 400 Prob. cruzamento: 0,95 Prob. mutação: 0,05	Tamanho população: 50 No. de gerações: 100 Prob. cruzamento: [0,5-0,9] Prob. mutação: [0,002-0,005]	Tamanho população: 100; 200 Número gerações: 40 Prob. cruzamento: 0,6; 0,9 Prob. mutação: 0,001; 0,1	Tamanho população: 40 Número de gerações: 200 Prob. cruzamento: 0,85 Prob. mutação 1: 0,10 Prob. mutação 2: 0,05 Prob. inversão: 0,10	-	Tamanho população: 20 Número de gerações: 500 Prob. cruzamento: 0,80 Prob. mutação: 0,01
Tamanho do teste	20 máquinas e 20 peças	Íris Database	24 máquinas e 40 peças	15 máquinas e 30 peças	14 departamentos	10 peças e 7 máquinas

Quadro 3.2 - Principais características dos artigos revisados

A maioria dos artigos pesquisados sobre aplicações de algoritmo genético ao projeto de arranjo físico trata do problema de formação de células de manufatura (VENUGOPOL; NARENDRAN, 1992; MORAD; ZALZALA, 1996; GRAVEL; NSAKANDA; PRICE, 1998; UDDIN; SHANKER, 2002; GONÇALVES FILHO; GORGULHO JÚNIOR; ARGOUD, 2004). Também foi encontrada aplicação em sistemas flexíveis de manufatura (KUMAR; SHANKER, 2000) e para geração de arranjo físico de bloco (BALAKRISHNAN; CHENG, 2000; ADEL EL-BAZ, 2004; NORMAN; SMITH, 2006). A representação inteira para o cromossomo é a mais usada. Nos trabalhos sobre formação de células, a representação do cromossomo é a grupo-número, na qual a posição do gene refere-se à máquina e seu valor, a célula a qual a máquina é alocada. A função de aptidão visa minimizar o movimento intercelular e/ou a variação da carga de trabalho dentro da célula. Nos trabalhos sobre arranjo físico de bloco, a função de aptidão, em geral, visa minimizar o custo de movimentação de materiais. Demanda fixa e único período de produção são predominantes. Alguns trabalhos consideraram rotas alternativas (KUMAR; SHANKER, 2000; JEON; LEEP, 2006; UDDIN; SHANKER, 2002; GRAVEL; NSAKANDA; PRICE, 1998) e variação na demanda (NORMAN; SMITH, 2006).

O critério de parada em relação ao número de gerações prevalece. Os tipos de cruzamento mais usados foram o de ponto único, de dois pontos e uniforme, com algumas variações conforme cada problema.

Dos artigos revisados, apenas Jeon e Leep (2006) usaram algoritmo genético baseado em coeficiente de similaridade, para a identificação de família de peças. Entretanto, a similaridade entre as peças estava relacionada ao número de rotas alternativas devido à falha de máquina, e não em relação ao número de máquinas comuns requeridas pelas peças ou baseado em sequência de operações.

Em todos os trabalhos revisados sobre aplicações de algoritmos genéticos ao projeto de arranjo físico, o usuário deve especificar *a priori* o número de células ou departamentos desejado. Apenas Tseng e Yang (2001), e Garai e Chaudhuri (2004) desenvolveram algoritmo genético para o problema de agrupamento automático. Entretanto, o AG desenvolvido por estes autores não é específico para projeto de arranjo físico. Testes foram conduzidos em bases de dados como *Iris Plant Database* (plantas) e *Timit Database* (discurso).

Na revisão bibliográfica realizada não foram encontradas aplicações de algoritmos genéticos em projeto de arranjo físico modular, tema desta tese.

3.11 Considerações finais

Este capítulo apresentou os principais conceitos sobre algoritmos genéticos assim como uma revisão sobre aplicações dos mesmos ao projeto de arranjo físico. A seguir é apresentado o modelo do algoritmo genético de agrupamento desenvolvido.

4 O MODELO DESENVOLVIDO

4.1 Considerações iniciais

Este capítulo apresenta o algoritmo genético de agrupamento (AGA) desenvolvido para geração dos módulos básicos para projeto de arranjo físico modular. A seção 4.2 mostra os passos para execução do AGA e um fluxograma com os passos citados. Os principais elementos do algoritmo são explicados adiante. A interface com o usuário e os detalhes da implementação são apresentadas no capítulo 5.

Para melhor compreender os passos para a execução do AGA, é apresentado no quadro 4.1, o pseudo-código de um algoritmo genético típico.

```

/* Seja P(t) a população de cromossomos na geração t.*/
t ← 0
inicializar P(t)
avaliar P(t)
ENQUANTO o critério de parada não for satisfeito FAÇA
t ← t + 1
    selecionar P(t) a partir de P(t+1)
    aplicar cruzamento sobre P(t)
    aplicar mutação sobre P(t)
    avaliar P(t)
FIM ENQUANTO

```

Lacerda, Carvalho e Ludermir (1999)

Quadro 4.1 – Pseudo-código de um algoritmo genético típico

O primeiro passo de um AG típico é a geração aleatória da população inicial de cromossomos, que representam possíveis soluções do problema. A população é avaliada de acordo com uma função objetivo, e cada cromossomo recebe uma nota, denominada aptidão.

Os melhores cromossomos são selecionados e podem sofrer cruzamento e mutação, gerando descendentes para a próxima geração. O processo continua até que uma solução satisfatória seja encontrada (LACERDA; CARVALHO; LUDERMIR, 1999).

4.2 O algoritmo genético de agrupamento (AGA) desenvolvido

O algoritmo genético de agrupamento desenvolvido segue os passos do quadro 4.2.

- Passo 1 - Entrada de dados: peças e seqüência de operações.
- Passo 2 - Encontre as subsequências de operações (máquinas) consecutivas comuns entre todos os pares de seqüência.
- Passo 3 - Gere a matriz de similaridade entre subsequências comuns.
- Passo 4 - Escolha a forma de codificação do cromossomo, se R1, R2 ou R3 e o método de aplicação dos operadores genéticos, se probabilístico ou por faixa percentual.
- Passo 5 - Gere a população inicial.
- Passo 5.1 - Defina intervalo para o valor de K (número de grupos).
- Passo 5.2 - Inicie o indivíduo.
- Passo 5.2.1 - Plante as sementes nos grupos.
- Passo 5.2.2 - Agrupe as demais subsequências de acordo com a codificação do cromossomo usada: se codificação R1 ou R2 faça itens a) e b); se R3, faça o item c).
- a) Para cada subsequência não agrupada (D_i), procure pelo grupo (G_j) com o qual D_i possuir a maior similaridade média ($\overline{Sim}$), e insira D_i em G_j .
- b) Repita o item a) até que todas as subsequências sejam agrupadas. Mostre os agrupamentos formados. Vá para o passo 6.
- c) Cada subsequência não agrupada (D_i) é inserida em um grupo escolhido aleatoriamente. Mostre os agrupamentos formados. Vá para o passo 6.
- Passo 6 - Calcule o valor da função de aptidão para cada cromossomo.
- Passo 7 - Teste o critério de parada. Em caso afirmativo, mostre os cromossomos em ordem decrescente de aptidão e termine o algoritmo. Em caso negativo vá para o passo 8.
- Passo 8 - Determine as m melhores soluções (Regra do Elitismo) da população atual e leve para a nova população. Faça os passos 9 a 15 até que complete a nova população.
- Passo 9 - Selecione os cromossomos para cruzamento usando o procedimento da roleta.

continua

Quadro 4.2 – Algoritmo genético de agrupamento (AGA)

continuação

- Passo 10 - Se o método de aplicação dos operadores for probabilístico vá para o passo 11, se for por faixa percentual vá para o passo 13.
- Passo 11 - Aplique o operador de cruzamento se o cruzamento ocorrer. Corrija os cromossomos descendentes se necessário.
- Passo 12 - Aplique os operadores de mutação nos descendentes se a probabilidade de mutação ocorrer. Vá para o passo 15.
- Passo 13 - Aplique o operador de cruzamento aos $x\%$ (taxa de cruzamento) primeiros cromossomos sorteados e corrija os descendentes se necessário.
- Passo 14 - Aplique os operadores de mutação aos restantes dos cromossomos sorteados respeitando as taxas de cada mutação e ordem de seleção.
- Passo 15 - Adicione os descendentes à nova população.
- Passo 16 - Substitua a população atual pela nova e retorne ao passo 6.

Quadro 4.2 – Algoritmo genético de agrupamento (AGA)

A determinação das subsequências comuns, descrita no passo 2 do quadro 4.2 é feita de acordo com o algoritmo mostrado na figura 2.17 no capítulo 2.

A matriz de similaridade entre subsequências de operações comuns descrita no passo 3 do quadro 4.2 é calculada a partir da escolha pelo usuário de um dos coeficientes de similaridade descrito na seção 2.3.5.1 no capítulo 2: coeficiente baseado na Distância de Levenshtein (TAM, 1990), coeficiente LCS – mais longa subsequência comum (ASKIN; ZHOU, 1998) ou coeficiente *Merger* (HUANG, 2003).

A figura 4.1 mostra um fluxograma com os passos descritos no quadro 4.2. Os passos do quadro 4.2 estão identificados no canto superior de cada bloco do fluxograma. Os principais elementos do algoritmo são explicados adiante.

Figura 4.1 – Fluxograma do algoritmo genético de agrupamento (AGA)

4.3 Codificação do cromossomo

O algoritmo genético de agrupamento (AGA) permite a seleção de uma entre três formas de codificação do cromossomo, R1, R2 ou R3. As codificações R1 e R2 foram inspiradas nos trabalhos de Tseng e Yang (2001) e de Garai e Chaudhuri (2004), e a codificação R3 no de Hruschka (2001).

A codificação R1 é usada quando o usuário especifica o número desejado de módulos, portanto, o número de grupos, K , é fixo. O cromossomo é codificado através de números inteiros e zeros. Cada gene representa uma subsequência comum distinta, obtida no passo 2 do quadro 4.2. O tamanho do cromossomo equivale ao número de subsequências comuns. Seja o seguinte exemplo com 8 subsequências e 4 grupos.

(	2	0	0	3	1	0	4	)
	S1	S2	S3	S4	S5	S6	S7	S8

As subsequências sorteadas, S1, S4, S5 e S7 receberam aleatoriamente os valores 1 a 4. Essas subsequências funcionarão como sementes de agrupamentos iniciais. Os grupos serão formados pelas seguintes subsequências sementes: grupo 1: S5; grupo 2: S1; grupo 3: S4 e grupo 4: S7.

A codificação R2 é semelhante à anterior. A única diferença está na adição de um gene na última posição do cromossomo indicando o número de grupos do cromossomo específico. Esta codificação é usada quando o número de módulos não é especificado pelo usuário, ou seja, quando o algoritmo resolve o problema de agrupamento automático. O usuário apenas especifica um valor mínimo e máximo para K (número de grupos). O tamanho do cromossomo é $(n + 1)$, ou seja, número de subsequências comuns encontradas no passo 2 do quadro 4.2, mais um gene representando o número de grupos. Seja o exemplo com 8

subseqüências. O número de grupos, K , é sorteado aleatoriamente no intervalo $1 \leq K \leq 8$. No exemplo, $K = 4$. As subseqüências sementes sorteadas foram S1, S4, S5 e S7:

$$\begin{array}{cccccccccc} (& 2 & 0 & 0 & 4 & 1 & 0 & 3 & 0 & 4) \\ S1 & S2 & S3 & S4 & S5 & S6 & S7 & S8 & K \end{array}$$

Os grupos serão formados pelas seguintes subseqüências sementes: grupo 1: S5; grupo 2: S1; grupo 3: S4 e grupo 4: S7. Nesta codificação cada cromossomo possui um valor de K , sorteado aleatoriamente.

A designação dos genes zerados aos grupos, nas codificações R1 e R2, é feito de acordo com um procedimento descrito na seção 4.4.

A codificação R3 é chamada grupo-rótulo. Esta codificação também é usada quando o usuário não especifica o número de módulos, ou seja, o número de grupos, K , é uma otimização do problema. A diferença da representação anterior está no fato de que em R3 todas as subseqüências são designadas aos grupos aleatoriamente. O tamanho do cromossomo é $(n+1)$, ou seja, número de subseqüências comuns encontradas no passo 2 do quadro 4.2, mais um gene representando o número de grupos. Seja o exemplo com 8 subseqüências e número de grupos, K , variando no intervalo $1 \leq K \leq 8$. Suponha que o valor sorteado de K seja 3.

$$\begin{array}{cccccccccc} (& 2 & 1 & 2 & 1 & 3 & 3 & 2 & 1 & 3) \\ S1 & S2 & S3 & S4 & S5 & S6 & S7 & S8 & K \end{array}$$

Os grupos serão formados pelas subseqüências: grupo 1: S2, S4, S8; grupo 2: S1, S3, S7 e grupo 3: S5, S6.

4.4 Geração da população inicial

A geração da população inicial ocorre de acordo com os passos descritos a seguir.

Passo 1 – Definição do valor de K (número de grupos).

O número de grupos (K) deve variar entre um valor mínimo e máximo definidos pelo usuário, ou seja, $K_{\min} \leq K \leq K_{\max}$. Conforme a codificação do cromossomo, R1, R2 ou R3, algumas restrições devem ser respeitadas.

- a) Restrição 1: $K_{\min} = K_{\max}$ para a codificação R1. Como K é fixo, todos os indivíduos terão o mesmo número de grupos.
- b) Restrição 2: $K_{\min} \neq K_{\max}$ para codificações R2 ou R3. Sorteie o número de grupos (K) para cada indivíduo aleatoriamente conforme distribuição uniforme entre $K_{\min}$ e $K_{\max}$. Para a função de aptidão soma das similaridades médias, $K_{\min} \geq 1$ e $K_{\max} \leq N_{\text{Subseq}}$ (número de subsequências comuns únicas). Para as funções de aptidão largura da silhueta média e taxa do critério da variância, $K_{\min} \geq 2$ e $K_{\max} \leq N_{\text{Subseq}}$. As funções de aptidão são detalhadas adiante.

Passo 2 – Iniciação de um indivíduo.

Neste passo são formados os grupos em cada cromossomo, do seguinte modo:

- a) Plante as sementes nos grupos.

São selecionadas K subsequências aleatoriamente e colocadas uma em cada grupo, para garantir que haja ao menos uma subsequência por grupo.

- b) Agrupe as demais subsequências de acordo com a codificação do cromossomo.

Para codificações R1 ou R2: Para cada subsequência não agrupada (D_i), procure pelo grupo (G_j) com o qual D_i possuir a maior similaridade média ($\overline{Sim}$), e insira D_i em G_j . A similaridade média é calculada pela fórmula:

$$\overline{Sim}(G_j, D_i) = \frac{1}{G_j} \sum_{x \in G_j} M[x, D_i] \quad \text{onde} \quad (4.1)$$

D_i = subsequência não agrupada;

x = subsequência do grupo G_j ;

$|G_j| = \text{n}^\circ \text{ de elementos do grupo } G_j;$

$M = \text{valor da similaridade na matriz de subsequências de operações.}$

Este procedimento foi inspirado nos trabalhos de Tseng e Yang (2001) e Garai e Chaudhuri (2004) que realizaram um pré-agrupamento do conjunto de dados com o objetivo de reduzir o tempo computacional do algoritmo. Entretanto, esses autores trabalharam com distância do centróide e implementaram o método do vizinho mais próximo.

O procedimento proposto, para R1 ou R2, visa formar agrupamentos iniciais com o intuito de guiar o algoritmo genético a partir de boas soluções iniciais.

Dois casos especiais devem ser considerados para as codificações R1 e R2.

- a) Quando a subsequência D_i possui similaridade igual a zero para com todas as outras subsequências e o número de grupos (K) do cromossomo é menor que o máximo, ou seja, $\text{Sim}(D_i) = 0$ e $K < K_{\max}$. Neste caso deve ser criado um novo grupo para D_i .
- b) Quando $\text{Sim}(D_i) = 0$ e $K = K_{\max}$, ou seja, não se pode criar novo grupo, pois K está no máximo. Neste caso, D_i é inserida em um grupo existente aleatoriamente.

Para a codificação R3: Cada subsequência não agrupada (D_i) é inserida em um grupo escolhido aleatoriamente.

Passo 3 – Classificação dos indivíduos.

Os indivíduos são classificados conforme o procedimento a seguir, para posteriormente serem selecionados para cruzamento e/ou mutação.

- a) Calcule o valor de aptidão para cada indivíduo conforme a função usada;

- b) Ordene os indivíduos em ordem decrescente dos valores de aptidão e guarde os valores de melhor e pior aptidão;
- c) Calcule a média dos valores de aptidão.

4.5 Função de aptidão

Em um algoritmo genético, o papel da função de aptidão é avaliar a qualidade da solução. No algoritmo genético de agrupamento (AGA), o usuário pode selecionar uma das funções de aptidão: soma das similaridades médias, largura da silhueta média ou taxa do critério da variância. Essas funções são detalhadas a seguir.

4.5.1 Soma das similaridades médias

Esta função, desenvolvida no decorrer desta tese, é definida a partir da soma da similaridade entre subsequências pertencentes ao mesmo grupo. A média é feita através da divisão pelo número de combinações das subsequências, duas a duas. A similaridade é dada pela matriz de similaridade inicial (passo 3 do quadro 4.2).

$$\text{Função de Aptidão (FA)} = \sum_{C=1}^k \frac{1}{C(n,2)} \sum_{i=1}^{n-1} \sum_{j=i+1}^n \text{Sim}(D_i, D_j) \quad \text{onde,} \quad (4.2)$$

$\text{Sim}(D_i, D_j)$ = similaridade entre subsequências do grupo C;

n = número de subsequências do grupo C;

$C(n,2)$ = número de combinações das similaridades do grupo C duas a duas;

k = número de grupos.

O termo $\frac{1}{C(n,2)}$ na equação 4.2 é equivalente a $\frac{2}{n.(n-1)}$.

Para grupo com apenas uma subsequência, $\text{Sim}(D_i, D_j) = 0$. A princípio poderia se pensar que a similaridade entre uma subsequência e ela mesma teria o valor máximo, no caso igual a 1 (um). Entretanto, o termo similaridade é aplicado quando há no mínimo dois elementos, para que se possam ser comparados (EVERITT; LANDAU; LEESE, 2001). Portanto, grupos com apenas uma subsequência foram definidos com $\text{Sim}(D_i, D_j) = 0$.

Um aspecto que merece ser comentado é o processo de construção da função soma das similaridades médias. A similaridade entre duas subsequências varia no intervalo $[0,1]$. Quanto maior a similaridade entre duas subsequências, mais operações consecutivas comuns elas possuem, e como cada operação é feita em uma máquina, pode-se dizer que mais semelhantes são as máquinas de um mesmo grupo. Dessa forma, inicialmente pensou-se em usar apenas a soma das similaridades entre subsequências como função de aptidão. Entretanto, como o algoritmo genético trabalha com a maximização da função de aptidão, houve a formação de um agrupamento único, com todas as subsequências. Portanto, a soma das similaridades foi descartada como função de aptidão. Em seguida tentou-se usar a soma das similaridades médias, mas com a média feita pelo número de elementos ou subsequências de cada grupo. Esta função, apesar de aparentemente melhor que a anterior, é influenciada pelo tamanho de cada grupo. Grupos com muitas subsequências provocam a diminuição da média. Dessa forma, o algoritmo genético tenderia a formar vários grupos e com poucos elementos cada. Por este motivo, a média foi definida pelo número de combinações de subsequências. Assim, nenhum grupo é privilegiado por possuir maior número de elementos, ou seja, todos os grupos formados possuem a mesma base de comparação.

4.5.2 Largura da silhueta média

Esta função, proposta por Kaufman e Rousseeuw (1990), trabalha com dissimilaridades entre elementos de um conjunto de dados. A largura da silhueta média maximiza a homogeneidade interna de cada grupo e a heterogeneidade entre grupos distintos.

Aplicada ao projeto de arranjo físico modular, a homogeneidade dentro de cada módulo pode ser obtida pela minimização das dissimilaridades entre suas subsequências. A heterogeneidade entre diferentes módulos pode ser medida pela maximização da dissimilaridade média entre as subsequências.

Em princípio, o conceito da silhueta foi desenvolvido para encontrar o número correto de grupos, uma vez que diferentes agrupamentos tenham sido obtidos. Segundo Hruschka (2001), é possível usar o mesmo conceito para estabelecer uma função de aptidão adequada para um algoritmo genético de agrupamento. Algoritmos genéticos trabalham em paralelo e com certo número de soluções, então é possível trabalhar com cromossomos que representam diferentes agrupamentos, tanto em termos do número de grupos como em relação a estrutura dos agrupamentos. Portanto, é razoável supor que a largura da silhueta média seja adequada para otimizar os agrupamentos de forma mais dinâmica, isto é, determinando o número correto de grupos em problemas de agrupamento automático (HRUSCHKA, 2001).

Considere uma subsequência i qualquer do conjunto de dados. Para definir a silhueta $s(i)$, primeiro definimos A , o grupo ao qual a subsequência pertence e calculamos:

$a(i)$ = dissimilaridade média de i em relação a todas as outras subsequências de A .

Agora considere qualquer grupo C diferente de A e definimos:

$d(i, C)$ = dissimilaridade média da subsequência i em relação a todas as subsequências de C .

Depois de calcular $d(i, C)$ para todos os grupos com $C \neq A$, selecionamos a menor delas:

$$b(i) = \min d(i, C), \quad C \neq A \quad (4.3)$$

O grupo B para o qual esta dissimilaridade é mínima [$d(i, B) = b(i)$] é chamado de vizinho da subsequência i . É a segunda melhor escolha da subsequência i . Note que $b(i)$ depende da capacidade dos grupos em diferir de A, portanto as silhuetas não são definidas para $K = 1$ (quando houver apenas um grupo).

A largura da silhueta de cada subsequência i é definida como:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (4.4)$$

Propriedades de $s(i)$:

- a) $-1 \leq s(i) \leq 1$;
- b) $s(i) = 0$ quando $A = \{i_1\}$ (o grupo A possui apenas uma subsequência);
- c) $2 \leq K \leq N_{\text{Subseq}}$ (número de subsequências comuns).

Quando $s(i)$ é muito próxima do valor 1 significa que a dissimilaridade “interna”, $a(i)$, é muito menor que a menor dissimilaridade externa, $b(i)$. Portanto podemos concluir que a subsequência i está bem classificada em seu grupo. Quando $s(i)$ se aproxima de zero, $a(i)$ e $b(i)$ são aproximadamente iguais, portanto não está claro se a subsequência i deveria ficar em A ou em seu vizinho B. Já quando a silhueta $s(i)$ está próxima do valor -1 , então $a(i)$ é muito maior que $b(i)$, o que significa que a subsequência i está mal classificada no grupo A. Ela está muito mais próxima de B que de A.

A média de $s(i)$ para $i = 1, 2, \dots, n$, ou seja, para todas as subsequências, é chamada largura da silhueta média para o conjunto de dados e é a função de aptidão usada no algoritmo genético de agrupamento (AGA).

$$\text{Função de Aptidão (FA)} = \overline{s(i)} \quad (4.5)$$

Este valor, $\overline{s(i)}$ é usado para seleção do melhor valor de K (número de grupos), que acontece quando $\overline{s(i)}$ for máximo (KAUFMAN; ROUSSEEUW, 1990).

Kaufman e Rousseeuw (1990) propõem uma interpretação para a largura da silhueta média. Esta interpretação é sumarizada na tabela 4.1.

Tabela 4.1 – Interpretação da silhueta média $\overline{s(i)}$

$\overline{s(i)}$	Interpretação Proposta
0.71 – 1.00	Uma forte estrutura foi encontrada.
0.51 – 0.70	Uma razoável estrutura foi encontrada.
0.26 – 0.50	A estrutura é fraca e pode ser artificial; tente métodos adicionais para este conjunto de dados.
≤ 0.25	Não foi encontrada estrutura substancial.

Kaufman e Rousseeuw (1990)

4.5.3 Taxa do critério da variância

A função taxa do critério da variância (*Variance Ratio Criterion – VRC*), proposta por Calinski e Harabasz (1974), considera o isolamento externo de um subconjunto de dados e a sua homogeneidade interna. Esta medida está associada à determinação do número correto de agrupamentos em um conjunto de dados.

A taxa do critério de variância foi usada tanto por Cowgill, Harvey e Watson (1999) como função de aptidão de um algoritmo genético de agrupamento aplicado a um conjunto de dados gerados por simulação de Monte Carlo, como também por Casillas, González de Lena e Martínez (2003) para agrupamento de documentos através de algoritmos genéticos. A taxa do critério da variância é definida como:

$$VRC = \frac{\text{trace } B / (k - 1)}{\text{trace } W / (nr - k)} , \quad \text{onde} \quad (4.6)$$

n_T é o número total de objetos e k é o número de grupos. *Trace B*, ou soma de quadrados entre grupos, é a variabilidade existente entre os grupos. *Trace W*, ou soma de quadrados dos grupos, é a variabilidade dentro dos grupos.

Propriedades de VRC:

- a) $VRC \geq 0$;
- b) $2 \leq k \leq N_{\text{Subseq}}$ (número de subsequências comuns).

No presente trabalho, o agrupamento de subsequências é feito conforme o grau de similaridade entre as mesmas, portanto para cada cromossomo é calculada a função *VRC*, que irá medir a taxa da variância da similaridade entre as subsequências. O melhor cromossomo é aquele que possuir o maior valor de *VRC*.

O número total de objetos, n_T , é definido como o número total de combinações de similaridades entre subsequências que não se repetem. A variabilidade dentro dos grupos, *trace W*, é calculada por:

$$TraceW = \sum_{C=1}^k \sum_{i=1}^{n-1} \sum_{j=i+1}^n [Sim(D_i, D_j) - \bar{S}_C]^2 \quad \text{onde} \quad (4.7)$$

$Sim(D_i, D_j)$ = similaridade entre subsequências do grupo C;

n = número de subsequências do grupo C;

k = número de grupos;

$\bar{S}_C$ = similaridade média do grupo C, ou seja,

$$\bar{S}_C = \sum_{C=1}^k \frac{1}{C(n,2)} \sum_{i=1}^{n-1} \sum_{j=i+1}^n Sim(D_i, D_j) \quad \text{onde,} \quad (4.8)$$

$C(n,2)$ = número de combinações das similaridades do grupo C duas a duas.

O termo $\frac{1}{C(n,2)}$ na equação 4.8 é equivalente a $\frac{2}{n.(n-1)}$.

A variabilidade entre grupos, *trace B* é definida por:

$$Trace\ B = \sum_{C=1}^k C(n,2)(\bar{S}_C - \bar{S}_{..})^2 \quad \text{onde} \quad (4.9)$$

$\bar{S}_C$ = similaridade média do grupo C;

$\bar{S}_{..}$ = média de todas as combinações de similaridades (que não se repetem), e

$$\bar{S}_{..} = \frac{1}{n_T} \sum_{C=1}^k \sum_{i=1}^{n-1} \sum_{j=i+1}^n Sim(D_i, D_j) \quad \text{onde} \quad (4.10)$$

n_T = número total de combinações de similaridades que não se repetem.

A função *VRC* é análoga à estatística de teste F usada em ANOVA (análise de variância) para o caso univariado. Grandes valores de *VRC* indicam que *trace B* (variabilidade entre grupos) é muito maior que *trace W* (variabilidade dentro dos grupos), portanto o isolamento entre os grupos é grande.

4.6 Operadores dos cromossomos

Neste trabalho foram desenvolvidos três operadores de cruzamento e quatro de mutação. A aplicação dos operadores irá depender da codificação do cromossomo. Para a codificação R1, na qual o usuário determina o número desejado de módulos (número de grupos, K, conhecido) somente podem ser aplicados um dos operadores de cruzamento e um dos operadores de mutação 3. Para as codificações R2 ou R3, nas quais o número de módulos formados é um resultado do algoritmo (K desconhecido), podem ser aplicados um dos

operadores de cruzamento, o operador de mutação 1, de mutação 2 e um dos operadores de mutação 3.

Operadores de cruzamento e mutação clássicos, tais como, cruzamento de ponto único ou de dois pontos, mutação *bit-a-bit* ou aleatória, não são adequados para problemas de agrupamento, como tratado neste trabalho. Nos problemas de agrupamento os genes estão relacionados, portando os operadores devem agir em grupos de genes. Os operadores clássicos agem sobre os genes, e não sobre os grupos de genes, e se aplicados a problemas de agrupamento levam à insensibilidade ao contexto (COLE, 1998).

4.6.1 Operadores de cruzamento

No decorrer desta pesquisa foram desenvolvidos três operadores de cruzamento, denominados Recombinação, Recombinação com Correção Aleatória e Ponto Único de Grupo. Esses operadores são excludentes, ou seja, para executar o algoritmo genético de agrupamento somente um deles é usado. Os operadores são detalhados a seguir.

4.6.1.1 Recombinação

O operador de cruzamento denominado Recombinação foi adaptado de Hruschka (2001), que desenvolveu um algoritmo genético de agrupamento e o aplicou a várias bases de dados, entre elas *Iris Plants Database* (classes de plantas), *Australian Credit Appoval Database* (cartão de crédito), *Wiscosin Breast Cancer Database* (dados sobre câncer) e dados meteorológicos. De um modo simplificado, alguns grupos do pai A são copiados no pai B, e vice-versa, formando dois descendentes, C e D.

As principais contribuições referentes ao operador Recombinação para esta tese, que o fazem diferir do operador de Hruschka (2001) são:

- Hruschka (2001) trabalhou com centróide como medida de proximidade. No presente trabalho, foi usada como medida de proximidade a menor das dissimilaridades médias da subsequência i para com cada grupo contido no descendente, ou seja, $b(i) = \min d(i, G_j)$. O uso desta medida é detalhada adiante.
- A introdução do mecanismo de correção dos descendentes para evitar cromossomos inválidos. A seguir é apresentado um exemplo que ilustra a ação do operador.

Sejam dois cromossomos pais, A e B, com codificação R3, constituídos de 8 subsequências comuns:

A - 1 1 2 3 1 4 3 3 n° de grupos, $K_A = 4$

B - 1 2 3 1 1 4 4 3 n° de grupos, $K_B = 4$

O número de grupos não está representado no cromossomo.

Para o cromossomo A o grupo 1 é composto pelas subsequências S1, S2 e S5, o grupo 2 pela subsequência S3, o grupo 3 pelas subsequências S4, S7 e S8 e o grupo 4 pela subsequência S6.

Escolha aleatoriamente t ($1 \leq t \leq K_A$) grupos de A para copiar em C (descendente).

Por exemplo, selecione os grupos 1 e 2 de A (em negrito), ou seja, $t = \{1, 2\}$:

A - **1** **1** **2** 3 **1** 4 3 3

B - 1 2 3 1 1 4 4 3

Estes grupos modificam os grupos 1, 2 e 3 de B. O grupo 4 de B não é modificado.

A - **1** **1** **2** 3 **1** 4 3 3

B - **1** **2** **3** 1 **1** 4 4 3

Os genes de B sublinhados também serão modificados no descendente C. Copie os genes selecionados de A, ou seja, {1,2}, em C:

C - **1 1 2** 0 **1** 0 0 0

Os genes completados com zeros em C indicam posições com potencial para receber genes não alterados de B, ou representam genes que serão reposicionados no grupo mais próximo.

Por exemplo, o grupo 4 de B é copiado em C, pois não foi mudado. Os grupos 1 e 3 de B (sublinhados) serão reposicionados no grupo mais próximo. Assim:

C - **1 1 2** 0 **1** 4 4 0

Para reposicionamento dos genes zerados em C primeiro calcula-se a dissimilaridade média da subsequência i (gene zerado) para com cada grupo (G_j) contido no descendente C. O gene zerado será reposicionado no grupo com o qual possuir a menor das dissimilaridades médias, ou seja, $b(i) = \min d(i, G_j)$, com $1 \leq j \leq k$. k é o número de grupos do cromossomo.

Para gerar o descendente D é feito o processo inverso. Copie em D todos os genes de B que foram alterados. Assim:

D - 1 2 3 1 1 0 0 3

Os genes completados com zeros em D indicam posições com potencial para receber genes não alterados de A, ou representam genes que serão reposicionados no grupo mais próximo.

Por exemplo, o grupo 4 de A é copiado em D, pois não foi mudado. O grupo 3 de A não é copiado (foi mudado), então o gene zerado de D será reposicionado no grupo mais próximo.

Assim:

D - 1 2 3 1 1 4 0 3

Mecanismo de Correção dos Descendentes

Após o cruzamento, algumas correções devem ser feitas nos descendentes para evitar inconsistência. São elas:

1) Re-identificação dos grupos.

Ao criar os descendentes deve-se re-identificar os grupos, pois o grupo 1 do pai A é diferente do grupo 1 do pai B.

2) Verificação do número mínimo e máximo de grupos.

O número de grupos do descendente deve estar entre o número de grupos mínimo e máximo estabelecidos pelo usuário, ou seja, $K_{\min} \leq K \leq K_{\max}$. Duas situações podem ocorrer.

Situação1: Quando $K < K_{\min}$. Nesta situação é preciso criar grupos. O seguinte procedimento deve ser adotado:

- a) Selecione os grupos que possuem no mínimo duas subsequências;
- b) Para cada subsequência de cada grupo selecionado, calcule $a(i)$, onde $a(i)$ = dissimilaridade média de i em relação a todas as outras subsequências do grupo A (grupo a qual pertence);
- c) Ordene $a(i)$ de modo decrescente. A semente do novo grupo será a primeira subsequência da ordenação. Caso necessite de outro grupo, a segunda subsequência da ordenação será a semente, respeitando a condição a), e assim por diante.

Situação 2: Quando $K > K_{\max}$. Nesta situação é preciso extinguir grupos. O seguinte procedimento deve ser adotado:

- a) Para cada grupo calcule $a(i)$ = dissimilaridade média da subsequência i em relação a todas as outras subsequências do grupo A (grupo ao qual pertence) e calcule $\overline{a(i)}$. Selecione o grupo com maior $\overline{a(i)}$, ou seja, com a menor coesão interna;

- b) Distribua cada subsequência do grupo selecionado para o grupo com o qual possuir a menor dissimilaridade média: $b(i) = \min d(i, G_j)$, ou seja, a menor das dissimilaridades médias da subsequência i para com cada grupo G_j do cromossomo;
- c) Repita o procedimento se atingir o número de grupos máximo permitido ($K_{\max}$).

4.6.1.2 Recombinação com correção aleatória

Este operador de cruzamento gera os dois descendentes do mesmo modo que o operador Recombinação. A diferença está na correção do número mínimo e máximo de grupos permitido nos descendentes. O mecanismo de correção dos descendentes do operador Recombinação com Correção Aleatória é descrito a seguir.

Mecanismo de Correção dos Descendentes

Após o cruzamento, algumas correções devem ser feitas nos descendentes para evitar inconsistência. São elas:

1) Re-identificação dos grupos.

Ao criar os descendentes deve-se re-identificar os grupos, pois o grupo 1 do pai A é diferente do grupo 1 do pai B.

2) Verificação do número mínimo e máximo de grupos.

O número de grupos do descendente deve estar entre o número de grupos mínimo e máximo estabelecidos pelo usuário, ou seja, $K_{\min} \leq K \leq K_{\max}$. Duas situações podem ocorrer.

Situação 1: Quando $K < K_{\min}$. Nesta situação é preciso criar grupos. O seguinte procedimento deve ser adotado:

- a) Escolha aleatoriamente um grupo que possua no mínimo duas subseqüências.
- b) Selecione um ponto de partição aleatoriamente e divida o grupo, gerando dois novos grupos.
- c) Repita o procedimento até se atingir o número de grupos mínimo permitido ($K_{\min}$).

Situação 2: Quando $K > K_{\max}$. Nesta situação é preciso extinguir grupos. O seguinte procedimento deve ser adotado:

- a) Selecione um grupo aleatoriamente.
- b) Distribua cada subseqüência do grupo selecionado para o grupo com o qual possuir a menor dissimilaridade média: $b(i) = \min d(i, G_j)$, ou seja, a menor das dissimilaridades médias da subseqüência i para com cada grupo G_j do cromossomo;
- c) Repita o procedimento até se atingir o número de grupos máximo permitido ($K_{\max}$).

4.6.1.3 Ponto único de grupo

O operador de cruzamento denominado Ponto Único de Grupo age nos grupos de subseqüências comuns. Este operador foi inspirado no trabalho de Gonçalves Filho e Tiberti (2006). As principais contribuições referentes ao operador Ponto Único de Grupo para esta tese, que o fazem diferir do operador de Gonçalves Filho e Tiberti (2006) são:

- No operador Ponto Único de Grupo houve a preocupação em preservar nos descendentes o mesmo número de grupos dos cromossomos pais. Para tal foi definido um procedimento para verificação e preenchimento de grupos vazios.
- Em Gonçalves Filho e Tiberti (2006) os genes representam máquinas, enquanto no presente trabalho, subseqüências comuns.

- Neste trabalho as correções nos descendentes são realizadas na porção do cromossomo que foi trocada, enquanto em Gonçalves Filho e Tiberti (2006) na parte “fixa” do cromossomo.
- Em Gonçalves Filho e Tiberti (2006) em um grupo não pode haver menos que duas máquinas. Neste trabalho não há esta restrição quanto às subsequências.
- Introdução de critério para designação de elementos (subsequências) faltantes (passo 6 do exemplo a seguir).
- Na implementação deste operador não foi necessário usar as formas 2 e 3 de codificação do cromossomo propostas por Gonçalves Filho e Tiberti (2006). Os grupos e os elementos dos grupos são identificados automaticamente a partir da codificação original do cromossomo definida no item 4.3 do capítulo 4. Entretanto, no passo 1 do exemplo a seguir, a forma 2 de Gonçalves Filho e Tiberti (2006) foi usada para facilitar a visualização da ação do operador.

A seguir é apresentado um exemplo que ilustra a ação deste operador.

Sejam dois cromossomos, A e B, com 5 e 3 grupos respectivamente, ou seja, $K_A = 5$ e $K_B = 3$.

A - 1 2 3 1 3 4 5 4

B- 1 1 2 3 1 2 3 3

Para o cromossomo B, o grupo 1 é composto pelas subsequências S1, S2 e S5, o grupo 2 pela subsequência S3 e S6, e o grupo 3 pelas subsequências S4, S7 e S8.

Passo 1: Expresse a codificação original em termos de grupos. Os cromossomos A e B são re-escritos da forma:

A - 1 4 / 2 / 3 5 / 6 8 / 7

B - 1 2 5 / 3 6 / 4 7 8

O primeiro grupo do cromossomo A é formado pelas subsequências S1 e S4, o segundo grupo pela subsequência S2, e assim por diante. As barras representam a separação entre os grupos.

Passo 2: Gere um número aleatório inteiro entre 1 e $k-1$, onde k é o número de grupos do cromossomo pai com menos grupos. Este valor define o ponto do cruzamento.

No exemplo, como B possui o menor número de grupos, o ponto de cruzamento é escolhido aleatoriamente entre 1 e 2. Suponha que o valor 2 seja escolhido.

Passo 3: Os descendentes são gerados trocando os grupos com rótulo superior ao ponto de cruzamento. Portanto, os grupos trocados serão: grupos 3, 4 e 5 de A e grupo 3 de B.

Passo 4: Exclusão de subsequências repetidas. Esta troca de grupos pode violar a restrição de que cada subsequência esteja em apenas um grupo. Para corrigir este problema, as subsequências que foram trocadas e que já estejam presentes na parte “fixa” do cromossomo descendente são excluídas (subsequências sublinhadas).

C - 1 4 / 2 / 4 7 8

D - 1 2 5 / 3 6 / 3 5 / 6 8 / 7

Portanto, a subsequência 4 no descendente C e as subsequências 3, 5 e 6 no descendente D são excluídas. Assim:

C - 1 4 / 2 / 7 8

D - 1 2 5 / 3 6 / / 8 / 7

Passo 5: Preenchimento de grupos vazios. Após a exclusão das subsequências, caso haja grupo vazio no descendente, o que não é permitido, escolha aleatoriamente uma subsequência sem grupo (não presente no cromossomo) e a designe ao grupo vazio. Repita este procedimento até que os grupos vazios sejam preenchidos.

No descendente D, há um grupo vazio. Então, a subsequência 4 ainda sem grupo, será designada ao grupo vazio. Assim, os descendentes são re-escritos da forma:

C - 1 4 / 2 / 7 8

D - 1 2 5 / 3 6 / 4 / 8 / 7

Passo 6: Designação de subsequências faltantes. Se ainda houver subsequência sem grupo, insira-a no grupo com o qual possuir menor dissimilaridade média.

No descendente C, as subsequências 3, 5 e 6, ainda sem grupo serão designadas aos grupos conforme a menor dissimilaridade média. No descendente D não há mais subsequências a serem designadas.

Passo 7: Verificação de grupos vazios. Caso ainda haja grupos vazios (o número de subsequências sem grupo não foi suficiente), escolha aleatoriamente um grupo com mais de duas subsequências e retire aleatoriamente uma subsequência para ser colocada no novo grupo.

4.6.2 Operadores de mutação

Aplicada ao problema de agrupamento, a mutação corresponde a mover um objeto de um grupo para outro grupo. Como realizar a mutação depende da codificação do cromossomo (COLE, 1998).

Nesta tese foram desenvolvidos quatro operadores de mutação. Os operadores de mutação 1 e 2 foram inspirados no trabalho de Hruschka (2001) e os operadores de mutação 3 no de Cole (1998). As principais contribuições referentes aos operadores de mutação desenvolvidos que os fazem diferir dos operadores dos autores *supra* citados são:

- A forma de inserção dos objetos (subseqüências) nos grupos pelo operador de mutação 1. Hruschka (2001) os insere no grupo com centróide mais próximo. No presente trabalho as subseqüências são inseridas no grupo com a menor dissimilaridade média.
- A forma de escolha das sementes no operador de mutação 2. Em Hruschka (2001) as sementes são os objetos mais próximos e mais distantes do centróide. No presente trabalho as sementes são as subseqüências com menor e maior dissimilaridade média para com as subseqüências do mesmo grupo.
- O operador *Move* Aleatório é semelhante ao de Cole (1998), com a diferença de que neste trabalho, o grupo selecionado deve possuir no mínimo dois objetos (subseqüências).
- A criação do operador *Move* Direcionado.

Os operadores de mutação são descritos a seguir.

Mutação 1 (*MERGE*)

Opera em cromossomos que satisfazem a restrição $K > K_{\min}$. Este operador elimina aleatoriamente um grupo e insere suas subseqüências nos demais grupos. Cada subseqüência é

inserida no grupo com o qual possuir a menor dissimilaridade média: $b(i) = \min d(i, G_j)$, ou seja, a menor das dissimilaridades médias da subsequência i para com cada grupo G_j do cromossomo. A mutação 1 faz diminuir o número de grupos.

Mutação 2 (*BREAK*)

Opera em cromossomos que satisfazem a restrição $K < K_{\max}$. Este operador divide um grupo selecionado aleatoriamente (com no mínimo dois elementos) em dois novos grupos. A semente do primeiro subgrupo será a subsequência com menor $a(i)$, ou seja, menor dissimilaridade média da subsequência i em relação às outras subsequências do grupo selecionado. A semente do segundo subgrupo será a subsequência com maior $a(i)$. Feito isso, as subsequências restantes do grupo selecionado são deslocadas para os subgrupos mais próximos. O subgrupo mais próximo será aquele com menor dissimilaridade média: $b(i) = \min d(i, G_j)$, ou seja, a menor das dissimilaridades médias da subsequência i para com cada grupo G_j do cromossomo. Os demais grupos ficam intactos. A mutação 2 faz aumentar o número de grupos.

Mutação 3 (*MOVE*)

Dois tipos de operadores de mutação 3 foram desenvolvidos, *Move* Direcionado e *Move* Aleatório, entretanto, na execução do algoritmo genético somente um deles é usado. A mutação 3 preserva o número de grupos (K), mas muda o tamanho de dois grupos.

Move Direcionado

Selecione aleatoriamente uma subsequência de um grupo com no mínimo dois elementos. Mova esta subsequência para o grupo com o qual possuir a menor dissimilaridade média: $b(i) = \min d(i, G_j)$, ou seja, a menor das dissimilaridades médias da subsequência i para com cada grupo G_j do cromossomo.

Move Aleatório

Selecione aleatoriamente uma subsequência de um grupo com no mínimo dois elementos. Mova esta subsequência para qualquer outro grupo aleatoriamente.

4.7 Processo de seleção e geração da nova população

O modo de substituição da população adotado foi o geracional com elitismo. A cada geração, os N pais (N é o tamanho da população) são substituídos por $(N-m)$ filhos, conforme o procedimento descrito a seguir.

Passo 1 – Regra do Elitismo.

Para garantir que as boas soluções permaneçam de uma geração para outra e não sejam perdidas no processo de seleção, os m melhores cromossomos podem ser copiados integralmente para a nova geração. O usuário pode selecionar a porcentagem de cromossomos copiados, com $0 \leq m < N$, onde N é o tamanho da população. O restante da população é submetido ao processo de seleção pela Roleta para em seguida sofrerem cruzamento e/ou mutação.

Passo 2 - Inicia a Roleta para selecionar os indivíduos.

O processo detalhado de seleção pela Roleta é descrito a seguir.

a) Faça a normalização do valor de aptidão de cada indivíduo.

Segundo Lacerda, Carvalho e Ludermir (1999), os indivíduos podem apresentar valores de aptidão muito próximos. Quando isto ocorre, o processo de seleção pode se tornar praticamente aleatório, como se cada indivíduo possuísse uma estreita faixa na Roleta. Para evitar que isso ocorra, é feita uma normalização dos valores da aptidão. No presente trabalho

foi definido um fator para normalização (*Fat*), calculado pela diferença entre os valores de aptidão do melhor e do pior indivíduo, dividida pelo número de indivíduos da população:

$$Fat = \frac{(MelhorAptidão - PiorAptidão)}{N} \quad (4.11)$$

b) Calcule a probabilidade de seleção para cada indivíduo.

A probabilidade de seleção (P_i) de cada indivíduo equivale a uma determinada faixa da roleta, e é dada por:

$$P_i = (N - i) \cdot Fat + Fat, \quad \text{com } 1 \leq i \leq N, \quad (4.12)$$

i é a posição de cada indivíduo na população ordenada (do melhor para o pior). Isso fará com que os indivíduos com melhor aptidão tenham maior chance de seleção, pois possuirão maior faixa da roleta. Caso $Fat = 0$ (zero), assume-se $Fat = 1/N$, onde N é o tamanho da população. A constante Fat foi adicionada ao segundo termo da equação 4.12 para permitir que mesmo os indivíduos piores classificados tenham uma pequena chance de seleção.

c) Selecione os indivíduos aleatoriamente pela Roleta.

A roleta é montada na forma de um vetor (r) com $(N+1)$ elementos, onde N é o número de indivíduos da população. Cada elemento $r[i]$ corresponde a uma faixa (intervalo) da roleta equivalente à probabilidade de seleção do i -ésimo indivíduo da população (P_i), sendo o valor extremo do intervalo calculado pela fórmula:

$$r[i] = r[i+1] + P_i, \quad \text{para } 1 \leq i \leq N \quad (4.13)$$

ainda, $r[N+1] = 0$.

O procedimento adotado para seleção de um indivíduo é:

- Gera-se um número aleatório (x) entre 0 e $r[1]$ (comprimento da roleta), isto é $0 \leq x < r[1]$.
- O vetor r é varrido em busca de um índice i cujo valor x esteja entre $r[i]$ e $r[i+1]$, isto é, $r[i+1] \leq x < r[i]$, onde i é a posição do indivíduo selecionado.

Seja como exemplo o vetor r com $N = 5$. i representa a posição do indivíduo em ordem decrescente de aptidão, portanto $i=1$ representa o indivíduo com maior aptidão.

Primeiro calculamos a probabilidade de seleção de cada indivíduo pela fórmula: $P_i = (N-i)Fat + Fat$. Supondo $Fat = 1$, temos:

$$P(1) = (5-1) 1 + 1 = 5$$

$$P(2) = (5-2) 1 + 1 = 4$$

$$P(3) = (5-3) 1 + 1 = 3$$

$$P(4) = (5-4) 1 + 1 = 2$$

$$P(5) = (5-5) 1 + 1 = 1$$

Portanto, para cada i -ésimo indivíduo na população ordenada, temos uma probabilidade de ocorrência:

i:	1	2	3	4	5	6
P(i):	5	4	3	2	1	

Agora calculamos $r[i]$, ou seja, os extremos do intervalo (faixa da roleta) pela fórmula: $r[i] = r[i+1] + P_i$. Assim:

$$r[6] = 0, \text{ pois } r[N+1] = 0$$

$$r[5] = r[6] + P_5 = 0 + 1 = 1$$

$$r[4] = r[5] + P_4 = 1 + 2 = 3$$

$$r[3] = r[4] + P_3 = 3 + 3 = 6$$

$$r[2] = r[3] + P_2 = 6 + 4 = 10$$

$$r[1] = r[2] + P_1 = 10 + 5 = 15$$

i:	1	2	3	4	5	6
P(i):	5	4	3	2	1	
r[i]:	[15, 10)	[10, 6)	[6, 3)	[3, 1)	[1, 0)	[0]

$r[i]$ pode ser melhor ilustrado da forma:

Sorteamos um número aleatório entre 0 e $r[1]$, portanto $0 \leq x \leq 15$. Suponha que $x = 0,5$.

Agora, o vetor r é varrido em busca do índice i cujo valor x esteja entre $r[i]$ e $r[i+1]$, isto é, $r[i+1] \leq x < r[i]$, onde i corresponde à posição do indivíduo selecionado na população.

Portanto, o indivíduo selecionado para reprodução será $i = 5$.

Passo 3 - Gere os $(N-m)$ indivíduos restantes da nova população.

Os indivíduos restantes totalizam $(N-m)$, onde N é o número de indivíduos da população e m é o número de indivíduos copiados pela Regra do Elitismo. Os indivíduos restantes da nova população são gerados com base em um dos métodos descritos a seguir, selecionados pelo usuário. A descrição dos operadores de cruzamento e mutação é mostrada adiante.

Método Probabilístico

Este é o método tradicional (GOLDBERG, 1989) de aplicação dos operadores de cruzamento e mutação, conforme suas taxas de probabilidade. O método obedece as seguintes etapas:

- Selecione dois indivíduos (pais) aleatoriamente pela Roleta (item c) do passo 2);
- Aplique o operador de cruzamento para gerar dois descendentes, se a probabilidade de cruzamento ocorrer, caso contrário, os dois descendentes serão apenas cópias de seus pais;

- c) Aplique os operadores mutação 1, mutação 2 e mutação 3 em cada descendente, se a probabilidade de mutação ocorrer.

O operador é aplicado quando sua taxa de probabilidade for maior ou igual a um valor (x) gerado aleatoriamente entre 0 e 1 (limites das taxas de probabilidade).

Seja o exemplo, com as seguintes taxas de probabilidade definidas pelo usuário:

Pcruzamento = 0.8, Pmutação 1 = 0.05, Pmutação 2 = 0.05 e Pmutação 3 = 0.1.

Suponha que para o cruzamento, $x = 0.673$ (gerado aleatoriamente). Portanto, o cruzamento irá ocorrer, pois $0.8 > 0.673$.

Método Faixa Percentual

O método faixa percentual foi utilizado por Hruschka (2001). Os operadores de cruzamento e mutação são aplicados em um determinado número de indivíduos sorteados aleatoriamente, conforme as etapas:

- a) Sorteie ($N-m$) indivíduos aleatoriamente pela Roleta (item c) do passo 2);
- b) Nos primeiros $x\%$ (taxa de cruzamento) indivíduos sorteados, aplique o operador de cruzamento entre o 1º e o último indivíduo, seguido do 2º com o penúltimo, e assim por diante; caso $x\%$ seja um número ímpar, apenas copie o indivíduo central (sem par) para a nova população;
- c) Aplique os operadores de mutação nos indivíduos restantes:

Para codificação R1: Aplica-se o operador de mutação 3 em todos os indivíduos restantes. As mutações 1 e 2 não são aplicadas para essa codificação.

Para codificações R2 ou R3: Aplicam-se todas as mutações (1, 2 e 3), respeitando a ordem de seleção e as taxas de cada mutação.

Seja o exemplo com as seguintes taxas de probabilidade definidas pelo usuário:

$P_{\text{cruzamento}} = 0.5$, $P_{\text{mutação1}} = 0.25$; $P_{\text{mutação2}} = 0.25$; $P_{\text{mutação3}} = 0.0$.

Neste exemplo, a primeira metade dos cromossomos sorteados pela Roleta sofre cruzamento, pois $P_c = 0.5$. O primeiro cromossomo sorteado é cruzado com o $n/2$, em que n representa o número de cromossomos sorteados, o segundo cromossomo com $[(n/2)-1]$ e assim por diante. No exemplo, a mutação 1 é aplicada aos cromossomos que pertencem ao intervalo $[(n/2)+1, 3n/4]$ e a mutação 2 é aplicada aos cromossomos que pertencem ao intervalo $[(3n/4)+1, n]$. A mutação 3 não é aplicada, pois $P_{\text{mutação 3}} = 0$.

Independente do método usado, por probabilidade ou por faixa percentual, se for gerado menos que $(N-m)$ indivíduos, a população será completada com indivíduos gerados aleatoriamente pela Roleta (item c) do passo 2).

Passo 4 - Classificação dos indivíduos.

Classifique os indivíduos da nova população por ordem decrescente da aptidão e obtenha os valores médio, melhor e pior da aptidão.

4.8 Critério de parada

O usuário deve especificar o número de gerações desejado para término do processo de evolução e parada do algoritmo. Não há um número pré-definido para este valor, ele é determinado experimentalmente e deve ser suficiente para permitir a convergência da população. Para determinadas amostras de peças 200 gerações são suficientes, em outras são necessárias 500 ou mais. Após o término, o algoritmo ordena os indivíduos em ordem decrescente pelo valor da aptidão.

4.9 Considerações finais

Este capítulo apresentou o modelo do algoritmo genético de agrupamento (AGA) e os principais elementos do mesmo. A interface com o usuário e os detalhes da implementação são apresentados no capítulo 5.

5 IMPLEMENTAÇÃO DO ALGORITMO GENÉTICO DE AGRUPAMENTO

5.1 Considerações iniciais

Este capítulo trata dos detalhes da implementação do algoritmo genético de agrupamento (AGA) para obtenção dos módulos no projeto de arranjo físico modular. São apresentadas a interface com o usuário e os testes de verificação e validação do algoritmo. O algoritmo foi desenvolvido em Delphi.

Os testes de verificação são testes controlados cujo objetivo é verificar se algoritmo genético de agrupamento, AGA, faz o correto agrupamento das subsequências de operações. Para tanto foram realizados testes nos quais a resposta já era conhecida.

O objetivo dos testes de validação é realizar a comparação dos módulos básicos de arranjo físico gerados a partir do AGA com aqueles gerados na etapa 4 do procedimento baseado em análise de agrupamentos (*cluster analysis*) proposto por Huang (2003), descrito na seção 2.3.5.2.

5.2 Dados de entrada

O primeiro passo na execução do algoritmo é importar o arquivo (.txt) de dados contendo a sequência de operações de cada peça. Um exemplo do arquivo de entrada é apresentado no Apêndice A. Uma vez importada, a sequência de operações é mostrada como na figura 5.1. O conjunto de peças deste exemplo é o mesmo usado por Vakharia e Wemmerlöv apud Huang (2003).

Na parte superior da janela está o menu principal. A parte à esquerda refere-se à área de projeto. Nela são mostrados objetos importados que podem ser usados durante o projeto,

tais como matrizes de similaridade conhecidas, tabelas com seqüências de peças, e também uma lista de objetos gerados ao longo da execução do programa. Na parte à direita são mostradas as tabelas geradas como resultado da ação do usuário.

	ções (Seq)	Tempo de processamento (Tp) (minutos/
1	1, 4, 8, 9	96, 36, 36, 72
2	1, 4, 7, 4, 8, 7	36, 120, 20, 120, 24, 20
3	1, 2, 4, 7, 8, 9	96, 48, 36, 120, 36, 72
4	1, 4, 7, 9	96, 36, 120, 72
5	1, 6, 10, 7, 9	96, 72, 200, 120, 72
6	6, 10, 7, 8, 9	36, 120, 60, 24, 36
7	6, 4, 8, 9	72, 36, 48, 48
8	3, 5, 2, 6, 4, 8, 9	144, 120, 48, 72, 36, 48, 48
9	3, 5, 6, 4, 8, 9	144, 120, 72, 36, 48, 48
10	4, 7, 4, 8	120, 20, 120, 24
11	6	72
12	11, 7, 12	192, 150, 80
13	11, 12	192, 60
14	11, 7, 10	288, 180, 360
15	1, 7, 11, 10, 11, 12	15, 70, 54, 45, 54, 30
16	1, 7, 11, 10, 11, 12	15, 70, 54, 45, 54, 30
17	11, 7, 12	192, 150, 80
18	6, 7, 10	108, 180, 360
19	12	60

Figura 5.1 – Tela de entrada com seqüência de operações

A figura 5.1 mostra que a peça 1 é processada nas máquinas 1, 4, 8 e 9. O tempo de processamento nas respectivas máquinas é 96, 36, 36 e 72 minutos por lote. O tempo não é necessário para a execução do algoritmo genético, apenas foi colocado no arquivo de entrada pensando em futuras extensões do programa.

5.3 Geração das subsequências comuns únicas

Após importar os dados de entrada, o próximo passo é gerar as subsequências comuns únicas. Esta opção está disponível no menu “Algoritmos”. As subsequências comuns únicas são geradas de acordo com o algoritmo apresentado na figura 2.17 no capítulo 2. Para isso é apresentada uma tela para que o usuário confirme o tipo de dado a ser gerado, no caso, tabela de subsequências comuns únicas. Caso desejado, o usuário pode alterar o nome da tabela e escrever algum comentário. A figura 5.2 mostra a tela de confirmação.

Figura 5.2 – Confirmação para encontrar subsequências comuns únicas

Após selecionar Executar, são geradas as subsequências comuns únicas, mostradas na figura 5.3.

Figura 5.3 – Subseqüências comuns únicas

Neste exemplo, foram encontradas 17 subseqüências comuns únicas rotuladas de S1 a S17. A primeira subseqüência tem as operações executadas nas máquinas 1 e 4, a segunda, nas máquinas 1, 4 e 7, e assim por diante (figura 5.3).

5.4 Geração da matriz de similaridade

A partir das subseqüências comuns únicas, no menu “Algoritmos” torna-se disponível a opção “Gerar Matriz de Similaridade”. O usuário é solicitado a escolher um dos coeficientes de similaridade para que se possa ser gerada a matriz de similaridade. Os coeficientes disponíveis são: coeficiente baseado na Distância de Levenshtein, coeficiente LCS (mais

longa subsequência comum) e coeficiente *Merger*. Esses coeficientes estão detalhados na seção 2.3.5.1. A figura 5.4 mostra a tela para seleção do coeficiente de similaridade.

Figura 5.4 - Seleção do coeficiente de similaridade

Neste exemplo, o coeficiente escolhido foi o *Merger* (HUANG, 2003). Após a seleção do coeficiente, a matriz de similaridade gerada é apresentada como na figura 5.5.

Figura 5.5 – Matriz de similaridade

Como a matriz de similaridade é simétrica, optou-se por mostrar somente os valores posicionados à direita da diagonal principal. Neste exemplo, a similaridade entre as subsequências S1 e S2 é 0,94 (figura 5.5).

5.5 Definição dos parâmetros

Após gerar a matriz de similaridade, o usuário deve selecionar a opção “Algoritmo Genético” no menu “Algoritmos”, para definição dos parâmetros do AGA. A tela de parâmetros é mostrada na figura 5.6.

Figura 5.6 – Seleção dos parâmetros

O usuário deverá definir o tamanho da população, o número de gerações para término do algoritmo e a taxa de transferência dos indivíduos (Elitismo). Em relação a estes três parâmetros, não há restrições em relação ao número máximo permitido. Para saber quantos indivíduos serão transferidos para a nova população segundo a regra do Elitismo, basta multiplicar a taxa de transferência pelo tamanho da população. Na figura 5.6, como a taxa de transferência é 0,02 e o tamanho da população 50, apenas um indivíduo da população será transferido para a nova geração, no caso o melhor indivíduo. Também devem ser definidos o método de geração da nova população (se probabilístico ou por faixa percentual), e as taxas

de probabilidade ou faixa de aplicação dos operadores. Estas devem variar entre 0 (zero) e 1(um).

O usuário deverá escolher a forma de codificação dos cromossomos, se R1, R2 ou R3, e o número de grupos (K) mínimo e máximo permitido. Para evitar consumo excessivo de memória, o número de grupos máximo permitido é 255. Na prática significa uma fábrica com no máximo 255 módulos, quantia relativamente grande.

O usuário deverá selecionar uma das funções de aptidão (soma das similaridades médias, largura da silhueta média ou taxa do critério da variância) e especificar o passo para atualização das estatísticas. O método de seleção adotado foi o da Roleta, detalhado no capítulo 4. Os métodos torneio e aleatório foram apresentados na figura 5.6 objetivando futuras extensões do AGA.

5.6 Estatísticas geradas

Após mandar executar o algoritmo, o usuário pode acompanhar, na tela “Estatísticas”, o gráfico de convergência dos valores de aptidão. Este gráfico mostra a evolução dos valores de aptidão para o melhor e o pior indivíduo, e a evolução da aptidão média da população (figura 5.7).

Figura 5.7 – Estatísticas: valores de aptidão

Além do gráfico de valores da aptidão, a tela “Estatísticas” mostra o número de vezes em que os operadores foram aplicados, e o número e o percentual de sucesso de aplicação de cada operador. Por exemplo, a figura 5.7 mostra que foram aplicados 10275 cruzamentos. Este número equivale aproximadamente aos 85% de probabilidade da taxa de cruzamento, uma vez que o tamanho da população é de 50 indivíduos, o critério para término de 500 gerações e os cromossomos para cruzamento são tomados aos pares. Desse total (10275), 967 cruzamentos foram realizados, ou seja, resultaram em indivíduos diferentes dos pais. Outras três estatísticas são apresentadas:

- Número de indivíduos iguais ao melhor na população final: os grupos são formados pelas mesmas subsequências, e os grupos possuem os mesmos rótulos. Seja o exemplo:

Melhor indivíduo: (8, 8, 4, 4, 7, 7, 6, 3, 3, 1, 1, 5, 1, 5, 2, 5, 2)

Igual ao melhor: (8, 8, 4, 4, 7, 7, 6, 3, 3, 1, 1, 5, 1, 5, 2, 5, 2)

- Número de indivíduos similares ao melhor na população final: os grupos são formados pelas mesmas subsequências, mas algum rótulo de um indivíduo similar é diferente do rótulo do melhor indivíduo. Seja o exemplo:

Melhor indivíduo: (8, 8, 4, 4, 7, 7, 6, 3, 3, 1, 1, 5, 1, 5, 2, 5, 2)

Similar ao melhor: (8, 8, 5, 5, 7, 7, 6, 3, 3, 1, 1, 4, 1, 4, 2, 4, 2)

Os rótulos dos grupos 5 e 4 foram trocados, mas os grupos são formados pelas mesmas subsequências.

- Número máximo de indivíduos similares na população final: mostra a maior massa de indivíduos, ou seja, que possuem os mesmos agrupamentos.

Na tela “Estatísticas”, também é possível visualizar o gráfico “Posições Seleccionadas”. Este gráfico, apresentado na figura 5.8, mostra o número de vezes que os indivíduos foram seleccionados durante todo o processo de seleção. Por exemplo, a figura 5.8 mostra que o indivíduo da primeira posição (no caso o melhor cromossomo, pois o método de seleção é o da Roleta), foi seleccionado mais de 900 vezes. Já o indivíduo da 50ª posição foi o menos seleccionado.

Figura 5.8 – Estatísticas: posições selecionadas

5.7 Visualização das populações final e inicial

As populações final e inicial podem ser visualizadas nas respectivas janelas. A visualização é apresentada tanto na forma do cromossomo codificado, como em relação aos grupos formados, como ilustram as figuras 5.9 e 5.10.

Figura 5.9 – População final: grupos formados

A população final é ordenada de modo decrescente de aptidão, ou seja, o melhor indivíduo é mostrado na primeira posição. A figura 5.9 mostra que foram encontrados oito grupos ($K = 8$) e o valor da aptidão do melhor indivíduo foi 0,61101, para a largura da silhueta média. Os grupos foram formados pelas subseqüências: {S1, S2}, {S3, S4}, {S5, S6}, {S7}, {S8, S9}, {S10, S11, S13}, {S12, S14, S16} e {S15, S17}.

A figura 5.10 mostra a população final em termos da codificação do cromossomo.

Figura 5.10 – População final: cromossomo codificado

A figura 5.11 apresenta os grupos gerados na população inicial.

	K	Aptidão	Grupos
1	17	0.00000	[[S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
2	17	0.00000	[[S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
3	17	0.00000	[[S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
4	16	-0.05556	[[S1], [S2], [S3], [S4], [S5, S7], [S6], [S8], [S9], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
5	10	-0.07788	[[S1], [S2], [S3, S15, S16], [S4, S9, S12], [S5, S6], [S7, S13], [S8], [S10], [S11, S17], [S14]]
6	2	-0.08499	[[S1, S2, S6, S9, S11, S12, S14, S17], [S3, S4, S5, S7, S8, S10, S13, S15, S16]]
7	2	-0.08655	[[S1, S3, S4, S7, S9, S11, S12, S13, S14, S15, S16], [S2, S5, S6, S8, S10, S17]]
8	2	-0.08965	[[S1, S3, S6, S7, S8, S10, S11, S12, S16], [S2, S4, S5, S9, S13, S14, S15, S17]]
9	16	-0.09804	[[S1], [S2, S9], [S3], [S4], [S5], [S6], [S7], [S8], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
10	16	-0.10381	[[S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9, S11], [S10], [S12], [S13], [S14], [S15], [S16], [S17]]
11	16	-0.10458	[[S1, S10], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
12	16	-0.10824	[[S1], [S2], [S3, S15], [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S11], [S12], [S13], [S14], [S16], [S17]]
13	16	-0.11111	[[S1], [S2], [S3], [S4, S8], [S5], [S6], [S7], [S9], [S10], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
14	14	-0.11953	[[S1, S3], [S2], [S4], [S5], [S6], [S7, S10], [S8, S9], [S11], [S12], [S13], [S14], [S15], [S16], [S17]]
15	5	-0.14794	[[S1, S3, S4], [S2, S6, S10, S11], [S5, S12, S16], [S7, S8, S9, S13, S15], [S14, S17]]
16	6	-0.14951	[[S1, S2, S3, S11, S12], [S4], [S5, S16, S17], [S6, S7, S15], [S8, S9, S14], [S10, S13]]
17	15	-0.16503	[[S1], [S2], [S3], [S4, S16], [S5], [S6, S12], [S7], [S8], [S9], [S10], [S11], [S13], [S14], [S15], [S17]]

Figura 5.11 – População inicial

Após o término da execução do algoritmo, o usuário pode salvar os parâmetros do algoritmo, a convergência dos valores de aptidão, as estatísticas, e as populações inicial e final. Esses resultados são salvos em um texto (.txt). Um exemplo de arquivo de saída é apresentado no Apêndice B.

5.8 Verificação do algoritmo genético de agrupamento

A verificação do algoritmo genético implementado ocorreu segundo a execução de testes controlados. Nestes testes a resposta é conhecida. O objetivo é verificar se algoritmo

genético de agrupamento, AGA, faz o correto agrupamento das subsequências de operações. Foram realizados quatro tipos de testes com aumento gradual de dificuldade. Os testes são apresentados a seguir.

Teste 1 – Conjuntos de Sequências Iguais.

Este teste possui dois objetivos. O primeiro é verificar se o algoritmo gera grupos totalmente disjuntos na população inicial (uma subsequência em cada grupo), visto que não há nenhum par de subsequências com operações em comum. O segundo é verificar se o valor da aptidão permanece igual a zero, independente da função de aptidão usada, pois a similaridade entre duas quaisquer subsequências é zero. Neste teste, a tabela de sequência de operações é formada por 15 peças, sendo iguais 3 a 3. Exemplo: {P1, P2 e P3}, {P4, P5 e P6} e assim por diante, como mostra a tabela 5.1.

Tabela 5.1 – Sequência de operações: teste 1

Peça (P)	Sequência de operações (Seq)
1	1, 2, 3, 4
2	1, 2, 3, 4
3	1, 2, 3, 4
4	5, 6, 7, 8
5	5, 6, 7, 8
6	5, 6, 7, 8
7	9, 10, 11, 12
8	9, 10, 11, 12
9	9, 10, 11, 12
10	13, 14, 15, 16
11	13, 14, 15, 16
12	13, 14, 15, 16
13	17, 18, 19, 20
14	17, 18, 19, 20
15	17, 18, 19, 20

O algoritmo gerou cinco subsequências comuns únicas (figura 5.12), conforme esperado, visto que existem cinco conjuntos de peças com seqüências totalmente diferentes.

No.	Subseqüências comuns únicas
S1	1, 2, 3, 4
S2	13, 14, 15, 16
S3	17, 18, 19, 20
S4	5, 6, 7, 8
S5	9, 10, 11, 12

Figura 5.12 – Teste 1: subsequências comuns únicas

A matriz de similaridade gerada de acordo com o coeficiente *Merger* é mostrada na figura 5.13.

	S1	S2	S3	S4	S5
S1	1,00	0,00	0,00	0,00	0,00
S2		1,00	0,00	0,00	0,00
S3			1,00	0,00	0,00
S4				1,00	0,00
S5					1,00

Figura 5.13 – Teste 1: matriz de similaridade

Excluindo a diagonal principal, a similaridade entre as subsequências é zero, ou seja, não há nenhum par de operações consecutivas comuns entre duas quaisquer subsequências.

Os parâmetros do AG são apresentados na tabela 5.2.

Tabela 5.2 – Parâmetros: teste 1

Parâmetros do AG
Tamanho da população: 50
Número de gerações: 500
Taxa de transferência: 0,02
Método de geração: probabilístico
Taxa de cruzamento: 0,85
Taxa de mutação 1, 2 e 3: 0,05
Codificação do cromossomo: R2
No. de grupos entre: 2 e 5
Função de Aptidão: largura da silhueta média

As estatísticas geradas e a população final são mostradas nas figuras 5.14 e 5.15 respectivamente.

Figura 5.14 – Estatísticas: teste 1

O gráfico mostra que os valores da melhor aptidão, da pior e da média das aptidões permaneceram em 0 (zero) ao longo da execução do algoritmo. Isso confirma o esperado, pois para a função largura da silhueta média, $\overline{s(i)}$, o valor 0 (zero) indica que as dissimilaridades “interna”, $a(i)$, e “externa”, $b(i)$, são aproximadamente iguais, portanto não está claro se a subsequência i deveria ficar no grupo A ou em seu vizinho B. Também para as funções soma das similaridades médias e taxa do critério da variância o gráfico apresentou o mesmo aspecto da figura 5.14, com valores da melhor, pior e da média das aptidões zerados. Para a função soma das similaridades médias isso é justificado pelo fato da similaridade ser zero entre

quaisquer duas subseqüências, e para a função taxa do critério da variância é justificado pela ausência de variação dentro dos grupos e entre os grupos.

A figura 5.15 mostra que todos os indivíduos da população inicial possuem cinco grupos ($K = 5$). Isto era esperado, pois se usou a representação R2, na qual são feitos agrupamentos iniciais. Este exemplo refere-se ao caso especial descrito no passo 2 da seção 4.4 (geração da população inicial): se a subseqüência D_i possuir similaridade igual a zero para com todas as outras subseqüências e se $K < K_{\max}$, é criado um novo grupo para D_i . Caso Sim (D_i) = 0 e $K = K_{\max}$, ou seja, igual a 5, não há mais subseqüências para serem alocadas, portanto cada subseqüência ficará em um grupo. Este caso especial é aplicado para as codificações R1 e R2.

Figura 5.15 – População inicial: teste 1, codificação R2

A figura 5.16 ilustra os indivíduos da população final. Os valores da aptidão permaneceram zerados, e o número de grupos, K, variou entre 2 e 3. Isso ocorreu principalmente devido a atuação dos operadores de mutação, na tentativa de mudar as subsequências de grupos e com isso melhorar o valor da aptidão. Neste caso, $K = 2$, (figura 5.16) não significa o número de grupos ótimo, pois qualquer valor de K poderia corresponder ao melhor indivíduo, visto que a aptidão é zero para todos os indivíduos.

	K	Aptidão	Grupos
1	2	0,00000	[[S1], [S2, S3, S4, S5]]
2	2	0,00000	[[S1], [S2, S3, S4, S5]]
3	2	0,00000	[[S1], [S2, S3, S4, S5]]
4	2	0,00000	[[S1], [S2, S3, S4, S5]]
5	2	0,00000	[[S1], [S2, S3, S4, S5]]
6	2	0,00000	[[S1], [S2, S3, S4, S5]]
7	2	0,00000	[[S1], [S2, S3, S4, S5]]
8	2	0,00000	[[S1, S2, S4, S5], [S3]]
9	2	0,00000	[[S1], [S2, S3, S4, S5]]
10	3	0,00000	[[S1], [S2, S4, S5], [S3]]
11	2	0,00000	[[S1], [S2, S3, S4, S5]]
12	2	0,00000	[[S1], [S2, S3, S4, S5]]

Figura 5.16 – População final: teste 1

Para os dados do teste 1, a população final sempre apresentará mesmo valor de aptidão (igual a zero), independente dos valores dos parâmetros do AG.

O número de grupos (K) da população inicial poderá variar em virtude da codificação do cromossomo usada. Para codificação R3, na qual os grupos são gerados aleatoriamente, conforme passo 2 da seção 4.4, uma possível população é mostrada na figura 5.17.

	K	Aptidão	Grupos
1	3	0,00000	[[S1, S2], [S3, S4], [S5]]
2	3	0,00000	[[S1], [S2, S5], [S3, S4]]
3	2	0,00000	[[S1], [S2, S3, S4, S5]]
4	5	0,00000	[[S1], [S2], [S3], [S4], [S5]]
5	5	0,00000	[[S1], [S2], [S3], [S4], [S5]]
6	4	0,00000	[[S1, S4], [S2], [S3], [S5]]
7	3	0,00000	[[S1, S4], [S2, S5], [S3]]
8	4	0,00000	[[S1], [S2], [S3, S4], [S5]]
9	2	0,00000	[[S1, S5], [S2, S3, S4]]
10	2	0,00000	[[S1, S3, S5], [S2, S4]]
11	5	0,00000	[[S1], [S2], [S3], [S4], [S5]]
12	4	0,00000	[[S1, S4], [S2], [S3], [S5]]
13	4	0,00000	[[S1], [S2, S4], [S3], [S5]]
14	3	0,00000	[[S1, S2, S5], [S3], [S4]]
15	2	0,00000	[[S1], [S2, S3, S4, S5]]
16	4	0,00000	[[S1], [S2, S3], [S4], [S5]]

Figura 5.17 – População inicial: teste 1, codificação R3

A figura 5.17 mostra que a alocação das subsequências aos grupos foi feita totalmente aleatória, conforme codificação R3.

Teste 2 – Sequências Idênticas.

O objetivo deste teste é verificar se o algoritmo gera apenas um agrupamento, mesmo desrespeitando as condições de número de grupos mínimo ($K_{\min}$) e máximo ($K_{\max}$).

Neste teste, as 15 peças possuem sequência de operações idênticas, ou seja, todas as peças são processadas nas mesmas máquinas e na mesma ordem. As peças são processadas nas máquinas {1, 2, 3, 4}.

Os parâmetros do AG são os mesmos da tabela 5.2, exceto pelo número de grupos (K), que forçosamente foi especificado entre 1 e 17. Este intervalo de K não respeitou a condição $2 \leq K \leq N_{\text{Subseq}}$ para a largura da silhueta média, onde N_{Subseq} é o número de subsequências comuns. A figura 5.18 mostra a população final obtida.

Figura 5.18 – População final: teste 2, largura da silhueta média

As figuras 5.19 e 5.20 mostram respectivamente para as funções soma das similaridades médias e taxa do critério da variância, a tela de parâmetros com a condição de K forçada, no intervalo $4 \leq K \leq 17$.

The screenshot shows the 'Algoritmo Genético' window with the 'Parâmetros' tab selected. The 'Matriz de Similaridade' field contains 'SeqOper_Test2 - Subseqüências de Operações - Matriz de Similaridade'. Under 'População de Indivíduos', the 'Tamanho da população' is 50, 'Número de gerações' is 500, 'Taxa de transferência' is 0,02, and 'Método de geração' is 'Probabilístico'. The 'Taxa de cruzamento' is 0,850, and the three mutation rates (Merge, Break, Move) are all 0,050. Under 'Cromossomos', 'Codificação' is 'R2' and 'Nro. de grupos entre' is '4 a 17'. The 'Função de aptidão' is 'Soma da similaridade média'. Under 'Método de seleção', 'Roleta' is selected. The status bar at the bottom shows '500/500' and '00:00:00'.

Figura 5.19 - Parâmetros para teste 2: $4 \leq K \leq 17$, soma das similaridades médias

The screenshot shows the same 'Algoritmo Genético' window, but the 'Função de aptidão' is now 'Taxa do critério da variância'. All other parameters remain the same as in Figure 5.19. The status bar at the bottom shows 'Execução terminada!', '500/500', and '00:00:00'.

Figura 5.20 - Parâmetros para teste 2: $4 \leq K \leq 17$, taxa do critério da variância

O algoritmo sempre formará apenas um agrupamento, para todas as funções de aptidão, como mostra a figura 5.21.

	K	Aptidão	Cromossomo
1	1	0,00000	[1]
2	1	0,00000	[1]
3	1	0,00000	[1]
4	1	0,00000	[1]
5	1	0,00000	[1]
6	1	0,00000	[1]
7	1	0,00000	[1]
8	1	0,00000	[1]
9	1	0,00000	[1]
10	1	0,00000	[1]
11	1	0,00000	[1]
12	1	0,00000	[1]

Execução terminada! 500/500 00:00:00

Figura 5.21 - População final para teste 2: $4 \leq K \leq 17$

Teste 3 – Sequências com algumas máquinas em comum.

O objetivo deste teste é verificar se o algoritmo forma corretamente os agrupamentos quando a solução ótima é conhecida. A tabela 5.3 mostra as sequências de operações usadas para este teste.

Tabela 5.3 - Sequência de operações: teste 3

Peça (P)	Sequência de operações (Seq)
1	1, 2, 3, 4, 50, 60
2	1, 2, 3, 4, 51, 61
3	1, 2, 5, 6, 52, 62
4	1, 2, 5, 6, 53, 63
5	9, 10, 12, 13, 56, 66
6	9, 10, 12, 13, 57, 67
7	11, 12, 13, 9, 58, 68
8	11, 12, 13, 9, 59, 69
9	10, 12, 13, 70, 80
10	10, 12, 13, 71, 81
11	14, 15, 6, 72, 82
12	14, 15, 6, 73, 83
13	14, 15, 16, 17, 74, 84
14	14, 15, 16, 17, 75, 85
15	15, 16, 17, 76, 86
16	15, 16, 17, 77, 87
17	18, 19, 78, 88
18	18, 19, 79, 89

As subsequências comuns únicas geradas pela aplicação do algoritmo apresentado na figura 2.17 no capítulo 2, são apresentadas na tabela 5.4.

Tabela 5.4 - Subsequências comuns únicas: teste3

No.	Subsequências comuns únicas
S1	1, 2
S2	1, 2, 3, 4
S3	1, 2, 5, 6
S4	10, 12, 13
S5	11, 12, 13, 9
S6	12, 13
S7	14, 15
S8	14, 15, 16, 17
S9	14, 15, 6
S10	15, 16, 17
S11	18, 19
S12	9, 10, 12, 13

A figura 5.22 apresenta a matriz de similaridade gerada pela aplicação do coeficiente *Merger* (HUANG, 2003) nas subsequências da tabela 5.4.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
S1	1,00	0,94	0,94	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
S2		1,00	0,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
S3			1,00	0,00	0,00	0,00	0,00	0,00	0,31	0,00	0,00	0,00
S4				1,00	0,65	0,94	0,00	0,00	0,00	0,00	0,00	0,98
S5					1,00	0,94	0,00	0,00	0,00	0,00	0,00	0,50
S6						1,00	0,00	0,00	0,00	0,00	0,00	0,94
S7							1,00	0,94	0,94	0,44	0,00	0,00
S8								1,00	0,65	0,98	0,00	0,00
S9									1,00	0,33	0,00	0,00
S10										1,00	0,00	0,00
S11											1,00	0,00
S12												1,00

Figura 5.22 – Matriz de similaridade: teste 3

Os parâmetros do AG são os mesmos da tabela 5.2, exceto pelo número de grupos que variou entre 2 e 12 (número de subsequências comuns). Este teste foi realizado com as três funções de aptidão.

Largura da Silhueta Média

A figura 5.23 mostra os agrupamentos formados na população final para a função de aptidão largura da silhueta média.

	K	Aptidão	Grupos
1	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
2	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
3	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
4	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
5	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
6	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
7	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
8	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
9	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
10	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
11	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
12	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]

Figura 5.23 - População final: teste 3, largura da silhueta média

O melhor indivíduo da população final apresentou cinco grupos ($K=5$), com aptidão igual a 0,76638. Para este, os agrupamentos formados foram: [S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10] e [S11]. As seguintes análises podem ser feitas:

- O grupo formado pelas subsequências S1, S2 e S3 possui as máquinas $1 \rightarrow 2$ em comum (tabela 5.4), e alta similaridade, como mostra a figura 5.22. Portanto, era de se esperar que as subsequências ficassem no mesmo grupo.
- O grupo formado pelas subsequências S4, S5, S6 e S12 possui as máquinas $12 \rightarrow 13$ em comum. Além disso, S4 está totalmente contida em S12, com as três operações em comum, como mostra a tabela 5.4.
- A subsequência S11 permaneceu isolada em um grupo, pois como ilustra a figura 5.22, a similaridade dela para com todas as outras subsequências é zero.
- A subsequência S7 possui similaridade igual a 0,94 para com S9, portanto permaneceram juntas em um grupo. O mesmo ocorreu com S8 e S10 que possuem similaridade igual a

0,98 (figura 5.22). Poderia se pensar que S7, S8, S9 e S10 deveriam permanecer em um único grupo, pois possuem a máquina 15 em comum. Entretanto, a similaridade entre S7 e S10 e entre S9 e S10 é muito baixa, respectivamente 0,44 e 0,33 (figura 5.22). Devido a este motivo e aliado à função de aptidão, as subsequências foram separadas em dois grupos. A figura 5.24 ilustra um cromossomo da população final que as subsequências S7, S8, S9 e S10 permaneceram unidas. Entretanto, este indivíduo (45ª posição) apresenta aptidão igual a 0,70519, menor que a aptidão ótima.

The screenshot shows the 'Algoritmo Genético' window with the 'População Final' tab selected. The 'Grupos' sub-tab is active, displaying a table of chromosomes. The 45th chromosome is highlighted in blue.

	K	Aptidão	Grupos
39	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
40	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
41	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
42	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
43	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
44	5	0,76638	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
45	4	0,70519	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S8, S9, S10], [S11]]
46	4	0,69671	[[S1, S2, S3, S11], [S4, S5, S6, S12], [S7, S9], [S8, S10]]
47	4	0,69671	[[S1, S2, S3, S11], [S4, S5, S6, S12], [S7, S9], [S8, S10]]
48	6	0,64572	[[S1, S3], [S2], [S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]
49	6	0,54779	[[S1, S2, S3], [S4, S5, S6, S12], [S7, S9], [S8], [S10], [S11]]
50	4	0,48969	[[S1, S2, S3, S4, S5, S6, S12], [S7, S9], [S8, S10], [S11]]

At the bottom of the window, there are buttons for 'Executar' and 'Salvar', and a status bar showing '500/500' and '00:00:00'.

Figura 5.24 - Cromossomo na 45ª posição com S7, S8, S9 e S10 agrupadas

Este teste mostrou que o algoritmo genético agrupou corretamente as subsequências para a função de aptidão largura da silhueta média.

Soma das Similaridades Médias

Para a função soma das similaridades médias, o melhor cromossomo, ora apresentava 5 grupos, ora 6 grupos, para o mesmo valor da aptidão, 2,58796, como ilustram as figuras 5.25, 5.26 e 5.27.

	K	Aptidão	Grupos
1	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
2	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
3	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
4	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
5	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
6	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
7	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
8	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
9	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
10	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
11	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]
12	6	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10], [S11]]

Figura 5.25 – População final: teste3, soma das similaridades médias, S5, S10 e S11 isoladas

A figura 5.25 mostra que o algoritmo formou seis grupos (K=6). A subsequência S11 permaneceu isolada, como deveria ser, já que sua similaridade para com as demais subsequências é zero (figura 5.22). Entretanto, as subsequências S5 e S10 também permaneceram isoladas, apesar de possuírem alta similaridade com S6 (0,94) e S8 (0,98) respectivamente (tabela 5.4).

	K	Aptidão	Grupos
1	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
2	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
3	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
4	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
5	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
6	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
7	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
8	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
9	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
10	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
11	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]
12	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5, S11], [S7, S8, S9], [S10]]

Execução terminada! 500/500 00:00:00

Figura 5.26 - População final: teste3, soma das similaridades médias, S5 e S11 agrupadas

A figura 5.26 mostra que o algoritmo formou cinco grupos (K=5). Para o melhor cromossomo, as subsequências S5 e S11 permaneceram no mesmo grupo, apesar da similaridade entre elas ser zero. A subsequência S10 permaneceu isolada, apesar da similaridade igual a 0,98 com S8.

	K	Aptidão	Grupos
1	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
2	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
3	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
4	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
5	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
6	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
7	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
8	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
9	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
10	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
11	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]
12	5	2,58796	[[S1, S2, S3], [S4, S6, S12], [S5], [S7, S8, S9], [S10, S11]]

Execução terminada! 500/500 00:00:00

Figura 5.27 – População final: teste3, soma das similaridades médias, S10 e S11 agrupadas

A figura 5.27 apresenta um resultado onde ocorreu o inverso do anterior. A subsequência S5 permaneceu isolada em um grupo, apesar da similaridade 0,94 com S6. A subsequência S10 foi agrupada com S11 apesar da similaridade zero entre elas, e 0,98 com S8.

Esta alternância de S11, ora com S5, ora com S10 e ora isolada não afetou o valor da função de aptidão, visto que nos três casos [S11], [S5, S11] e [S10, S11], a similaridade média de cada agrupamento é zero. Além disso, a aptidão não foi alterada, pois os demais grupos formados foram os mesmos, [S1, S2, S3], [S4, S6, S12] e [S7, S8, S9], como mostram as figuras 5.25 a 5.27.

Assim, este teste mostrou que com a função soma das similaridades médias, subsequências totalmente dissimilares foram colocadas no mesmo grupo. Além disso, diferentes agrupamentos foram formados para o mesmo valor de aptidão. Isso ocorreu porque

a função soma permite agrupar elementos de diversas maneiras de forma que a soma seja a máxima.

Dessa forma, este teste demonstrou que a heterogeneidade entre diferentes grupos é uma característica importante para projeto de arranjo físico modular, e a função soma das similaridades médias não contempla esta característica.

Portanto, a função soma das similaridades médias não é indicada para geração de grupos ou módulos em projeto de arranjo físico modular.

Taxa do Critério da Variância

Para a função taxa do critério da variância, não houve constância do número de grupos do melhor cromossomo, nem dos agrupamentos formados. Ora o melhor cromossomo apresentava 5 grupos, ora 2 grupos, ora 4. O cromossomo com maior aptidão é mostrado na figura 5.28, com $K=5$ e aptidão igual a 3749,23.

	K	Aptidão	Grupos
1	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
2	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
3	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
4	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
5	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
6	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
7	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
8	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
9	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
10	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
11	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]
12	5	3749,23021	[[S1, S3], [S2, S5, S7, S11], [S4, S6, S12], [S8, S10], [S9]]

Figura 5.28 – Cromossomo com maior aptidão: teste 3, taxa do critério da variância

Entretanto, a análise dos agrupamentos formados mostra que o algoritmo colocou no mesmo grupo subsequências cuja similaridade é zero (vide figura 5.22), como é o caso do segundo grupo da figura 5.28, formado por S2, S5, S7 e S11.

A figura 5.29 apresenta outro resultado gerado para a taxa do critério da variância, com aptidão máxima igual a 3749,23.

	K	Aptidão	Grupos
1	5	3749,23021	[[S1, S3], [S2, S5, S9, S11], [S4, S6, S12], [S7], [S8, S10]]
2	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
3	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
4	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
5	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
6	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
7	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
8	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
9	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
10	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
11	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]
12	2	14,05348	[[S1, S2, S3, S5, S7, S8, S9, S10, S11], [S4, S6, S12]]

Figura 5.29 – Cromossomo com maior aptidão e agrupamentos diferentes: teste 3, taxa do critério da variância.

Na figura 5.29, apesar do número de grupos e da aptidão do melhor cromossomo possuírem o mesmo valor que na figura 5.28, os agrupamentos formados são diferentes. Os grupos [S2, S5, S7, S11] e [S9] da figura 5.28 foram transformados em [S2, S5, S9, S11] e [S7] na figura 5.29.

Este teste mostra que a função taxa do critério da variância tende a colocar no mesmo grupo subsequências cujo valor da similaridade sejam próximos, por exemplo, as subsequências do grupo [S2, S5, S9, S11] possuem todas similaridades iguais a zero (figura

5.22). As subsequências do grupo [S4, S6, S12] também possuem similaridades próximas, 0,94 e 0,98 (figura 5.22). Isso é explicado pelo fato da função taxa do critério da variância (*VRC*) tender a minimizar a variação existente dentro dos grupos, representada por *trace W* na equação 4.7 no capítulo 4, visto que o objetivo é a maximização da função *VRC*. Assim, pode-se concluir que a função taxa do critério da variância não é indicada quando se trabalha com a variável similaridade, e, portanto, não é indicada para geração de grupos ou módulos em projeto de arranjo físico modular.

Teste 4 – Confirmação do número de grupos.

Este teste é uma evolução do teste 3. Às 12 subsequências comuns únicas mostradas na tabela 5.4, foram acrescentadas 8 subsequências. Por sua semelhança às 12 subsequências existentes, as novas subsequências deverão ser designadas aos grupos já formados no teste 3 (figura 5.23). O objetivo deste é verificar se o algoritmo genético continua a formar os mesmos grupos quando se sabe *a priori* em quais grupos as novas subsequências devem entrar. A tabela 5.5 mostra as subsequências comuns únicas do teste 4.

Tabela 5.5 - Subseqüências comuns únicas: teste 4

No.	Subseqüências comuns únicas
S1	1, 2
S2	1, 2, 3, 4
S3	1, 2, 5, 6
S4	10, 12, 13
S5	11, 12, 13, 9
S6	12, 13
S7	14, 15
S8	14, 15, 16, 17
S9	14, 15, 6
S10	15, 16, 17
S11	18, 19
S12	9, 10, 12, 13
S13	1, 2, 3
S14	3, 4, 5
S15	10, 13
S16	12, 9
S17	11, 13
S18	15, 6
S19	15, 16
S20	16, 17

Na tabela 5.5, as subseqüências S13 a S20 representam as 8 novas subseqüências.

A figura 5.30 apresenta a matriz de similaridade gerada pela aplicação do coeficiente *Merger* (HUANG, 2003) nas subseqüências da tabela 5.5.

Subseqüências Teste 4 - Matriz de Similaridade																				
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
S1	1,00	0,94	0,94	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,94	0,00	0,00	0,00	0,00	0,00	0,00	0,00
S2		1,00	0,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,98	0,65	0,00	0,00	0,00	0,00	0,00	0,00
S3			1,00	0,00	0,00	0,00	0,00	0,00	0,31	0,00	0,00	0,00	0,65	0,31	0,00	0,00	0,00	0,44	0,00	0,00
S4				1,00	0,65	0,94	0,00	0,00	0,00	0,00	0,00	0,98	0,00	0,00	0,78	0,44	0,44	0,00	0,00	0,00
S5					1,00	0,94	0,00	0,00	0,00	0,00	0,00	0,50	0,00	0,00	0,44	0,81	0,81	0,00	0,00	0,00
S6						1,00	0,00	0,00	0,00	0,00	0,00	0,94	0,00	0,00	0,50	0,50	0,50	0,00	0,00	0,00
S7							1,00	0,94	0,94	0,44	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,50	0,50	0,00
S8								1,00	0,65	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,44	0,94	0,94
S9									1,00	0,33	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,94	0,44	0,00
S10										1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,44	0,94	0,94
S11											1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
S12												1,00	0,00	0,00	0,81	0,44	0,44	0,00	0,00	0,00
S13													1,00	0,33	0,00	0,00	0,00	0,00	0,00	0,00
S14														1,00	0,00	0,00	0,00	0,00	0,00	0,00
S15															1,00	0,00	0,50	0,00	0,00	0,00
S16																1,00	0,00	0,00	0,00	0,00
S17																	1,00	0,00	0,00	0,00
S18																		1,00	0,50	0,00
S19																			1,00	0,50
S20																				1,00

Figura 5.30 – Matriz de similaridade: teste 4

Os parâmetros do AG são os mesmos da tabela 5.2, exceto pelo número de grupos que variou entre 2 e 20 (número de subseqüências comuns). Este teste foi realizado com a função de aptidão largura da silhueta média, visto que o teste 3 demonstrou que tanto a função soma das similaridades médias como a função taxa do critério da variância são inadequadas para geração de módulos em projeto de arranjo físico modular.

Largura da Silhueta Média

A figura 5.31 mostra os agrupamentos formados na população final para a função de aptidão largura da silhueta média.

Algoritmo Genético			
Parâmetros Estatísticas População Inicial População Final			
Cromossomos Grupos			
	K	Aptidão	Grupos
1	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
2	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
3	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
4	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
5	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
6	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
7	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
8	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
9	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
10	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
11	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
12	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
13	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
14	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
15	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]
16	5	0,61215	[[S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20], [S11]]

Execução terminada! 500/500 00:00:00

Figura 5.31 - População final: teste 4, largura da silhueta média

O melhor indivíduo da população final apresentou cinco grupos ($K=5$), com aptidão igual a 0,61215. Para este, os agrupamentos formados foram: [S1, S2, S3, S13, S14], [S4, S5, S6, S12, S15, S16, S17], [S7, S9, S18], [S8, S10, S19, S20] e [S11]. As seguintes análises podem ser feitas:

- As subsequências S13 ($1 \rightarrow 2 \rightarrow 3$) e S14 ($3 \rightarrow 4 \rightarrow 5$) foram designadas corretamente ao primeiro grupo, já formado por S1 ($1 \rightarrow 2$), S2 ($1 \rightarrow 2 \rightarrow 3 \rightarrow 4$) e S3 ($1 \rightarrow 2 \rightarrow 5 \rightarrow 6$). A subsequência S13 está contida em S2 e S14 é formada pelas máquinas 3 e 4, presentes em S2 e pela máquina 5 presente em S3.

- As subsequências S15 ($10 \rightarrow 13$), S16 ($12 \rightarrow 9$) e S17 ($11 \rightarrow 13$) foram designadas corretamente ao segundo grupo, já formado por S4 ($10 \rightarrow 12 \rightarrow 13$), S5 ($11 \rightarrow 12 \rightarrow 13 \rightarrow 9$), S6 ($12 \rightarrow 13$) e S12 ($9 \rightarrow 10 \rightarrow 12 \rightarrow 13$). As máquinas 10 e 13 da subsequência S15 estão presentes em S4 e S12, assim como as máquinas 12 e 9 de S16 em S5 e S12 e as máquinas 11 e 13 de S17 em S5.
- A subsequência S18 ($15 \rightarrow 6$) está contida em S9 ($14 \rightarrow 15 \rightarrow 6$) e foi designada corretamente ao terceiro grupo.
- As subsequências S19 ($15 \rightarrow 16$) e S20 ($16 \rightarrow 17$) estão contidas em S8 ($14 \rightarrow 15 \rightarrow 16 \rightarrow 17$), e S10 ($15 \rightarrow 16 \rightarrow 17$), e foram alocadas corretamente ao quarto grupo.

Este teste mostrou que o algoritmo genético agrupou corretamente as novas subsequências para a função de aptidão largura da silhueta média. O número de grupos permaneceu o mesmo ($K=5$) do teste 3, como era esperado. As novas subsequências (S13 a S20) apenas foram designadas aos grupos já existentes.

5.9 Validação do algoritmo genético de agrupamento

O algoritmo genético de agrupamento (AGA) foi validado através da comparação com os módulos básicos de arranjo físico gerados pelo procedimento baseado em análise de agrupamentos (*cluster analysis*) proposto por Huang (2003). O mesmo conjunto de peças usado por Huang (2003) será usado para validação do AGA. Este conjunto de peças é o mesmo usado no início deste capítulo para explicação do funcionamento do algoritmo, portanto os passos até a geração da matriz de similaridade não serão repetidos, vistos que estão ilustrados nas figuras 5.1 a 5.5.

A tabela 5.6 apresenta o conjunto de peças usado por Huang (2003) com as seqüências de operações (seqüência de máquinas).

Tabela 5.6 – Seqüência de operações: teste de validação 1

Peça (P)	Seqüência de operações (Seq)
1	1, 4, 8, 9
2	1, 4, 7, 4, 8, 7
3	1, 2, 4, 7, 8, 9
4	1, 4, 7, 9
5	1, 6, 10, 7, 9
6	6, 10, 7, 8, 9
7	6, 4, 8, 9
8	3, 5, 2, 6, 4, 8, 9
9	3, 5, 6, 4, 8, 9
10	4, 7, 4, 8
11	6
12	11, 7, 12
13	11, 12
14	11, 7, 10
15	1, 7, 11, 10, 11, 12
16	1, 7, 11, 10, 11, 12
17	11, 7, 12
18	6, 7, 10
19	12

Vakharia e Wemmerlöv apud Huang (2003)

As subsequências comuns únicas geradas a partir das seqüências de operações da tabela 5.6 foram mostradas na figura 5.3, e para efeito didático são repetidas na tabela 5.7.

Tabela 5.7 – Subseqüências comuns únicas

Id.	Subseqüências comuns únicas
S1	$1 \rightarrow 4$
S2	$1 \rightarrow 4 \rightarrow 7$
S3	$1 \rightarrow 7 \rightarrow 11 \rightarrow 10 \rightarrow 11 \rightarrow 12$
S4	$11 \rightarrow 12$
S5	$11 \rightarrow 7$
S6	$11 \rightarrow 7 \rightarrow 12$
S7	$3 \rightarrow 5$
S8	$4 \rightarrow 7$
S9	$4 \rightarrow 7 \rightarrow 4 \rightarrow 8$
S10	$4 \rightarrow 8$
S11	$4 \rightarrow 8 \rightarrow 9$
S12	$6 \rightarrow 10 \rightarrow 7$
S13	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$
S14	$7 \rightarrow 10$
S15	$7 \rightarrow 8 \rightarrow 9$
S16	$7 \rightarrow 9$
S17	$8 \rightarrow 9$

A matriz de similaridade gerada a partir da aplicação do coeficiente *Merger*, mostrada na figura 5.5, apresentou praticamente os mesmos resultados que a de Huang (2003), diferindo em alguns valores apenas na ordem de centésimos.

Como Huang (2003) usou análise de agrupamento (*cluster analysis*), o número de módulos desejado foi especificado em 4 (quatro). Os módulos básicos gerados por Huang (2003) são mostrados na tabela 5.8.

Tabela 5.8 – Módulos básicos de *layout*

Id. Módulo	Grupo de subsequência comum	Módulo de <i>Layout</i>
M1	S7	$3 \rightarrow 5$
M2	S3, S4, S5, S6	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$
M3	S12, S14, S16	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$
M4	S1, S2, S8, S9, S10, S11, S13, S15, S17	1 $\downarrow$ $6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\updownarrow$ 7

Huang (2003)

A tabela 5.8 mostra que o módulo M1 é formado apenas pela subsequência S7, que corresponde às máquinas 3 e 5, conforme mostrado na tabela 5.7. O módulo M2 é formado pelas subsequências S3, S4, S5 e S6, e corresponde às máquinas 1, 7, 10, 11 e 12, e assim por diante. Na coluna Módulo de *Layout*, as setas entre as máquinas representam operações consecutivas, e foram representadas a partir da tabela 5.7.

Como Huang (2003) especificou quatro módulos, os testes de validação foram feitos com a codificação do cromossomo R1, onde o número de grupos é fixo ($K=4$), e também com a codificação R3 com $K_{\min} = K_{\max} = 4$. Na codificação R1 é feito o pré-agrupamento das subsequências para geração da população inicial, como detalhado na seção 4.4. Na codificação R3, a designação das subsequências aos grupos é feita totalmente aleatória. Como o número de grupos deve permanecer constante, não são permitidos os operadores mutação 1 (*merge*) e mutação 2 (*break*).

Os testes de validação foram realizados apenas com a função largura da silhueta média, visto que a soma das similaridades médias e a taxa do critério da variância se mostraram inadequadas nos testes de verificação do algoritmo.

Os testes de validação foram realizados com os primeiros operadores de cruzamento e de mutação 3 desenvolvidos, respectivamente, Recombinação e *Move Direcionado*.

Portanto foram realizados 2 tipos de teste de validação, com R1 e R3 para a função largura da silhueta média. Para cada um deles foram testadas 5 (cinco) configurações de parâmetros do AG: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005. Para o método de geração probabilístico os valores das taxas de cruzamento e mutação 3 foram: ($P_c = 0,85$; $P_{m3} = 0,05$) e ($P_c = 0,50$; $P_{m3} = 0,50$). Para o método faixa percentual os valores dos operadores foram: ($P_c = 0,50$; $P_{m3} = 0,50$), ($P_c = 0,70$; $P_{m3} = 0,30$) e ($P_c = 0,30$; $P_{m3} = 0,70$).

Nos testes de validação foram analisadas as semelhanças e diferenças em relação ao trabalho de Huang, que usou a soma da similaridade em seu modelo matemático (MULVEY e CROWDER apud HUANG, 2003).

Teste de Validação 1: codificação R1

Em todas as cinco configurações de parâmetros foi atingido o valor máximo da função largura da silhueta média, 0,4190. A população final convergiu para o melhor cromossomo, com os seguintes agrupamentos: [S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]. A tabela 5.9 expressa esses agrupamentos em termos dos módulos gerados.

Tabela 5.9 – Teste de Validação 1: módulos gerados

Id. Módulo	Grupo de subsequência comum	Módulo de <i>Layout</i>
M1	S1, S2, S8, S9	$1 \rightarrow 4 \rightarrow 8$ $\updownarrow$ 7
M2	S3, S4, S5, S6	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\searrow \downarrow$ 12
M3	S7, S12, S14, S16	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$ $3 \rightarrow 5$
M4	S10, S11, S13, S15, S17	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\uparrow$ 7

Comparando-se os módulos do trabalho de Huang (tabela 5.8) e os módulos gerados pelo AGA (tabela 5.9), as seguintes análises podem ser feitas:

- O módulo M2 gerado pelo AGA é igual ao de Huang, composto pelas subsequências S3, S4, S5 e S6.
- A única diferença do módulo M3 do AGA em relação ao módulo M3 de Huang é a inclusão da subsequência S7 (3→5) naquele. A inclusão de S7 em M3 pelo algoritmo genético foi realizada com o objetivo de maximizar a função de aptidão, respeitando a condição de que o número de módulos permanecesse fixo em 4, como estabelecido pelo usuário.
- O módulo M4 de Huang foi dividido nos módulos M1 e M4 do AGA, também com o objetivo de maximizar a função de aptidão. Essa divisão gerou necessidade de três máquinas a mais, as máquinas 4, 7 e 8. Os módulos gerados pelo AGA, expressos na tabela 5.9, maximizam a função largura da silhueta média, diferente da função usada por Huang (2003), portanto, não se deve supor que os módulos dos dois trabalhos sejam iguais. Entretanto, será apresentada no capítulo 7 uma base para comparação do arranjo físico final.

Para as configurações (probabilístico, $P_c = 0,50$, $P_{m3} = 0,50$), (faixa percentual, $P_c = 0,70$, $P_{m3} = 0,30$) e (faixa percentual, $P_c = 0,30$, $P_{m3} = 0,70$) em uma das rodadas o AGA encontrou a mesma solução que Huang (2003), entretanto o cromossomo foi o pior classificado, na 200ª posição, com aptidão igual a 0,32235, bem abaixo da máxima. A figura 5.32 mostra os agrupamentos formados.

	K	Aptidão	Grupos
189	4	0,39300	[[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14], [S10, S11, S13, S15, S16, S17]]
190	4	0,39300	[[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14], [S10, S11, S13, S15, S16, S17]]
191	4	0,39300	[[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14], [S10, S11, S13, S15, S16, S17]]
192	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
193	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
194	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
195	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
196	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
197	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
198	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
199	4	0,36500	[[S1, S2, S8, S9], [S3, S4, S5, S6, S7], [S10, S11, S13, S15, S17], [S12, S14, S16]]
200	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]

Execução terminada! 500/500 00:00:01

Figura 5.32 – Teste de Validação 1: cromossomo na 200ª posição

Teste de Validação 2: codificação R3

Em todas as cinco configurações de parâmetros o melhor cromossomo apresentou aptidão igual a 0,4076 com os seguintes agrupamentos: [S1, S2, S8, S9, S10], [S3, S4, S5, S6, S12, S14], [S7], [S11, S13, S15, S16, S17]. A tabela 5.10 mostra esses agrupamentos em termos dos módulos. O valor máximo da função largura da silhueta média, 0,4190, não foi atingido em nenhuma configuração de parâmetros.

Tabela 5.10 – Teste de Validação 2: módulos gerados

Id. Módulo	Grupo de subsequência comum	Módulo de <i>Layout</i>
M1	S1, S2, S8, S9, S10	$1 \rightarrow 4 \rightarrow 8$ $\updownarrow$ 7
M2	S3, S4, S5, S6, S12, S14
M3	S7	$3 \rightarrow 5$
M4	S11, S13, S15, S16, S17	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\uparrow$ 7

Comparando-se os módulos do trabalho de Huang (tabela 5.8) e os módulos gerados pelo AGA (tabela 5.10), as seguintes análises podem ser feitas:

- O algoritmo isolou a subsequência S7 ($3 \rightarrow 5$) em um módulo, como no trabalho de Huang (2003). As máquinas 3 e 5 de S7 não estão presentes em nenhuma outra subsequência comum (tabela 5.7).
- O módulo M4 de Huang foi dividido nos módulos M1 e M4 do AGA.
- As subsequências do módulo M3 de Huang foram separadas nos módulos M2 e M4 do AGA, sendo S12 e S14 para M2 e S16 para M4.
- Os quatro módulos gerados pelo AGA apresentaram o mesmo número de máquinas que Huang (2003), ou seja, 17 máquinas. As diferenças estão no número de alguns tipos de máquinas. Os módulos de Huang (2003) necessitam de 1 máquina do tipo 4, 1 máquina 8, 2 máquinas 9 e 2 máquinas 10, enquanto o AGA gerou 2 máquinas 4, 2 máquinas 8, 1 máquina 9 e 1 máquina 10. O número de tipos das demais máquinas foram os mesmos.

Para a configuração (probabilístico, $P_c = 0,85$, $P_{m3} = 0,05$), em uma das rodadas o AGA encontrou a mesma solução que Huang (2003), entretanto o valor da função de aptidão foi 0,32235, abaixo da aptidão máxima. A figura 5.33 mostra os agrupamentos formados.

	K	Aptidão	Grupos
1	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
2	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
3	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
4	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
5	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
6	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
7	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
8	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
9	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
10	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
11	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]
12	4	0,32235	[[S1, S2, S8, S9, S10, S11, S13, S15, S17], [S3, S4, S5, S6], [S7], [S12, S14, S16]]

Execução terminada! 500/500 00:00:01

Figura 5.33 – Teste de Validação 2: solução idêntica a de Huang (2003)

5.10 Considerações finais

Este capítulo apresentou a implementação do algoritmo genético de agrupamento para geração de módulos de arranjo físico modular, os testes controlados para verificação do algoritmo e os testes de validação do mesmo. A seguir são apresentados os experimentos realizados para análise do comportamento do algoritmo genético de agrupamento.

6 EXPERIMENTAÇÃO

6.1 Considerações iniciais

Este capítulo apresenta os experimentos realizados com o algoritmo genético de agrupamento (AGA). Os experimentos objetivaram analisar o comportamento do AGA frente às várias possibilidades de combinações de parâmetros existentes. Neste capítulo não será analisada a formação dos grupos em termos dos módulos básicos de arranjo físico. Isto será feito no capítulo 7, a partir dos resultados gerados desta experimentação.

Nos experimentos, vários aspectos são analisados, como o número de vezes em que se atingiu o valor máximo de aptidão, a convergência do algoritmo e a eficácia dos operadores de cruzamento e de mutação.

O conjunto de dados de entrada é o mesmo usado no trabalho de Huang (2003), já apresentado na tabela 5.6 com subsequências comuns mostradas na tabela 5.7.

Inicialmente o comportamento do AGA foi analisado com número de módulos ou grupos fixo em quatro, ou seja, $K=4$, como especificado por Huang (2003). Em seguida experimentos foram realizados com o objetivo de avaliar o comportamento no AGA na resolução do problema de agrupamento automático. Neste caso, o AGA deve encontrar sozinho o melhor número e formação de grupos que maximiza a função de aptidão largura da silhueta média.

6.2 Experimentos preliminares

Estes experimentos foram realizados com os primeiros operadores de cruzamento e de mutação 3 desenvolvidos, respectivamente Recombinação e *Move* Direcionado. Estes operadores foram descritos no item 4.6 no capítulo 4.

A configuração de parâmetros usada nos experimentos preliminares foi: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005; número de grupos: 4.

De acordo com COLE (1998), os valores dos parâmetros usados em algoritmos genéticos aplicados a problemas de agrupamento variam consideravelmente. O tamanho da população varia entre 10 e 1000, o número de gerações entre 40 a 200 ou até a convergência completa. Cole (1998) trabalhou com população entre 50 e 400 indivíduos.

No presente trabalho, para determinar o tamanho da população, inicialmente foram realizados experimentos com 50 indivíduos. Entretanto, este número não se mostrou suficiente para representar o conjunto de entrada formado pelas 17 subsequências comuns. Portanto, aumentou-se o tamanho da população para 200 indivíduos. Este número melhor representou o conjunto de dados, produzindo diversidade na população inicial. O número de gerações foi especificado em 500 para permitir a evolução da população.

Segundo Lacerda, Carvalho e Ludermir (1999) o melhor indivíduo pode ser perdido de uma geração para outra devido à ocorrência de cruzamento ou mutação. Portanto, é interessante que o melhor indivíduo seja transferido de uma geração para outra sem alterações. Assim, a taxa de transferência de 0,005 foi escolhida de modo que apenas um indivíduo ($200 \times 0,005 = 1$), o melhor, seja transferido para a próxima geração, para manter as boas soluções.

Para cada combinação de parâmetros foram realizadas 10 rodadas de simulação e obtido o número de vezes em que se atingiu o valor máximo da função de aptidão, igual a

0,4190. O melhor cromossomo, com este valor máximo de aptidão, para número de grupos igual a 4, é formado pelos seguintes agrupamentos: [S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]. Os resultados dos experimentos preliminares são apresentados nas tabelas 6.1 e 6.2.

Tabela 6.1 – Experimentos Preliminares: método probabilístico

Taxa Cruzamento	Taxa Mutação 3	No. Vezes Aptidão Máxima	
		R1	R3
1	0	6	0
1	0,05	7	0
1	0,10	10	0
1	0,30	8	0
1	0,50	9	0
0,85	0	6	0
0,85	0,05	7	0
0,85	0,10	7	0
0,85	0,30	7	0
0,85	0,50	6	0
0,50	0	10	0
0,50	0,05	7	0
0,50	0,10	9	0
0,50	0,30	8	0
0,50	0,50	9	0
0	0,05	8	2
0	0,10	9	1
0	0,30	9	0
0	0,50	10	4
0	1	10	2

Tabela 6.2 – Experimentos Preliminares: método faixa percentual

Taxa Cruzamento	Taxa Mutação 3	No. Vezes Aptidão Máxima	
		R1	R3
1	0	6	0
0,95	0,05	8	0
0,85	0,15	7	0
0,70	0,30	9	0
0,50	0,50	10	0
0,30	0,70	9	0
0	1	10	4

As seguintes análises podem ser feitas a partir dos experimentos preliminares apresentados nas tabelas 6.1 e 6.2:

- Na codificação R1, na qual é feita um pré-agrupamento das seqüências, com a diminuição da taxa de cruzamento houve aumento do número de vezes em que se encontrou a aptidão máxima, independente do método de geração da nova população. Além disso, quando foi aplicada somente mutação, a resposta do algoritmo foi a melhor. Assim, o cruzamento piorou a população.
- Quando se usou a codificação R3, na qual a população inicial é totalmente aleatória, o algoritmo somente encontrou o valor máximo de aptidão quando não foi usado cruzamento, independente do método de geração da nova população.

A codificação R1 já parte de boas soluções iniciais, como descrito no item 4.4 do capítulo 4. Em algumas vezes o valor máximo de aptidão, igual a 0,4190, foi encontrado na própria população inicial. Isso significa que os bons resultados do AGA devem-se à eficácia da otimização realizada em R1 para geração da população inicial, e não à atuação dos operadores de cruzamento e mutação. Portanto, a codificação R1 não deve ser usada para avaliação do desempenho do AGA para o conjunto de peças usado.

A partir das análises dos experimentos preliminares e para melhor avaliação dos operadores de cruzamento do tipo Recombinação e mutação 3 *Move* Direcionado, foram realizados mais dois experimentos com os operadores puros, agora com 30 rodadas de simulação. Esses experimentos são mostrados a seguir.

Experimento 1: cruzamento puro do tipo Recombinação

O objetivo deste experimento é avaliar a eficácia do operador de cruzamento do tipo Recombinação. Os parâmetros do algoritmo genético estão especificados na tabela 6.3.

Tabela 6.3 – Parâmetros: experimento 1

Parâmetros do AG
Tamanho da população: 200
Número de gerações: 500
Taxa de transferência: 0,005
Codificação do cromossomo: R3
No. de grupos: 4
Método de geração: probabilístico
Tipo de cruzamento: Recombinação
Taxa de cruzamento: 1
Tipo de mutação 3: nenhum
Taxa de mutação 3: 0
Função de aptidão: largura da silhueta média
Método de seleção: Roleta

Conforme mostra a tabela 6.3, a taxa de cruzamento foi especificada em 1 (um), ou seja, há 100% de probabilidade do cruzamento ocorrer. O cruzamento do tipo Recombinação foi detalhado no item 4.6.1.1.

O operador de cruzamento foi aplicado 49500 vezes. Este número equivale aproximadamente aos 100% da taxa de cruzamento, uma vez que a o tamanho da população é de 199 indivíduos (1 indivíduo não sofre cruzamento, pois é copiado para a próxima geração por elitismo, conforme taxa de transferência), o critério de parada é de 500 gerações e os indivíduos são tomados aos pares para o cruzamento. Deste total (49500), foram realizados em média (sobre as 30 rodadas) 587,8 cruzamentos. Portanto, em média, a cada rodada, apenas 1,18% da população foi modificada pela ação do operador de cruzamento, resultando em pelo menos um indivíduo diferente dos pais. Este percentual mostra que o cruzamento do

tipo Recombinação tende a gerar descendentes idênticos aos pais, ou seja, os grupos são formados pelas mesmas subsequências dos pais, apesar do rótulo do grupo poder ser diferente.

A figura 6.1 apresenta os valores de aptidão encontrados em 30 rodadas de simulação e sua frequência de ocorrência.

Figura 6.1 – Valores de aptidão: experimento 1

Os valores de aptidão encontrados variaram entre 0,3352 e 0,4076, sendo este o valor mais freqüente (9 vezes), como mostra a figura 6.1. O valor de aptidão máximo, igual a 0,4190, não foi encontrado em nenhuma das 30 rodadas de simulação (0%).

A convergência do algoritmo para o maior valor de aptidão foi muito rápida, 32,66 gerações em média. Aplicando somente o operador Recombinação, com 100% de probabilidade, a população evoluiu até certo ponto e estagnou em uma solução local. Portanto, o operador Recombinação quando aplicado sozinho permitiu pouca evolução da população e não foi capaz de encontrar a melhor solução.

Experimento 2: mutação 3 pura do tipo *Move Direcionado*

O objetivo deste experimento é avaliar a eficácia do operador de mutação 3 do tipo *Move Direcionado*. Os parâmetros do algoritmo genético estão especificados na tabela 6.4.

Tabela 6.4 – Parâmetros: experimento 2

Parâmetros do AG
Tamanho da população: 200
Número de gerações: 500
Taxa de transferência: 0,005
Codificação do cromossomo: R3
No. de grupos: 4
Método de geração: probabilístico
Tipo de cruzamento: nenhum
Taxa de cruzamento: 0
Tipo de mutação 3: <i>Move</i> Direcionado
Taxa de mutação 3: 1
Função de aptidão: largura da silhueta média
Método de seleção: Roleta

Conforme mostra a tabela 6.4, a taxa de mutação 3 foi especificada em 1 (um), ou seja, há 100% de probabilidade da mutação ocorrer. Na mutação 3 do tipo *Move* Direcionado, uma subsequência é retirada aleatoriamente de um grupo e movida para o grupo com o qual possuir a menor dissimilaridade média, como descrito no item 4.6.2.

O operador de mutação 3 foi aplicado 99000 vezes. Este número equivale aproximadamente aos 100% da taxa de mutação 3 (199 indivíduos x 500 gerações). Deste total (99000), foram realizadas em média (sobre as 30 rodadas) 5946,1 mutações, ou seja, o indivíduo resultante foi diferente do pai. Portanto, em média, a cada rodada, apenas 6% da população foi modificada.

A figura 6.2 apresenta os valores de aptidão encontrados em 30 rodadas de simulação e sua frequência de ocorrência.

Figura 6.2 – Valores de aptidão: experimento 2

A figura 6.2 mostra que em 8 das 30 rodadas de simulação (26,66%), o operador de mutação 3 do tipo *Move Direcionado* encontrou o melhor cromossomo com valor de aptidão máximo igual a 0,4190. Os valores de aptidão encontrados variaram entre 0,3673 e 0,4190.

Na população final, em média, 186,6 entre 200 indivíduos, convergiram para a mesma solução. Isso comprova que apesar da alta taxa de mutação (100%), o operador *Move Direcionado* não foi capaz de produzir muita diversidade na população.

Os experimentos 1 e 2 comprovaram o mal desempenho do operador de cruzamento Recombinação e do operador de mutação 3 *Move Direcionado*, apontado nos experimentos preliminares das tabelas 6.1 e 6.2. Portanto, surgiu a necessidade de desenvolvimento de outros operadores de cruzamento e de mutação 3, apresentados a seguir.

6.3 Novos operadores de cruzamento e mutação 3

Como o operador de cruzamento do tipo Recombinação foi ineficaz, foram desenvolvidos dois novos operadores de cruzamento.

O primeiro deles surgiu de uma alteração no mecanismo de correção dos descendentes do operador Recombinação. Neste, quando havia nos descendentes maior ou menor número de grupos que o permitido, a correção era direcionada de acordo com a dissimilaridade entre as subsequências. Na alteração realizada, a correção dos descendentes é feita aleatoriamente. Este operador de cruzamento foi chamado Recombinação com Correção Aleatória, detalhado no item 4.6.1.2 do capítulo 4.

O segundo operador de cruzamento desenvolvido foi chamado Ponto Único de Grupo. Este operador, descrito no item 4.6.1.3, preserva nos descendentes o mesmo número de grupos dos pais.

Como mostrado no experimento 2, o operador *Move* Direcionado não foi capaz de produzir muita diversidade na população. Dessa forma, foi desenvolvido outro operador de mutação 3, o *Move* Aleatório, descrito no item 4.6.2.

Para avaliar o desempenho destes três operadores foram realizados os experimentos 3, 4 e 5, apresentados a seguir.

Experimento 3: cruzamento puro do tipo Recombinação com Correção Aleatória

O objetivo deste experimento é avaliar a eficácia do operador de cruzamento do tipo Recombinação com Correção Aleatória. Os parâmetros do algoritmo genético estão especificados na tabela 6.5.

Tabela 6.5 – Parâmetros: experimento 3

Parâmetros do AG
Tamanho da população: 200
Número de gerações: 500
Taxa de transferência: 0,005
Codificação do cromossomo: R3
No. de grupos: 4
Método de geração: probabilístico
Tipo de cruzamento: Recombinação com Correção Aleatória
Taxa de cruzamento: 1
Tipo de mutação 3: nenhum
Taxa de mutação 3: 0
Função de aptidão: largura da silhueta média
Método de seleção: Roleta

Conforme mostra a tabela 6.5, a taxa de cruzamento foi especificada em 1 (um), ou seja, há 100% de probabilidade do cruzamento ocorrer. O cruzamento do tipo Recombinação com Correção Aleatória difere do operador de Recombinação pela forma de correção dos descendentes, como descrito no item 4.6.1.2.

De 49500 cruzamentos aplicados, foram realizados em média (sobre as 30 rodadas) 2647,6 cruzamentos. Portanto, em média, a cada rodada, 5,34% da população foi modificada pela ação do operador de cruzamento do tipo Recombinação com Correção Aleatória, resultando em pelo menos um indivíduo diferente dos pais. Este percentual, acima do percentual do experimento 1 (1,18%), mostra que o mecanismo de correção aleatória nos descendentes proporcionou aumento da diversidade da população. Na correção aleatória, quando o cruzamento gera um indivíduo com número de grupos menor que o mínimo ($K < K_{\min}$), escolhe-se um grupo aleatoriamente para sofrer divisão, em um ponto de partição também selecionado aleatoriamente. Quando o cruzamento gerar indivíduo com número de grupos maior que o máximo ($K > K_{\max}$) escolhe-se um grupo aleatoriamente para destruir e

inserem-se suas subsequências nos grupos restantes de acordo com a menor dissimilaridade média.

A figura 6.3 apresenta os valores de aptidão encontrados em 30 rodadas de simulação e sua frequência de ocorrência.

Figura 6.3 – Valores de aptidão: experimento 3

O valor de aptidão máximo, igual a 0,4190, foi encontrado em 6 das 30 rodadas de simulação (20%). Os dois melhores valores de aptidão, 0,4076 e 0,4190, foram encontrados 83,33% das rodadas (25 vezes em 30).

A convergência do algoritmo com o operador de cruzamento do tipo Recombinação com Correção Aleatória ocorreu em média, em 82,7 gerações. Este número mostra que a convergência do operador de cruzamento Recombinação com Correção Aleatória foi mais lenta e para valores maiores de aptidão que a convergência do operador Recombinação. Os valores de aptidão variaram entre 0,3985 e 0,4190, como mostra a figura 6.3. Dessa forma, o operador Recombinação com Correção Aleatória permitiu maior evolução da população que o operador Recombinação.

Portanto, pode-se afirmar que o operador de cruzamento do tipo Recombinação com Correção Aleatória é melhor que o operador Recombinação, devido à evolução da população para valores maiores de aptidão.

Experimento 4: cruzamento puro do tipo Ponto Único de Grupo

O objetivo deste experimento é avaliar a eficácia do operador de cruzamento do tipo Ponto Único de Grupo. Os parâmetros do algoritmo genético estão especificados na tabela 6.6.

Tabela 6.6 – Parâmetros: experimento 4

Parâmetros do AG
Tamanho da população: 200
Número de gerações: 500
Taxa de transferência: 0,005
Codificação do cromossomo: R3
No. de grupos: 4
Método de geração: probabilístico
Tipo de cruzamento: Ponto Único de Grupo
Taxa de cruzamento: 1
Tipos de mutação 3: nenhum
Taxa de mutação 3: 0
Função de aptidão: largura da silhueta média
Método de seleção: Roleta

Conforme mostra a tabela 6.6, a taxa de cruzamento foi especificada em 1 (um), ou seja, há 100% de probabilidade do cruzamento ocorrer. O cruzamento do tipo Ponto Único de Grupo preserva nos descendentes o mesmo número de grupos dos cromossomos pais. Este operador está detalhado no item 4.6.1.3.

De 49500 cruzamentos aplicados, foram realizados em média (sobre as 30 rodadas) 1584,2 cruzamentos. Portanto, em média, a cada rodada, 3,2% da população foi modificada pela ação do operador de cruzamento do tipo Ponto Único de Grupo, resultando em pelo

menos um indivíduo diferente dos pais. Este percentual mostra que o nível de modificação da população por este cruzamento foi intermediário entre os de Recombinação e Recombinação com Correção Aleatória.

A figura 6.4 apresenta os valores de aptidão encontrados em 30 rodadas de simulação e sua frequência de ocorrência.

Figura 6.4 – Valores de aptidão: experimento 4

O valor de aptidão máximo, igual a 0,4190, foi encontrado em 2 das 30 rodadas de simulação (6,66%). Os dois melhores valores de aptidão, 0,4076 e 0,4190, foram encontrados em 60% das rodadas (18 vezes em 30).

A convergência do algoritmo com o operador de cruzamento do tipo Ponto Único de Grupo ocorreu em média, em 95,8 gerações, um pouco mais lenta que o operador Recombinação com Correção Aleatória. Apesar dos valores de aptidão encontrados serem bem variados, entre 0,3654 e 0,4190, como mostra a figura 6.4, o operador Ponto Único de Grupo, quando aplicado sozinho, foi capaz de encontrar a melhor solução.

Portanto, o operador Ponto Único de Grupo obteve melhor desempenho que o operador Recombinação (experimento 1). Entretanto, somente com o teste de 100% de probabilidade de cruzamento, não é possível realizar comparações entre o operador Ponto

Único de Grupo e o operador Recombinação com Correção Aleatória (experimento 3). Dessa forma, serão apresentados adiante testes combinados, variando-se as taxas de cruzamento e de mutação.

Experimento 5: mutação 3 pura do tipo *Move* Aleatório

O objetivo deste experimento é avaliar a eficácia do operador de mutação 3 do tipo *Move* Aleatório. Os parâmetros do algoritmo genético estão especificados na tabela 6.7.

Tabela 6.7 – Parâmetros: experimento 5

Parâmetros do AG
Tamanho da população: 200
Número de gerações: 500
Taxa de transferência: 0,005
Codificação do cromossomo: R3
No. de grupos: 4
Método de geração: probabilístico
Tipo de cruzamento: nenhum
Taxa de cruzamento: 0
Tipo de mutação 3: <i>Move</i> Aleatório
Taxa de mutação 3: 1
Função de aptidão: largura da silhueta média
Método de seleção: Roleta

Conforme mostra a tabela 6.7, a taxa de mutação 3 foi especificada em 1 (um), ou seja, há 100% de probabilidade da mutação ocorrer. Na mutação 3 do tipo *Move* Aleatório, uma subsequência é retirada aleatoriamente de um grupo e movida para outro grupo também aleatoriamente, como descrito no item 4.6.2.

De 99000 vezes em que o operador de mutação 3 do tipo *Move* Aleatório foi aplicado, foram realizadas em média (sobre as 30 rodadas) 74228,3 mutações, ou seja, o indivíduo resultante foi diferente do pai. Portanto, em média, a cada rodada, 75% da população foi

modificada. Este número, bem acima dos 6% do operador *Move Direcionado* comprova que a aleatoriedade presente no operador de mutação é fundamental para o aumento da diversidade da população.

Em todas as 30 rodadas de simulação (100%), melhor cromossomo apresentou o valor de aptidão máximo, igual a 0,4190, como ilustra a figura 6.5.

Figura 6.5 – Valores de aptidão: experimento 5

Entre os 200 indivíduos da população final, em média somente os dois primeiros indivíduos apresentaram a mesma solução. O restante da população apresentou valores de aptidão bem variados. Isso ocorreu devido à diversificação da população pelo operador de mutação 3 *Move Aleatório*. Além disso, este operador é altamente eficaz, e consegue, sozinho, quando aplicado com alta probabilidade, encontrar a melhor solução com aptidão máxima.

6.4 Experimentos combinados: número de grupos fixo

Os experimentos 1, 3 e 4 foram realizados respectivamente com os operadores de cruzamento do tipo Recombinação, Recombinação com Correção Aleatória e Ponto Único de Grupo. O operador de cruzamento Recombinação com Correção Aleatória se mostrou muito

superior ao operador do tipo Recombinação, como apresentado no experimento 3. O operador Ponto Único de Grupo foi eficaz em encontrar o valor máximo da função de aptidão, entretanto os valores de aptidão obtidos foram muito diversificados, como apresentado na figura 6.4. Por isso novos testes foram necessários com este operador de cruzamento.

O operador de mutação 3 do tipo *Move* Aleatório (experimento 5) foi mais eficaz que o operador *Move* Direcionado (experimento 2), pois foi capaz de encontrar a aptidão máxima em 100% das rodadas de simulação. O operador *Move* Direcionado somente a encontrou em 26,66% das rodadas.

Portanto, foram realizados experimentos combinados com os dois melhores operadores de cruzamento, Recombinação com Correção Aleatória e Ponto Único de Grupo, e com o operador de mutação 3 *Move* Aleatório.

Os parâmetros do algoritmo genético foram os mesmos dos experimentos 1 a 5, ou seja: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005; codificação do cromossomo: R3; número de grupos: 4; função de aptidão: largura da silhueta média; método de seleção: roleta.

Foram realizados experimentos para os dois métodos de geração da nova população, probabilístico e por faixa percentual. Esses experimentos são apresentados a seguir.

6.4.1 Método probabilístico

No método probabilístico, como descrito no item 4.7 do capítulo 4, a cada operador é associada uma probabilidade de ocorrência. As probabilidades dos operadores para os experimentos combinados são apresentadas na tabela 6.8, assim como a identificação dos experimentos.

Tabela 6.8 – Experimentos combinados: método probabilístico, número de grupos fixo

Operador de Cruzamento	Probabilidade de Cruzamento (P_c)	Probabilidade de mutação 3 <i>Move</i> Aleatório (P_{m_3})	Identificação do Experimento
Recombinação com Correção Aleatória	0,85	0,05	Experimento 6
		0,10	Experimento 7
		0,50	Experimento 8
	0,50	0,05	Experimento 9
		0,10	Experimento 10
		0,50	Experimento 11
	0,30	0,70	Experimento 36
	0,70	0,30	Experimento 37
Ponto Único de Grupo	0,85	0,05	Experimento 12
		0,10	Experimento 13
		0,50	Experimento 14
	0,50	0,05	Experimento 15
		0,10	Experimento 16
		0,50	Experimento 17
	0,30	0,70	Experimento 38
	0,70	0,30	Experimento 39

Os valores dos parâmetros de algoritmos genéticos aplicados a problemas de agrupamento variam consideravelmente. A literatura (COLE, 1998; LACERDA; CARVALHO; LUDERMIR, 1999; GOLDBERG, 1989) recomenda o uso de altas taxas de cruzamento (50% - 100%) e baixas taxas de mutação (1% - 10%). Neste trabalho, as probabilidades de cruzamento usadas foram de 85% e 50% (tabela 6.8). A taxa de mutação deve garantir a diversidade da população. Segundo aqueles autores, valores de mutação muito altos podem danificar a população, destruindo indivíduos considerados bons. Entretanto, como no presente trabalho o operador de mutação *Move* Aleatório se mostrou muito eficaz quando aplicado sozinho com alta probabilidade (experimento 5), para cada valor de cruzamento (85% e 50%) foram experimentados três valores de mutação: 5%, 10% e 50%. Outras duas combinações de cruzamento e mutação também foram usadas, 30% de cruzamento com 70% de mutação e vice-versa. Essas duas combinações foram

experimentadas para que adiante se possam fazer comparações entre os dois métodos de geração da nova população, probabilístico e por faixa percentual.

Para cada experimento da tabela 6.8 foram realizadas 30 rodadas de simulação. A figura 6.6 apresenta os valores de aptidão encontrados para os testes combinados usando o operador de cruzamento Recombinação com Correção Aleatória e o operador de mutação 3 *Move* Aleatório.

Figura 6.6 – Experimentos combinados, número de grupos fixo, método probabilístico: cruzamento Recombinação com Correção Aleatória e mutação *Move* Aleatório

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.6:

- A utilização de 50% de taxa de cruzamento foi mais eficaz comparada à taxa de 85%, para qualquer valor de mutação.
- Com altas taxas de mutação (50% e 70%), a melhor solução, 0,4190 é encontrada maior número de vezes.
- Para o conjunto de peças usado (tabela 5.6), o algoritmo genético apresentou melhor desempenho com 70% de taxa de mutação e 30% de taxa de cruzamento. A figura 6.6g mostra que a aptidão máxima, 0,4190, foi encontrada em 100% das rodadas de simulação.

A figura 6.7 apresenta os valores de aptidão encontrados para os experimentos combinados usando o operador de cruzamento Ponto Único de Grupo e o operador de mutação 3 *Move* Aleatório.

(a) Experimento 12: $P_c = 0,85$; $P_{m_3} = 0,05$ (d) Experimento 15: $P_c = 0,50$; $P_{m_3} = 0,05$ (b) Experimento 13: $P_c = 0,85$; $P_{m_3} = 0,10$ (e) Experimento 16: $P_c = 0,50$; $P_{m_3} = 0,10$ (c) Experimento 14: $P_c = 0,85$; $P_{m_3} = 0,50$ (f) Experimento 17: $P_c = 0,50$; $P_{m_3} = 0,50$ (g) Experimento 38: $P_c = 0,30$; $P_{m_3} = 0,70$ (h) Experimento 39: $P_c = 0,70$; $P_{m_3} = 0,30$

Figura 6.7 – Experimentos combinados, número de grupos fixo, método probabilístico: cruzamento Ponto Único de Grupo e mutação *Move* Aleatório

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.7:

- Para 5% de mutação (figuras 6.7a e 6.7d), se considerarmos o número de vezes em que a solução máxima foi obtida, pode-se dizer que o desempenho do algoritmo foi ligeiramente melhor quando se usou 85% de cruzamento. Já para 10% de mutação (figuras 6.7b e 6.7e), quando se usou 50% de cruzamento, o processo se inverteu. Entretanto, se considerarmos os dois melhores valores de aptidão, 0,4076 e 0,4190, observamos que esta diferença é mínima. A figura 6.7a mostra que as duas melhores soluções foram obtidas 29 vezes, contra 28 vezes na figura 6.7d. Na figura 6.7b, as duas melhores soluções foram obtidas 28 vezes, contra 29 vezes na figura 6.7e.
- Para 50% de mutação, o desempenho do algoritmo foi idêntico para as duas taxas de cruzamento, 85% (figura 6.7c) e 50% (figura 6.7f).
- Tanto para 85% como para 50% de taxa de cruzamento, o algoritmo genético apresentou melhor resultado com 50% de taxa de mutação.
- Para o conjunto de peças usado (tabela 5.6), a configuração que apresentou melhor desempenho foi com 70% de mutação e 30% de cruzamento, obtendo a melhor solução em 100% das rodadas de simulação.

Pelas figuras 6.6 e 6.7 pode-se concluir que para o conjunto de peças usado nos experimentos (tabela 5.6), e com o método de geração da nova população do tipo probabilístico, o desempenho do algoritmo é melhor com alta taxa de mutação (50% e 70%). As figuras 6.6 e 6.7 apontam também semelhanças no comportamento de ambos os operadores de cruzamento, mostrando um padrão de soluções geradas para cada configuração de parâmetros e a conseqüente convergência dos operadores. Portanto, pode-se dizer que, para

o conjunto de peças usado (tabela 5.6), quando o usuário define o número grupos ou módulos desejado (K fixo) e usa o método probabilístico, não há diferença significativa entre o operador de cruzamento Recombinação com Correção Aleatória e Ponto Único de Grupo.

6.4.2 Método faixa percentual

No método faixa percentual, os operadores são aplicados a todos os indivíduos sorteados, conforme a ordem de seleção, como descrito no item 4.7 do capítulo 4. As faixas percentuais dos operadores para os experimentos combinados são apresentadas na tabela 6.9.

Tabela 6.9 – Experimentos combinados: método faixa percentual, número de grupos fixo

Operador de Cruzamento	Probabilidade de Cruzamento (P_c)	Probabilidade de mutação 3 <i>Move</i> Aleatório (P_{m_3})	Identificação do Experimento
Recombinação com Correção Aleatória	0,30	0,70	Experimento 30
	0,50	0,50	Experimento 31
	0,70	0,30	Experimento 32
Ponto Único de Grupo	0,30	0,70	Experimento 33
	0,50	0,50	Experimento 34
	0,70	0,30	Experimento 35

Os valores de 50% para cruzamento e mutação foram os mesmos usados por Hruschka (2001), que propôs o método faixa percentual. Os valores 30% para cruzamento e 70% para mutação e vice-versa foram definidos de modo a cumprir a soma de 100% dos percentuais dos operadores.

Para cada experimento da tabela 6.9 foram realizadas 30 rodadas de simulação. A figura 6.8 apresenta os valores de aptidão encontrados para os experimentos combinados usando o operador de cruzamento Recombinação com Correção Aleatória e o operador de mutação 3 *Move* Aleatório.

Figura 6.8 – Experimentos combinados, número de grupos fixo, método faixa percentual: cruzamento Recombinação com Correção Aleatória e mutação *Move Aleatório*

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.8:

- O AGA encontrou a aptidão máxima de 0,4190, com grande frequência, para as três configurações de parâmetros.
- Para valores altos de mutação, 50% (figura 6.8b) e 70% (figura 6.8a), o AGA encontrou a aptidão máxima de 0,4190 em 100% das rodadas.
- Quando se usa o operador Recombinação com Correção Aleatória com o método de geração por faixa percentual, o aumento da taxa de cruzamento aliada à diminuição da taxa de mutação 3, reduz o número de vezes em que a aptidão máxima é encontrada.

A figura 6.9 apresenta os valores de aptidão encontrados para os experimentos combinados usando o operador de cruzamento Ponto Único de Grupo e o operador de mutação 3 *Move* Aleatório.

Figura 6.9 – Experimentos combinados, número de grupos fixo, método faixa percentual: cruzamento Ponto Único de Grupo e mutação *Move* Aleatório

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.9:

- O AGA encontrou a aptidão máxima de 0,4190, com grande frequência, para as três configurações de parâmetros.

- A configuração que apresentou melhor resultado foi a com 30% de cruzamento e 70% de mutação, onde a aptidão máxima de 0,4190 foi encontrada em 100% das rodadas.
- Quando se usa o operador Ponto Único de Grupo com o método de geração por faixa percentual, o aumento da taxa de cruzamento aliado à diminuição da taxa de mutação 3, reduz o número de vezes em que a aptidão máxima é obtida.

6.4.3 Comentários: número de grupos fixo

Pode-se concluir, a partir das figuras 6.6 a 6.9, que, para o conjunto de peças usado nos experimentos (tabela 5.6), independente do método de geração da nova população, probabilístico ou por faixa percentual, o desempenho do algoritmo genético de agrupamento é melhor com altas taxas de mutação (a partir de 50%). As figuras 6.6 a 6.9 apontam também semelhanças no comportamento de ambos os operadores de cruzamento, para ambos os métodos de geração. Há um padrão de soluções geradas para cada configuração de parâmetros, indicando a convergência dos dois operadores de cruzamento. Portanto, pode-se dizer que, para o conjunto de peças usado (tabela 5.6), quando o usuário define o número de grupos ou módulos desejado (K fixo), independente do método de geração da nova população usado, não há diferença significativa entre o operador de cruzamento Recombinação com Correção Aleatória e Ponto Único de Grupo.

Comparando-se o método probabilístico e o método faixa percentual através das mesmas configurações de parâmetros ($P_c = 0,70$, $P_{m_3} = 0,30$; $P_c = 0,50$, $P_{m_3} = 0,50$; $P_c = 0,30$, $P_{m_3} = 0,70$) observa-se comportamento semelhante entre os dois métodos.

Os resultados e análises dos experimentos com número de grupos fixo mostraram que o algoritmo genético de agrupamento (AGA) convergiu para a aptidão máxima, e, portanto,

foi eficaz em resolver o problema de K agrupamentos, ou seja, aquele em que o usuário define *a priori* o número de módulos.

6.5 Experimentos combinados: número de grupos variável

O objetivo dos experimentos combinados com número de grupos variável é avaliar o comportamento do algoritmo genético de agrupamento (AGA) quando da resolução do problema de agrupamento automático. Neste, o usuário não especifica o número desejado de grupos ou módulos. O AGA deve encontrar sozinho o melhor número e configuração de grupos para o qual o valor da função de aptidão largura de silhueta média é máximo.

Como o conjunto de dados é o mesmo dos experimentos anteriores, formado por 17 subsequências comuns, o número de grupos mínimo ($K_{\min}$) foi definido em 2 e o máximo ($K_{\max}$) em 16. A função largura da silhueta média opera com no mínimo dois grupos. O número máximo foi definido como o número de subsequências menos um. Se o valor de $K_{\max}$ fosse 17, o valor da silhueta seria zero, com cada subsequência em um grupo.

Para avaliar o comportamento do AGA com número de grupos variável, os parâmetros do algoritmo genético foram os mesmos dos experimentos anteriores, ou seja: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005; codificação do cromossomo: R3; função de aptidão: largura da silhueta média; método de seleção: roleta.

Quando se trabalha com número de grupos variável, o AGA permite a escolha da codificação do cromossomo do tipo R2 ou R3, como detalhado no item 4.3. A codificação R2 é idêntica à R1, a não ser pela inclusão de um gene representando o número de grupos do cromossomo. Ambas já partem de boas soluções iniciais. Como demonstrado nos experimentos preliminares (item 6.2), este tipo de codificação não deve ser aplicada para avaliação do desempenho do AGA para o conjunto de peças usado. Dessa forma, os

experimentos com número de grupos variável foram realizados somente com a codificação R3, na qual a população inicial é gerada aleatoriamente.

Quando o número de grupos é variável, o AGA pode usar todos os três operadores de mutação. Dessa forma, dois tipos de experimentos foram realizados. No primeiro, os operadores de mutação 1 e 2 foram mantidos zerados, para que se pudesse comparar a resposta do AGA com número de grupos variável com número de grupos fixo. No segundo, todos os três operadores de mutação foram usados. Esses experimentos são detalhados a seguir.

6.5.1 Número de grupos variável: sem operadores de mutação 1 e 2

Os experimentos com número de grupos variável entre 2 e 16, e sem o uso dos operadores de mutação 1 (*Merge*) e 2 (*Break*) foram realizados com o método de geração probabilístico. A tabela 6.10 mostra as probabilidades de aplicação dos operadores e a identificação dos experimentos.

Tabela 6.10 – Experimentos combinados: número de grupos variável

Operador de Cruzamento	Probabilidade de Cruzamento (Pc)	Probabilidade de mutação 3 <i>Move</i> Aleatório (Pm ₃)	Identificação do Experimento
Recombinação com Correção Aleatória	0,85	0,05	Experimento 18
		0,10	Experimento 19
		0,50	Experimento 20
	0,50	0,05	Experimento 21
		0,10	Experimento 22
		0,50	Experimento 23
Ponto Único de Grupo	0,85	0,05	Experimento 24
		0,10	Experimento 25
		0,50	Experimento 26
	0,50	0,05	Experimento 27
		0,10	Experimento 28
		0,50	Experimento 29

Para cada experimento da tabela 6.10 também foram realizadas 30 rodadas de simulação. A figura 6.10 apresenta os valores de aptidão encontrados para os testes combinados usando o operador de cruzamento Recombinação com Correção Aleatória e o operador de mutação 3 *Move* Aleatório.

Figura 6.10—Experimentos combinados, número de grupos variável, método probabilístico: cruzamento Recombinação com Correção Aleatória e mutação *Move Aleatório*

Nos gráficos da figura 6.10, as colunas representam a frequência dos valores de aptidão e a linha indica o número de grupos encontrado pelo AGA para cada valor de aptidão.

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.10:

- O valor máximo da função de aptidão foi 0,6110. O cromossomo com aptidão máxima apresenta 8 (oito) grupos e é formado pelas subsequências: [S1, S2], [S3, S4], [S5, S6], [S7], [S8, S9], [S10, S11, S13], [S12, S14, S16], [S15, S17]. Portanto, 8 é o melhor número de módulos para o conjunto de dados usado (tabela 5.6). Com 8 grupos, o valor da função de aptidão largura da silhueta média é maximizado.
- A configuração de parâmetros em que mais se encontrou o valor de aptidão máximo 0,6110 foi o experimento 22 (figura 6.10e) com 50% de cruzamento e 10% de mutação. Entretanto, somente em 10 entre 30 rodadas de simulação, este valor foi encontrado. Isso representa apenas 1/3 (um terço) de probabilidade de acerto.
- Em todas as configurações (figuras 6.10a a 6.10f) o AGA gerou soluções variadas, o que demonstra a não convergência do algoritmo para a melhor solução.

A figura 6.11 apresenta os valores de aptidão encontrados para os experimentos combinados usando o operador de cruzamento Ponto Único de Grupo e o operador de mutação 3 *Move* Aleatório.

Figura 6.11 – Experimentos combinados, número de grupos variável, método probabilístico: cruzamento Ponto Único de Grupo e mutação *Move* Aleatório

Nos gráficos da figura 6.11, as colunas representam a frequência dos valores de aptidão e a linha indica o número de grupos encontrado pelo AGA para cada valor de aptidão.

As seguintes análises podem ser feitas a partir dos experimentos combinados apresentados na figura 6.11:

- O valor de aptidão máximo, 0,6110, foi encontrado, para a maioria das configurações de parâmetros, somente 2 vezes em 30 rodadas, o que representa apenas 6,66%.
- A figura 6.11 mostra que o AGA apresentou uma gama de soluções, independente da configuração de parâmetros usada. Portanto, o algoritmo não convergiu para a melhor solução.

Nos experimentos apresentados nas figura 6.10 e 6.11 foram aplicados apenas os operadores de cruzamento e mutação 3 (*Move*). Os operadores de mutação 1 (*Merge*) e mutação 2 (*Break*) foram mantidos zerados. Esses operadores, *Merge* e *Break* são os que alteram a estrutura dos grupos, ou seja, o número de grupos e o tamanho dos mesmos no cromossomo. Assim, como esses operadores não foram usados, o número de grupos de um indivíduo da população inicial foi mantido ao longo do processo de evolução. Se um indivíduo não possuir 8 grupos na população inicial (oito é o número de grupos da aptidão máxima, 0,6110), o AGA, sem os operadores de mutação 1 e 2, respectivamente *Merge* e *Break*, não encontra o melhor número de grupos.

Os experimentos das figuras 6.10 e 6.11 foram feitos com o método probabilístico. Entretanto, como, sem os operadores *Merge* e *Break* o AGA não encontrou o melhor número de grupos, não foram realizados experimentos para o método de geração por faixa percentual.

6.5.2 Número de grupos variável: todos os operadores

Para avaliar a resposta do AGA na resolução do problema de agrupamento automático, ou seja, aquele em que o algoritmo deve encontrar o melhor número de grupos ou módulos, foram realizados experimentos onde são aplicados todos os operadores, de cruzamento, de mutação 1 (*Merge*), 2 (*Break*) e 3 (*Move*), para os dois métodos de geração da nova população, probabilístico e por faixa percentual. Os experimentos são apresentados a seguir.

6.5.2.1 Método probabilístico

As probabilidades de aplicação dos operadores e a identificação dos experimentos são apresentadas na tabela 6.11, a seguir.

A primeira coluna da tabela 6.11 refere-se ao tipo de operador de cruzamento usado, Recombinação Aleatória ou Ponto Único de Grupo. As probabilidades de cruzamento são especificadas na segunda coluna por P_c . As probabilidades de mutação são representadas por P_{m_1} (mutação 1 ou *Merge*), P_{m_2} (mutação 2 ou *Break*) e P_{m_3} (mutação 3 ou *Move*). A última coluna mostra a identificação do experimento.

Tabela 6.11 – Experimentos combinados: número de grupos variável, método probabilístico

Operador de Cruzamento	Pc	Pm ₁	Pm ₂	Pm ₃	Identificação do Experimento
Recombinação com Correção Aleatória	0,85	0,05	0,05	0,05	Experimento 40
		0,05	0,50	0,05	Experimento 41
		0,05	0,50	0,10	Experimento 42
		0,05	0,50	0,50	Experimento 43
		0,10	0,50	0,05	Experimento 44
		0,50	0,50	0,05	Experimento 45
	0,50	0,05	0,05	0,05	Experimento 46
		0,05	0,50	0,05	Experimento 47
		0,05	0,50	0,10	Experimento 48
		0,05	0,50	0,50	Experimento 49
		0,10	0,50	0,05	Experimento 50
		0,50	0,50	0,05	Experimento 51
Ponto Único de Grupo	0,85	0,05	0,05	0,05	Experimento 60
		0,05	0,50	0,05	Experimento 61
		0,05	0,50	0,10	Experimento 62
		0,05	0,50	0,50	Experimento 63
		0,10	0,50	0,05	Experimento 64
		0,50	0,50	0,05	Experimento 65
	0,50	0,05	0,05	0,05	Experimento 66
		0,05	0,50	0,05	Experimento 67
		0,05	0,50	0,10	Experimento 68
		0,05	0,50	0,50	Experimento 69
		0,10	0,50	0,05	Experimento 70
		0,50	0,50	0,05	Experimento 71

As taxas de cruzamento usadas foram 0,85 e 0,50 (tabela 6.11), como nos experimentos anteriores para o método probabilístico. Em relação aos valores das taxas de mutação, a seguinte estratégia foi usada:

- Inicialmente foram atribuídos valores pequenos (0,05) aos três tipos de mutação (experimentos 40, 46, 60, 66);
- Em seguida aumentou-se o valor da mutação 2 (*Break*) para 0,50, mantidos os valores das mutações 1 e 3, e foram observados os resultados (experimentos 41, 47, 61 e 67);
- Para verificar o efeito da mutação 3 (*Move*), os valores das mutações 1 e 2 foram mantidos em 0,05 e 0,50 respectivamente, e aumentou-se o valor de Pm₃ para 0,10 e 0,50, e foram observados os resultados (experimentos 42, 43, 48, 49, 62, 63, 68 e 69);

- Por fim observou-se o efeito do valor da mutação 1 (*Merge*) para altas taxas de mutação 2 (*Break*) e baixas taxas de mutação 3 (*Move*), através dos experimentos 44, 45, 50, 51, 64, 65, 70 e 71.

Para cada experimento da tabela 6.11 foram realizadas 30 rodadas de simulação. A figura 6.12 apresenta os valores de aptidão encontrados para os testes combinados usando 85% de cruzamento do tipo Recombinação com Correção Aleatória, os três operadores de mutação, com o método probabilístico.

Figura 6.12–Experimentos combinados, número de grupos variável, método probabilístico: 0,85 de cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3

A figura 6.12 apresenta os resultados dos experimentos combinados com número de grupos aberto entre 2 e 16 grupos, para que o AGA possa encontrar o melhor número de

configuração de grupos que maximiza a função de aptidão. Para esses experimentos foi usada a taxa de 85% de cruzamento do tipo Recombinação com Correção Aleatória. Todos os operadores de mutação foram aplicados. Nos gráficos da figura 6.12, as colunas representam a frequência dos valores de aptidão e a linha indica o número de grupos encontrado pelo AGA para cada valor de aptidão. As seguintes análises podem ser feitas a partir dos experimentos apresentados na figura 6.12:

- As configurações de parâmetros que apresentaram melhores resultados, em termos do número de vezes em que a aptidão máxima de 0,6110 foi encontrada, foram as com baixas taxas mutação 1 (*Merge*) e 3 (*Move*) e altas taxas de mutação 2 (*Break*) (gráficos (b), (c) e (e)).
- Mantidas baixa taxa de *Merge* ($Pm_1 = 0,05$) e alta de *Break* ($Pm_2 = 0,50$), o aumento da taxa de mutação 3 (*Move*) piora o resultado. Isso é mostrado no gráfico (d), onde foi aplicado 50% de mutação 3.
- O aumento da taxa de *Merge* ($Pm_1 = 0,50$) também contribui negativamente para o resultado, como apresentado no gráfico (f).
- Para as melhores configurações (gráficos (b), (c) e (e)), a aptidão máxima foi encontrada em 100% das rodadas de simulação. O cromossomo com aptidão máxima apresenta 8 (oito) grupos.

A seguir, na figura 6.13, são mostrados os experimentos com 50% de cruzamento do tipo Recombinação com Correção Aleatória.

Figura 6.13–Experimentos combinados, número de grupos variável, método probabilístico: 0,50 de cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3

As seguintes análises podem ser feitas a partir dos experimentos apresentados na figura 6.13:

- As configurações de parâmetros que apresentaram melhores resultados foram as mesmas que para 85% de cruzamento. Essas configurações usaram baixas taxas mutação 1 (*Merge*) e 3 (*Move*) e altas taxas de mutação 2 (*Break*) (gráficos (b), (c) e (e)).
- O aumento da taxa de mutação 3 (*Move*), mantidos baixo *Merge* ($Pm_1 = 0,05$) e alto *Break* ($Pm_2 = 0,50$), é bem menos sensível que quando se usou 85% de cruzamento, como mostrado no gráfico (d). Com 50% de cruzamento, mesmo com alto *Merge*, a aptidão máxima foi encontrada em 29 das 30 rodadas.

Para o operador Recombinação com Correção Aleatória, a diminuição da taxa de cruzamento de 85% para 50% melhora o resultado. Este fato é comprovado pelo aumento do número de vezes em que a aptidão máxima foi encontrada, mostrado nos gráficos (a), (d) e (f) das figuras 6.12 e 6.13.

As figuras 6.12 e 6.13 mostraram a convergência do AGA para a melhor solução com o operador de cruzamento Recombinação com Correção Aleatória. A aptidão máxima foi encontrada maior número de vezes em todas as configurações de parâmetros, com 100% de acerto para as melhores configurações.

A seguir, nas figuras 6.14 e 6.15, são apresentados os testes combinados usando o operador de cruzamento Ponto Único de Grupo com os três operadores de mutação, para o método probabilístico.

Figura 6.14 – Experimentos combinados, número de grupos variável, método probabilístico: 0,85 de cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3

Os experimentos da figura 6.14 apresentam os resultados com 85% de taxa de cruzamento do tipo Ponto Único de Grupo. Todos os operadores de mutação foram aplicados.

As seguintes análises podem ser feitas:

- Da mesma forma que para o operador Recombinação com Correção Aleatória, as configurações de parâmetros que apresentaram melhores resultados, em termos do número de vezes em que a aptidão máxima de 0,6110 foi encontrada, foram as com baixas taxas mutação 1 (*Merge*) e 3 (*Move*) e altas taxas de mutação 2 (*Break*) (gráficos (b), (c) (d) e (e)).
- Uma diferença do operador Ponto Único de Grupo para o anterior pode ser observada no gráfico (a) quando se aplica 5% para as três mutações. Neste caso o número de diferentes soluções geradas foi maior com o operador Ponto Único de Grupo, mostrando pior resultado para baixas taxas de mutação 1, 2 e 3.
- Outra diferença do operador anterior pode ser vista no gráfico (d). Com o operador Ponto Único de Grupo, o aumento da taxa de mutação 3 (*Move*) para 0,50, mantidos baixa taxa de *Merge* ($Pm_1 = 0,05$) e alta de *Break* ($Pm_2 = 0,50$) não piora o resultado. Mais que isso, atinge 100% de acerto em encontrar a aptidão máxima.
- A mutação do tipo *Merge*, quando aplicada com altas taxas ($Pm_1 = 0,50$), também contribuiu negativamente para o resultado, da mesma forma que quando usada com o operador de cruzamento anterior (gráfico (f)).
- Para as melhores configurações (gráficos (b), (c), (d) e (e)), a aptidão máxima foi encontrada em 100% das rodadas de simulação.

A figura 6.15 a seguir mostra os resultados dos experimentos com 50% de taxa de cruzamento do tipo Ponto Único de Grupo. Todos os operadores de mutação foram aplicados.

Figura 6.15 – Experimentos combinados, número de grupos variável, método probabilístico: 0,50 de cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3

As seguintes análises podem ser feitas a partir dos experimentos da figura 6.15:

- Para as configurações de parâmetros apresentadas nos gráficos (b), (c), (d) e (e), houve 100% de acerto em encontrar a aptidão máxima de 0,6110, da mesma forma que para 85% de cruzamento (figura 6.14).

- Com baixas taxas das três mutações (0,05) (gráfico (a)), o resultado melhorou com a redução da taxa de cruzamento de 85% para 50%. O número de vezes em que a aptidão máxima foi encontrada passou de 12 para 22 respectivamente, em 30 rodadas de simulação.
- Comparando-se os gráficos (f) das figuras 6.14a e 6.15b também observam-se melhores resultados com a redução do cruzamento de 85% para 50%. O número de vezes em que a aptidão máxima foi encontrada passou de 14 para 29 em 30 rodadas de simulação.

Dessa forma, para o operador Ponto Único de Grupo, a taxa de 50% de cruzamento é mais indicada.

Os experimentos das figura 6.12 a 6.15 mostraram a convergência de ambos os operadores de cruzamento, Recombinação com Correção Aleatória e Ponto Único de Grupo. A diferença no comportamento deles é muito pequena. Entretanto, os melhores resultados foram obtidos com o operador Ponto Único de Grupo com 50% de taxa de cruzamento.

6.5.2.2 Método faixa percentual

As taxas de aplicação dos operadores e a identificação dos experimentos são apresentadas na tabela 6.12.

Tabela 6.12 – Experimentos combinados: número de grupos variável, método faixa percentual

Operador de Cruzamento	Pc	Pm ₁	Pm ₂	Pm ₃	Identificação do Experimento
Recombinação com Correção Aleatória	0,85	0,05	0,05	0,05	Experimento 52
	0,50	0,40	0,05	0,05	Experimento 53
	0,50	0,05	0,40	0,05	Experimento 54
	0,30	0,05	0,50	0,15	Experimento 55
	0,10	0,05	0,60	0,25	Experimento 56
	0,35	0,20	0,40	0,05	Experimento 57
	0,25	0,30	0,40	0,05	Experimento 58
	0,05	0,40	0,50	0,05	Experimento 59
Ponto Único de Grupo	0,85	0,05	0,05	0,05	Experimento 72
	0,50	0,40	0,05	0,05	Experimento 73
	0,50	0,05	0,40	0,05	Experimento 74
	0,30	0,05	0,50	0,15	Experimento 75
	0,10	0,05	0,60	0,25	Experimento 76
	0,35	0,20	0,40	0,05	Experimento 77
	0,25	0,30	0,40	0,05	Experimento 78
	0,05	0,40	0,50	0,05	Experimento 79

Nos experimentos com o método faixa percentual, a soma das taxas dos operadores deve ser igual a 1, como descrito no item 4.7 do capítulo 4. Em relação aos valores das taxas dos operadores (tabela 6.12), a seguinte estratégia foi usada:

- Inicialmente foram atribuídos valores pequenos (0,05) aos três tipos de mutação (experimentos 52 e 72);
- Em seguida diminuiu-se a taxa de cruzamento (0,50) e aumentou-se a taxa de mutação 1 (*Merge*) (0,40) e observaram-se os resultados (experimentos 53 e 73);
- O mesmo processo anterior foi feito para o operador de mutação 2 (*Break*), mantida a taxa de 0,50 de cruzamento (experimentos 54 e 74);
- Em seguida reduziu-se ainda mais a taxa de cruzamento. Isso foi realizado com baixas taxas de mutação 1 (*Merge*) (experimentos 55, 56, 75 e 76), e em seguida com baixas taxas de mutação 3 (*Move*) (experimentos 57, 58, 59, 77, 78 e 79). Para esses experimentos foram mantidas altas taxas de mutação 2 (*Break*) em virtude do desempenho dos experimentos anteriores.

A figura 6.16 apresenta os valores de aptidão encontrados para os testes combinados usando o operador de cruzamento Recombinação com Correção Aleatória, os três operadores de mutação, com o método faixa percentual.

Figura 6.16 – Experimentos combinados, número de grupos variável, método faixa percentual: cruzamento tipo Recombinação com Correção Aleatória e mutações 1, 2 e 3

As seguintes análise podem ser feitas a partir dos gráficos apresentados na figura 6.16:

- A configuração apresentada no gráfico (a) com alta taxa de cruzamento e baixas taxas de mutação não foi satisfatória. A aptidão máxima foi encontrada em menos de 1/3 (um terço) das 30 rodadas de simulação.
- A redução do cruzamento e aumento da mutação 1, mantidas mutações 2 e 3 em 5% pioraram os resultados (gráfico (b)).
- Por outro lado, o resultado é ótimo quando se reduz o cruzamento, aumenta-se a taxa de mutação 2 e se mantém as taxas de mutação 1 e 3 em valores baixos (gráficos (c) e (d)).
- Altas taxas de mutação 2 (*Break*), 40%, 50% e 60%, são fundamentais para se encontrar o valor da aptidão máxima de 0,6110 (gráficos (c), (d), (e), (f), (g) e (h)).
- A diminuição da taxa de cruzamento mantida alta taxa de mutação 2 (*Break*), não afetou o desempenho do algoritmo, apresentando 100% de acerto em encontrar a aptidão máxima. Isso ocorreu tanto para baixas taxas de mutação 1 (*Merge*) (gráficos (c), (d) e (e)), como para baixas taxas de mutação 3 (*Move*) (gráficos (f), (g) e (h)).

A figura 6.17 apresenta os valores de aptidão encontrados para os testes combinados usando o operador de cruzamento Ponto Único de Grupo, os três operadores de mutação, com o método faixa percentual.

Figura 6.17 – Experimentos combinados, número de grupos variável, método faixa percentual: cruzamento tipo Ponto Único de Grupo e mutações 1, 2 e 3

As mesmas análises feitas para o operador Recombinação com Correção Aleatória são aplicadas ao operador Ponto Único de Grupo. Para as respectivas configurações de parâmetros das figuras 6.16 e 6.17, o comportamento de ambos os operadores, quando se usa o método faixa percentual, é muito semelhante.

Para as configurações mostradas nos gráficos (a) e (b), o operador de cruzamento Ponto Único de Grupo apresentou ligeiro aumento do número de soluções geradas, comparado ao operador Recombinação com Correção Aleatória, apesar do mesmo padrão de comportamento. Para as demais configurações houve praticamente 100% de acerto em encontrar a aptidão máxima.

Para o método faixa percentual foram experimentadas 8 configurações de parâmetros. Em 6 delas, para ambos os operadores de cruzamento, o AGA encontrou a solução máxima em praticamente 100% das rodadas de simulação. Portanto, o método faixa percentual se demonstrou eficaz na solução de problemas com número de grupos variável para o conjunto de peças usado (tabela 5.6).

6.5.3 Comentários: número de grupos variável

Para o conjunto de dados usado (tabela 5.6), os experimentos mostraram a convergência do AGA com ambos os operadores de cruzamento, Recombinação com Correção Aleatória e Ponto Único de Grupo, e para os dois métodos de geração da nova população. Para o método probabilístico, os melhores resultados foram obtidos com o operador Ponto Único de Grupo, com 50% de taxa de cruzamento. Para o método faixa percentual, o comportamento de ambos os operadores foi muito semelhante.

Dessa forma, a análise da aplicação do AGA ao conjunto de peças usado (tabela 5.6) para geração de módulos básicos de arranjo físico em problemas de agrupamento automático permite a recomendação do uso das taxas dos operadores com os seguintes valores:

- Método probabilístico: P_c (Cruzamento): 50%; P_{m_1} (*Merge*): 5% ou 10%; P_{m_2} (*Break*): 50%; P_{m_3} (*Move*): 5% ou 10%.
- Método faixa percentual: P_c (Cruzamento): 30%, 35% ou 50%; P_{m_1} (*Merge*): 5% ou 20%; P_{m_2} (*Break*): 40 ou 50%; P_{m_3} (*Move*): 5% ou 15%. Vale lembrar que para o método percentual a soma das taxas dos operadores deve ser 100%.

Nos experimentos apresentados nas figura 6.13 a 6.17 foram aplicados todos os operadores, de cruzamento, mutação 1 (*Merge*), mutação 2 (*Break*) e mutação 3 (*Move*). Esses experimentos comprovaram que o uso dos operadores de mutação *Merge* e *Break* são determinantes para a solução de problemas com número de grupos variável. Esses operadores são responsáveis por alterar o número de grupos, o tamanho e a conseqüente formação dos grupos. Dessa forma é que se faz a busca pelo melhor número e formação de grupos que maximiza a função de aptidão. Portanto, com a aplicação de todos os operadores genéticos, os experimentos demonstraram a eficácia do algoritmo genético de agrupamento (AGA) na resolução de problemas de agrupamento automático. O AGA encontrou sozinho, o melhor número e formação de módulos para o conjunto de dados.

7 ARRANJO FÍSICO FINAL

7.1 Considerações iniciais

As soluções apresentadas no capítulo 6 pelo algoritmo genético de agrupamento (AGA) fornecem os módulos básicos de arranjo físico. A partir do módulo básico é gerado o arranjo físico final, conforme etapas 5 a 8 do procedimento para geração do arranjo físico modular apresentado no item 2.3.5.2 do capítulo 2.

Neste capítulo é apresentado o arranjo físico final tanto para a melhor solução com número de grupos fixo como para a melhor solução com número de grupos variável.

Por questões didáticas são re-apresentados o conjunto de dados de entrada e as subsequências comuns únicas. O conjunto de dados de entrada composto por 19 peças foi apresentado na figura 5.1 do capítulo 5 e é mostrado na tabela 7.1.

Tabela 7.1 – Sequência de operações e tempos de processamento

Peça (P)	Sequência de Operações	Tempo Total de Processamento (min / lote)
1	1, 4, 8, 9	96-36-36-72
2	1, 4, 7, 4, 8, 7	36-120-20-120-24-20
3	1, 2, 4, 7, 8, 9	96-48-36-120-36-72
4	1, 4, 7, 9	96-36-120-72
5	1, 6, 10, 7, 9	96-72-200-120-72
6	6, 10, 7, 8, 9	36-120-60-24-36
7	6, 4, 8, 9	72-36-48-48
8	3, 5, 2, 6, 4, 8, 9	144-120-48-72-36-48-48
9	3, 5, 6, 4, 8, 9	144-120-72-36-48-48
10	4, 7, 4, 8	120-20-120-24
11	6	72
12	11, 7, 12	192-150-80
13	11, 12	192-60
14	11, 7, 10	288-180-360
15	1, 7, 11, 10, 11, 12	15-70-54-45-54-30
16	1, 7, 11, 10, 11, 12	15-70-54-45-54-30
17	11, 7, 12	192-150-80
18	6, 7, 10	108-180-360
19	12	60

Vakharia e Wemmerlöv apud Huang (2003)

A terceira coluna da tabela 7.1 mostra o tempo total de processamento para cada tipo de peça, ou seja, para o lote completo.

O procedimento para gerar as subsequências comuns únicas foi apresentado no item 5.3 do capítulo 5. As subsequências comuns são re-apresentadas na tabela 7.2.

Tabela 7.2 – Subseqüências comuns únicas

Id.	Subseqüências comuns únicas
S1	1 → 4
S2	1 → 4 → 7
S3	1 → 7 → 11 → 10 → 11 → 12
S4	11 → 12
S5	11 → 7
S6	11 → 7 → 12
S7	3 → 5
S8	4 → 7
S9	4 → 7 → 4 → 8
S10	4 → 8
S11	4 → 8 → 9
S12	6 → 10 → 7
S13	6 → 4 → 8 → 9
S14	7 → 10
S15	7 → 8 → 9
S16	7 → 9
S17	8 → 9

7.2 Arranjo físico final: número de grupos fixo

Os experimentos com número de grupos fixo realizados no capítulo 6 objetivaram avaliar o comportamento no AGA na resolução do problema de K agrupamentos. Neste, o usuário define *a priori* o número de grupos ou módulos desejado. Para K=4, o valor máximo da função de aptidão largura da silhueta média foi 0,4190, com o melhor cromossomo formado pelos grupos: [S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]. Os módulos gerados pelo melhor cromossomo a partir das subseqüências comuns foram mostrados na tabela 5.9 e são re-apresentados na tabela 7.3.

Tabela 7.3 – Módulos gerados: número de grupos fixo

Id. Módulo	Grupo de subsequência comum	Módulo de <i>Layout</i>
M1	S1, S2, S8, S9	$1 \rightarrow 4 \rightarrow 8$ $\updownarrow$ 7
M2	S3, S4, S5, S6	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\searrow \downarrow$ 12
M3	S7, S12, S14, S16	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$ $3 \rightarrow 5$
M4	S10, S11, S13, S15, S17	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\uparrow$ 7

Conforme a etapa 5 do procedimento para geração de arranjo físico modular detalhado no item 2.3.5.2, foi calculado o índice de semelhança entre todos os módulos apresentados na tabela 7.3, para geração de módulos funcionais se necessário. Sejam dois módulos M_i e M_j , o índice de semelhança, já apresentado no item 2.3.5.2, é dado por:

$$\text{Índice de Semelhança} = \frac{n_{ij}}{\min(n_i, n_j)} \quad (7.1)$$

onde n_{ij} é o número de operações comuns a ambos os módulos; n_i e n_j é o número de operações dos módulos M_i e M_j respectivamente. Caso haja operação repetida no módulo, ela é contada apenas uma vez. A tabela 7.4 mostra o índice de semelhança para os módulos apresentados na tabela 7.3.

Tabela 7.4 – Índice de semelhança: número de grupos fixo

Módulos	M1	M2	M3	M4
M1	-			
M2	0,5	-		
M3	0,25	0,4	-	
M4	0,75	0,2	0,6	-

O limitante V (*threshold*) estabelecido para que possa agregar dois módulos foi o mesmo usado por Huang (2003) quando propôs o procedimento para geração de arranjo físico modular (item 2.3.5.2), ou seja, $V = 0,8$. Dessa forma, como não há na tabela 7.4 índice de semelhança maior que o limitante V , não é necessário agregar os módulos básicos de arranjo físico.

De acordo com a etapa 6 do item 2.3.5.2, as seqüências de operações originais (tabela 7.1), são designadas aos módulos de arranjo físico, apresentados na tabela 7.3. Esta designação é feita de acordo com as seguintes guias gerais (IRANI e HUANG, 2000):

- Escolher o módulo que faça o maior número de operações consecutivas da peça, com o objetivo de reduzir a visita da peça a máquinas individuais, fora do seu módulo;
- Escolher o menor número de módulos possível para expressar uma seqüência de operações, visando reduzir o fluxo inter-módulos e o fluxo entre módulo e máquina individual;
- Se possível, escolher módulos com o menor número de máquinas para reduzir a distância intra-módulo.

A tabela 7.5 apresenta as seqüências de operações e sua designação aos módulos.

Tabela 7.5 – Sequência de operações designadas aos módulos: número de grupos fixo

Peça (P)	Sequência de Operações	Sequência aos Módulos
1	1, 4, 8, 9	M1 → M4
2	1, 4, 7, 4, 8, 7	M1 → 7
3	1, 2, 4, 7, 8, 9	M1 → 2 → M4 → 7 → M4
4	1, 4, 7, 9	M1 → M4
5	1, 6, 10, 7, 9	M2 → M3
6	6, 10, 7, 8, 9	M3 → M4
7	6, 4, 8, 9	M4
8	3, 5, 2, 6, 4, 8, 9	M3 → 2 → M4
9	3, 5, 6, 4, 8, 9	M3 → M4
10	4, 7, 4, 8	M1
11	6	M4
12	11, 7, 12	M2
13	11, 12	M2
14	11, 7, 10	M2 → 10
15	1, 7, 11, 10, 11, 12	M2
16	1, 7, 11, 10, 11, 12	M2
17	11, 7, 12	M2
18	6, 7, 10	M3 → 7 → M3
19	12	M2

Na tabela 7.5, a peça 1 é feita nos módulos M1 e M4. A peça 2 é executada no módulo M1 e ainda necessita da máquina 7, chamada operação residual. Dessa forma, conforme a etapa 6 do procedimento para geração de arranjo físico modular (item 2.3.5.2), após a designação das seqüências aos módulos, realiza-se o ajuste dos módulos através da absorção de operações residuais. A tabela 7.6 mostra os módulos de arranjo físico ajustados.

Tabela 7.6 – Módulos *layout* ajustados: número de grupos fixo

Id. Módulo	Módulo de <i>Layout</i> Ajustado
M1	$1 \rightarrow 4 \rightarrow 8$ $\uparrow$ 7
M2	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\downarrow$ 12
M3	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$ $3 \rightarrow 5$
M4	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\downarrow$ 7

A diferença dos módulos da tabela 7.6 e da tabela 7.3 está na inclusão de operações residuais devido ao ajuste nos seguintes módulos:

- M1: inclusão da operação consecutiva $8 \rightarrow 7$
- M2: inclusão da operação consecutiva $7 \rightarrow 10$
- M3: inclusão da operação consecutiva $6 \rightarrow 7$
- M4: inclusão da operação consecutiva $4 \rightarrow 7$

A tabela 7.7 apresenta as seqüências de operações após o ajuste dos módulos. A primeira coluna da tabela identifica a peça. A segunda coluna mostra a seqüência de operações ou máquinas, já que cada operação é feita em uma máquina. A terceira coluna da tabela mostra, além da seqüência de módulos, quais operações são feitas em quais módulos. Por exemplo, para a peça 1, a primeira operação é feita máquina 1 do módulo M1, a segunda, terceira e quarta operações são feitas respectivamente nas máquinas 4, 8 e 9 do módulo M4.

Tabela 7.7 – Seqüência de operações ajustadas aos módulos: número de grupos fixo

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	1, 4, 8, 9	M1 → M4 1 → 4, 8, 9
2	1, 4, 7, 4, 8, 7	M1 1, 4, 7, 4, 8, 7
3	1, 2, 4, 7, 8, 9	M1 → 2 → M4 1 → 2 → 4, 7, 8, 9
4	1, 4, 7, 9	M1 → M4 1, 4, 7 → 9
5	1, 6, 10, 7, 9	M2 → M3 1 → 6, 10, 7, 9
6	6, 10, 7, 8, 9	M3 → M4 6, 10 → 7, 8, 9
7	6, 4, 8, 9	M4 6, 4, 8, 9
8	3, 5, 2, 6, 4, 8, 9	M3 → 2 → M4 3, 5 → 2 → 6, 4, 8, 9
9	3, 5, 6, 4, 8, 9	M3 → M4 3, 5 → 6, 4, 8, 9
10	4, 7, 4, 8	M1 4, 7, 4, 8
11	6	M4 6
12	11, 7, 12	M2 11, 7, 12
13	11, 12	M2 11, 12
14	11, 7, 10	M2 11, 7, 10
15	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12
16	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12
17	11, 7, 12	M2 11, 7, 12
18	6, 7, 10	M3 6, 7, 10
19	12	M2 12

Na tabela 7.7, a máquina 2, necessária às peças 3 e 8 não foi absorvida em módulos, visto que esta máquina não está presente em nenhum deles.

A partir das tabelas 7.6 e 7.7 é feito um diagrama do arranjo físico baseado na seqüência de módulos ajustados do conjunto de peças. A figura 7.1 apresenta este diagrama.

Figura 7.1 – Diagrama de fluxo do arranjo físico modular: número de grupos fixo

O diagrama da figura 7.1 mostra o fluxo inter e intra-módulo. A máquina 2 é chamada máquina residual, pois não pertence a nenhum módulo.

De acordo com a classificação dos módulos de arranjo físico apresentada na figura 2.16 do capítulo 2, os módulos da figura 7.1 podem ser classificados em:

- M1 e M2: módulos funcionais. O módulo M1 é usado pelas peças 1, 2, 3, 4 e 10. Ao processar as peças 2 e 10 há um contra-fluxo da máquina 7 para a máquina 4. No módulo M2, ao processar as peças 15 e 16 há contra-fluxo entre as máquinas 10 e 11.
- M3 e M4: módulos tipo fluxo padronizado. Os módulos podem ser decompostos em *flowline* e *flowline* ramificado. O módulo M3 é usado pelas peças 5, 6, 8, 9 e 18, conforme tabela 7.7. Neste módulo, a partir da máquina 6 o fluxo é ramificado para as máquinas 7 e 10. No módulo M4, a partir da máquina 4 o fluxo é ramificado para as máquinas 7 e 8.

Conforme etapa 7 do procedimento para geração de arranjo físico modular descrito no item 2.3.5.2, deve-se calcular a necessidade de máquinas nos módulos. A equação 2.12

fornece a capacidade disponível por cada tipo de máquina (A_j). Esta equação é re-escrita na equação 7.2.

$$A_j = \Delta U_j \cdot C \quad \text{onde,} \quad (7.2)$$

ΔU_j = porcentagem de utilização da máquina j ;

C = duração do período de produção (horas).

Assim como Huang (2003), considerou-se a mesma porcentagem de utilização para todas as máquinas, $\Delta U_j = 80\%$ e $C = 8$ horas. Assim, $A_j = 6,4$ horas ou 384 minutos.

O número necessário de cada tipo de máquina por módulo de arranjo físico foi apresentado na equação 2.11 e é dado por:

$$N_{jk} = \sum_i \frac{T_{ijk}}{A_j} \quad \text{onde,} \quad (7.3)$$

N_{jk} = número de máquinas do tipo j requerido no módulo k ;

T_{ijk} = capacidade requerida pela peça i na máquina j no módulo k ;

A_j = capacidade disponível por máquina do tipo j .

Para melhor compreender o cálculo de N_{jk} , consideremos o exemplo a seguir:

Pela tabela 7.7 observamos que a máquina 1 do módulo M1 é requerida pelas peças 1, 2, 3 e

4. A tabela 7.1 fornece os tempos de processamento. Assim, a capacidade requerida, T_{ijk} é:

$T_{111} = 96$, $T_{211} = 36$, $T_{311} = 96$, e $T_{411} = 96$. Dessa forma, $\sum_i T_{ijk} = 96 + 36 + 96 + 96 = 324$.

Portanto,

$$N_{11} = 324 / 384 = 0,844$$

A tabela 7.8 mostra o número necessário de cada tipo de máquina por módulo de arranjo físico.

Tabela 7.8 – Número necessário de máquinas: número de grupos fixo

Módulo Máquina	M1	M2	M3	M4	Máquina Residual	Nº Necessário de Máquinas	Nº Existente de Máquinas
1	0,844	0,328	-	-	0	2	2
2	-	-	-	-	0,25	1	1
3	-	-	0,75	-	0	1	1
4	1,344	-	-	0,469	0	3	2
5	-	-	0,625	-	0	1	1
6	-	-	0,562	0,75	0	2	2
7	0,469	1,614	0,781	0,469	0	5	4
8	0,125	-	-	0,625	0	2	1
9	-	-	0,187	1,031*	0	2	2
10	-	1,172	1,771	-	0	4	3
11	-	2,812	-	-	0	3	3
12	-	0,885	-	-	0	1	1
Nº Total de Máquinas						27	23

A penúltima coluna da tabela 7.8 mostra o número necessário por tipo de máquina, considerando uma situação ideal. Por exemplo, para cobrir a carga de máquina, são necessárias uma máquina do tipo 1 no módulo M1 e outra no módulo M2, totalizando 2 máquinas. Portanto, em uma situação ideal é preciso um total de 27 máquinas. O número existente de máquinas apresentado na última coluna foi o mesmo usado por Huang (2003).

Segundo Huang (2003), baixa carga de máquina pode não justificar a compra deste tipo de máquina para o módulo. Ao invés disso, pode ser comprada uma máquina flexível multi-funcional para o módulo, ou o compartilhamento de máquinas por dois módulos adjacentes. Isso ocorre com a máquina 9 no módulo M4, que possui carga excedente de 0,031 (tabela 7.8). Essa carga poderia ser coberta pela máquina 9 do módulo M3. Com isso haveria a necessidade de apenas uma máquina 9 no módulo M4 e outra em M3. Para isso, conforme tabela 7.7, a peça 4, ao invés de ser processada nos módulos M1 e M4, a seria nos módulos M1 (operação 1), M4 (operações 4 e 7) e M3 (operação 9). Essa alteração não afeta a solução

gerada pelo algoritmo genético de agrupamento (AGA) em termos dos módulos básicos de arranjo físico gerados (tabela 7.3), pois não seria criado nenhum fluxo além dos existentes.

Após o cálculo das cargas das máquinas, conforme etapa 8 do procedimento para geração de arranjo físico modular (item 2.3.5.2) deve ser gerado o arranjo físico de bloco. Neste trabalho, um possível arranjo de bloco foi elaborado a partir da aplicação da heurística MST (*maximum spanning tree*), ou árvore de medida máxima (BAASE apud GONÇALVES FILHO, 2004c). A aplicação da heurística MST é apresentada no Apêndice C.

A figura 7.2 mostra um possível arranjo físico de bloco elaborado a partir da proximidade entre máquinas gerada pela aplicação da heurística MST.

Figura 7.2 – Arranjo físico de bloco: número de grupos fixo

Na figura 7.2 a máquina 9 é compartilhada entre os módulos M3 e M4. Para processamento do conjunto das 19 peças apresentado na tabela 7.1 é necessário um total de 27 máquinas. Considerando a disponibilidade de máquinas, apresentada na última coluna da tabela 7.8, quatro máquinas deveriam ser duplicadas, as máquinas 4, 7, 8 e 10. O item 7.2.2 adiante trata de alternativas à duplicação de máquinas.

A solução apresentada por Huang (2003) totalizou 24 máquinas, sendo necessária apenas a duplicação da máquina 10. A necessidade de 3 máquinas a mais geradas pelo AGA ocorreu pela divisão do módulo M4 de Huang (2003) nos módulos M1 e M4 gerados pelo algoritmo genético de agrupamento (AGA), diferença apontada no item 5.9 do capítulo 5. A solução gerada pelo AGA é a que maximiza a função objetivo (aptidão) usada, largura da silhueta média. Como Huang (2003) usou outra função objetivo, a soma das similaridades, não se espera que os resultados sejam os mesmos. Entretanto, foi estabelecida uma base para que os resultados possam ser comparados. A base de comparação foi a distância total percorrida pelo conjunto de peças, apresentada a seguir.

7.2.1 Análise da distância percorrida: número de grupos fixo

As máquinas do arranjo físico da figura 7.2 foram designadas a um *grid* apresentado na figura 7.3.

Figura 7.3 – *Grid* de arranjo físico: número de grupos fixo

O cálculo da distância percorrida por cada uma das 19 peças (tabela 7.1) foi realizado conforme a distância *city-block* ou distância Manhattan, apresentada no item 2.2.4.1 do capítulo 2. As seguintes considerações foram feitas:

- As máquinas foram representadas como pontos no *grid*.
- Todas as máquinas são do mesmo tamanho e as dimensões das mesmas foram desconsideradas.
- O ponto de partida para cálculo da distância de cada peça é a máquina referente à primeira operação.
- As distâncias intra e inter-módulos são unitárias.
- Somente é permitido movimento na horizontal e na vertical e o cálculo da distância é feito da máquina origem à máquina destino.
- No caso de distâncias alternativas para uma peça, foi considerada a distância mais curta.

As seqüências de operações do conjunto das 19 peças foram designadas ao *grid* de arranjo físico da figura 7.3 de acordo com as considerações citadas sobre a distância. A distância percorrida por cada peça é apresentada na tabela 7.9.

Tabela 7.9 – Distância percorrida por cada peça: número de grupos fixo

Peça (P)	Seqüência de Operações	Seqüência aos Módulos	Distância Percorrida
1	1, 4, 8, 9	M1 → M4 1 → 4, 8, 9	5
2	1, 4, 7, 4, 8, 7	M1 1, 4, 7, 4, 8, 7	6
3	1, 2, 4, 7, 8, 9	M1 → 2 → M4 1 → 2 → 4, 7, 8, 9	5
4	1, 4, 7, 9	M1 → M4 → M3 1 → 4, 7 → 9	5
5	1, 6, 10, 7, 9	M2 → M3 1 → 6, 10, 7, 9	4
6	6, 10, 7, 8, 9	M3 → M4 6, 10 → 7, 8, 9	5
7	6, 4, 8, 9	M4 6, 4, 8, 9	4
8	3, 5, 2, 6, 4, 8, 9	M3 → 2 → M4 3, 5 → 2 → 6, 4, 8, 9	8
9	3, 5, 6, 4, 8, 9	M3 → M4 3, 5 → 6, 4, 8, 9	6
10	4, 7, 4, 8	M1 4, 7, 4, 8	4
11	6	M4 6	0
12	11, 7, 12	M2 11, 7, 12	2
13	11, 12	M2 11, 12	1
14	11, 7, 10	M2 11, 7, 10	2
15	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12	7
16	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12	7
17	11, 7, 12	M2 11, 7, 12	2
18	6, 7, 10	M3 6, 7, 10	3
19	12	M2 12	0
Distância Total			76

A tabela 7.9 mostra que a peça 1 percorre 5 unidades de distância, a peça 2, 6 unidades, e assim por diante. Na tabela 7.9 já foi considerada a alteração da sequência de módulos da peça 4, tratada anteriormente.

Portanto, a distância total percorrida pelo conjunto de peças da tabela 7.9 é de 76 unidades.

A seguir, na figura 7.4, é mostrada a solução de arranjo físico modular de bloco apresentada por Huang (2003).

Huang (2003)

Figura 7.4 – Arranjo físico de bloco

As máquinas da figura 7.4 foram designadas a um *grid*, conforme as considerações anteriormente feitas para o cálculo da distância percorrida. A máquina 7 do módulo M3 e as máquinas 11 do módulo M2 foram designadas às posições permitidas do *grid*, visto que não é permitido movimento na diagonal. Entretanto, a designação foi feita de modo que a distância percorrida pelas peças seja a mínima. A figura 7.5 apresenta o *grid* de arranjo físico gerado para a solução apresentada por Huang (2003).

Figura 7.5 – *Grid* de arranjo físico para a solução de Huang (2003)

As seqüências de operações do conjunto das 19 peças foram designadas ao *grid* de arranjo físico da figura 7.5 de acordo com as considerações citadas sobre a distância. A distância percorrida por cada peça para a solução apresentada por Huang (2003) é apresentada na tabela 7.10.

Tabela 7.10 – Distância percorrida por cada peça: solução de Huang (2003)

Peça (P)	Seqüência de Operações	Seqüência aos Módulos	Distância Percorrida
1	1, 4, 8, 9	M4 1, 4, 8, 9	5
2	1, 4, 7, 4, 8, 7	M4 1, 4, 7, 4, 8, 7	8
3	1, 2, 4, 7, 8, 9	M4 → 2 → M4 1 → 2 → 4, 7, 8, 9	9
4	1, 4, 7, 9	M4 1, 4, 7, 9	5
5	1, 6, 10, 7, 9	M4 → M3 1 → 6, 10, 7, 9	6
6	6, 10, 7, 8, 9	M3 → M4 6, 10 → 7, 8, 9	6
7	6, 4, 8, 9	M4 6, 4, 8, 9	5
8	3, 5, 2, 6, 4, 8, 9	M1 → 2 → M4 3, 5 → 2 → 6, 4, 8, 9	9
9	3, 5, 6, 4, 8, 9	M1 → M4 3, 5 → 6, 4, 8, 9	7
10	4, 7, 4, 8	M4 4, 7, 4, 8	5
11	6	M4 6	0
12	11, 7, 12	M2 11, 7, 12	2
13	11, 12	M2 11, 12	1
14	11, 7, 10	M2 11, 7, 10	2
15	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12	9
16	1, 7, 11, 10, 11, 12	M2 1, 7, 11, 10, 11, 12	9
17	11, 7, 12	M2 11, 7, 12	2
18	6, 7, 10	M3 6, 7, 10	3
19	12	M2 12	0
Distância Total			93

Portanto, na solução apresentada por Huang (2003), a distância total percorrida pelo conjunto de peças da tabela 7.10 é de 93 unidades. Comparando-se essa distância com a apresentada na tabela 7.9, observa-se que a não duplicação de máquinas acarreta mais

transporte. Na solução de Huang (2003) são necessárias 24 máquinas com somente uma duplicação de máquina, entretanto, a distância percorrida é de 93 unidades. Na solução apresentada neste trabalho (figura 7.2) são necessárias 27 máquinas sendo quatro máquinas duplicadas, entretanto, a distância percorrida é de 76 unidades. Portanto, se for possível a duplicação de máquinas, há uma redução de 18,28% da distância percorrida.

Ao realizar o projeto do arranjo físico, cabe ao projetista definir pela duplicação ou não das máquinas. O projetista deve analisar o *trade-off* existente entre o custo da duplicação e o ganho pela redução da distância percorrida.

7.2.2 Caminhos para reduzir a duplicação de máquinas: número de grupos fixo

A solução para o arranjo físico modular mostrada na figura 7.2 apresenta uma situação ideal, onde é possível a duplicação de quatro máquinas, conforme necessidade mostrada na tabela 7.8. A duplicação das máquinas simplifica o fluxo de peças pela fábrica. Entretanto, quando a duplicação possui um alto custo, há necessidade de intervenção do projetista com o objetivo de minimizá-la. Para isso, alguns caminhos alternativos devem ser analisados:

- Verificar se o processo de fabricação da peça não pode ser refeito;
- Realizar mudanças no projeto da peça;
- Se o custo da máquina for muito pequeno pode compensar a duplicação. Neste caso pode haver um benefício maior em termos de data de entrega do lote de peças ou de tempo de transporte;
- Girar os módulos de modo que máquinas do mesmo tipo formem seções funcionais. Este é o princípio do arranjo físico celular híbrido (HCL), proposto por Irani e Huang (1998) e também aplicável ao projeto de arranjo físico modular. O HCL foi apresentado no item 2.3.3 do capítulo 2.

- Descaracterizar o módulo de arranjo físico somente para determinadas peças, em benefício da minimização da duplicação. Essa alternativa é analisada com mais detalhes a seguir.

Vamos supor que não se possam duplicar máquinas e, portanto, deve-se trabalhar com o número existente de máquinas mostrado na última coluna da tabela 7.8, ou seja, 23 máquinas. Com isso, considerando a carga de máquina da tabela 7.8, e o arranjo físico de bloco da figura 7.2, as seguintes alterações deveriam ser feitas:

- A carga de 0,344 da máquina 4 no módulo M1 deveria ser transferida para a máquina 4 do módulo M4. O módulo M1 teria apenas uma máquina 4.
- A carga de 0,469 da máquina 7 no módulo M4 seria distribuída para os módulos M1 e M3. No módulo M4 não haveria máquina 7.
- A carga de 0,125 da máquina 8 no módulo M1 deveria ser transferida para a máquina 8 do módulo M4. Não haveria máquina 8 no módulo M1.
- A carga de 0,172 da máquina 10 no módulo M2 deveria ser transferida para a máquina 10 do módulo M3. No módulo M2 haveria apenas uma máquina 10.

O arranjo físico com essas alterações é apresentado na figura 7.6.

Figura 7.6 – Arranjo físico de bloco sem duplicação de máquinas

A figura 7.7 apresenta o arranjo físico de bloco com 23 máquinas, ou seja, sem a duplicação de máquinas. Neste arranjo físico, máquinas do mesmo tipo em módulos diferentes foram posicionadas de modo a formar seções funcionais. No caso de quebra de máquina ou mudança na demanda, a peça pode ser transferida entre as máquinas do mesmo tipo.

Este arranjo físico possui implicações em termos de designação das seqüências aos módulos, de distância percorrida, de carga de máquina e de caracterização do próprio módulo. Essas implicações são analisadas a seguir.

A tabela 7.11 mostra a designação das seqüências aos módulos e a distância percorrida por cada peça para o arranjo físico da figura 7.6.

Tabela 7.11 – Distância percorrida por cada peça: sem duplicação de máquinas

Peça (P)	Seqüência de Operações	Seqüência aos Módulos	Distância Percorrida
1	1, 4, 8, 9	M1 → M4 1 → 4, 8, 9	4
2	1, 4, 7, 4, 8, 7	M1 → M4 → M1 1, 4, 7 → 4, 8 → 7	6
3	1, 2, 4, 7, 8, 9	M1 → 2 → M4 → M1 → M4 1 → 2 → 4 → 7 → 8, 9	8
4	1, 4, 7, 9	M1 → M3 1, 4, 7 → 9	4
5	1, 6, 10, 7, 9	M2 → M3 1 → 6, 10, 7, 9	5
6	6, 10, 7, 8, 9	M3 → M4 6, 10, 7 → 8, 9	6
7	6, 4, 8, 9	M4 6, 4, 8, 9	3
8	3, 5, 2, 6, 4, 8, 9	M3 → 2 → M4 3, 5 → 2 → 6, 4, 8, 9	7
9	3, 5, 6, 4, 8, 9	M3 → M4 3, 5 → 6, 4, 8, 9	5
10	4, 7, 4, 8	M1 → M4 4, 7, 4 → 8	4
11	6	M4 6	0
12	11, 7, 12	M2 11, 7, 12	3
13	11, 12	M2 11, 12	1
14	11, 7, 10	M2 11, 7, 10	2
15	1, 7, 11, 10, 11, 12	M2 → M3 → M2 1, 7, 11 → 10 → 11, 12	8
16	1, 7, 11, 10, 11, 12	M2 → M3 → M2 1, 7, 11 → 10 → 11, 12	8
17	11, 7, 12	M2 11, 7, 12	3
18	6, 7, 10	M3 6, 7, 10	2
19	12	M2 12	0
Distância Total			79

Comparando-se a tabela 7.11 (sem duplicação de máquinas) com a tabela 7.9 (com duplicação de máquinas), observa-se que as peças 2, 3, 4, 6, 10, 15 e 16 tiveram a seqüência

de módulos alterada. A distância total percorrida passou de 76 para 79 unidades, o que representa um aumento de 3,8%.

A tabela 7.12 mostra o número necessário de máquinas por módulo de arranjo físico, para a solução sem a duplicação de máquinas.

Tabela 7.12 – Número necessário de máquinas: número de grupos fixo, sem duplicação

Módulo Máquina	M1	M2	M3	M4	Máquina Residual	Nº Necessário de Máquinas	Nº Existente de Máquinas
1	0,844	0,328	-	-	0	2	2
2	-	-	-	-	0,25	1	1
3	-	-	0,75	-	0	1	1
4	1,031*	-	-	0,781*	0	2	2
5	-	-	0,625	-	0	1	1
6	-	-	0,562	0,75	0	2	2
7	0,781*	1,614	0,937*	0*	0	4	4
8	0*	-	-	0,75*	0	1	1
9	-	-	0,375*	0,844*	0	2	2
10	-	0,937*	2,005*	-	0	3	3
11	-	2,812	-	-	0	3	3
12	-	0,885	-	-	0	1	1
Nº Total de Máquinas						23	23

Na tabela 7.12, o símbolo asterisco indica máquinas que tiveram o número alterado, comparado à solução com a duplicação de máquinas (tabela 7.8). O número de máquinas excedente de 0,031 da máquina 4 no módulo M4 e de 0,005 da máquina 10 no módulo M3 são extremamente pequenos e não justificam a compra de máquina extra. Uma alternativa para essa pequena carga excedente é a execução de hora-extra.

Outra implicação da solução sem a duplicação de máquinas refere-se à caracterização dos módulos de arranjo físico. A tabela 7.13 ilustra as alterações realizadas em termos dos módulos.

Tabela 7.13 – Alteração nos módulos de *layout*: número de grupos fixo

Id. Módulo	Módulo de <i>Layout</i>	Nova estrutura
M1	$1 \rightarrow 4 \rightarrow 8$ $\downarrow$ 7	$1 \rightarrow 4 \leftrightarrow 7$
M2	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\downarrow$ 12	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\downarrow$ 12
M3	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$ $3 \rightarrow 5$	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$ $3 \rightarrow 5$
M4	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$ $\downarrow$ 7	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$

Na tabela 7.13, a segunda coluna mostra os módulos de arranjo físico ajustados, já apresentados na tabela 7.6. A terceira coluna apresenta a nova estrutura após as alterações para evitar a duplicação de máquinas. Na nova estrutura, quanto ao tipo de máquina presente nos módulos, foram alterados os módulos M1 e M4. No módulo M1 foi excluída a máquina 8. As peças 2 e 10 (tabela 7.9) que usavam esta máquina foram desviadas para a máquina 8 do módulo M4 (tabela 7.11). Dessa forma, houve o aumento do fluxo entre os módulos M1 e M4, mais especificamente entre as máquinas 4 e 8 de M4 com a máquina 7 de M1. Por outro lado, para as peças 2 e 10 não houve aumento da distância percorrida.

No módulo M4 foi excluída a máquina 7. As peças 3 e 4 (tabela 7.9) foram desviadas para a máquina 7 de M1 (tabela 7.11) e a peça 6 (tabela 7.9) para a máquina 7 de M3 (tabela 7.11). Essa alteração também contribuiu para o aumento do fluxo entre os módulos M1 e M4 (peça 3), mas por outro lado simplificou o fluxo da peça 4, com a redução do número de módulos visitados. Quanto à peça 6, a alteração foi pequena, com aumento do fluxo dentro de M3 e redução dentro de M4. Em termos de distância, as peças 3 e 6 passaram a percorrer um caminho maior, e a peça 4 teve a distância percorrida reduzida.

Com a exclusão de uma máquina 10 no módulo M2 (figura 7.6) as peças 15 e 16, antes processadas completamente dentro de M2, passaram a ser executadas em M2 e M3, aumentando o fluxo inter-modular.

A exclusão de uma máquina 4 no módulo M1 (figura 7.6) também contribuiu para o aumento do fluxo entre os módulos M1 e M4, já analisada anteriormente.

Portanto, na solução sem a duplicação de máquinas houve aumento do fluxo inter-modular. Devido à exclusão das máquinas 8 e 7 nos módulos M1 e M4 respectivamente pode-se dizer que os mesmos foram descaracterizados. Os módulos M2 e M3 permaneceram com os mesmos tipos de máquinas. Do conjunto de 19 peças, em 5 delas (peças 2, 3, 10, 15 e 16) houve aumento da complexidade do fluxo, com aumento do número de módulos visitados. Isso pode ser observado comparando-se as tabelas 7.9 e 7.11. Dessa forma, a descaracterização dos módulos M1 e M4 fez com que algumas peças saíssem do seu fluxo ideal e provocaram complexidade na programação dos módulos. Cabe ao projetista definir se essa descaracterização é benéfica comparada à não duplicação, visto que são quatro máquinas a menos com um aumento de apenas 3,8% na distância percorrida pelo conjunto de peças.

A solução apresentada neste trabalho sem a duplicação de máquinas ainda possui vantagem em relação à solução de Huang (2003) em termos de distância total percorrida. O arranjo físico da figura 7.6, sem a duplicação de máquinas, possui um total de 23 máquinas com 79 unidades de distância percorrida (tabela 7.11). A solução de Huang (2003) mostrada na figura 7.4 possui um total de 24 máquinas com 93 unidades de distância percorrida (tabela 7.10). Ou seja, este caminho alternativo sem a duplicação de máquinas, comparado à solução de Huang (2003), apresenta redução de 15% na distância total percorrida, e ainda com a redução de uma máquina.

7.3 Arranjo físico final: número de grupos variável

Os experimentos com número de grupos variável realizados no capítulo 6 objetivaram avaliar o comportamento no AGA na resolução do problema de agrupamento automático. Neste, o AGA encontrou sozinho o melhor número e formação de grupos que maximiza a função de aptidão largura da silhueta média. O valor máximo da função de aptidão foi 0,6110, com o melhor cromossomo formado pelos grupos: [S1, S2], [S3, S4], [S5, S6], [S7], [S8, S9], [S10, S11, S13], [S12, S14, S16], [S15, S17]. A tabela 7.14 expressa esses agrupamentos em termos dos módulos gerados.

Tabela 7.14 – Módulos gerados: número de grupos variável

Id. Módulo	Grupo de subsequência comum	Módulo de <i>Layout</i>
M1	S1, S2	$1 \rightarrow 4 \rightarrow 7$
M2	S3, S4	$1 \rightarrow 7 \rightarrow 11 \leftrightarrow 10$ $\downarrow$ 12
M3	S5, S6	$11 \rightarrow 7 \rightarrow 12$
M4	S7	$3 \rightarrow 5$
M5	S8, S9	$4 \leftrightarrow 7$ $\downarrow$ 8
M6	S10, S11, S13	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$
M7	S12, S14, S16	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$
M8	S15, S17	$7 \rightarrow 8 \rightarrow 9$

Conforme a etapa 5 do procedimento para geração de arranjo físico modular detalhado no item 2.3.5.2, foi calculado o índice de semelhança entre os módulos da tabela 7.14. A tabela 7.15 mostra o índice de semelhança.

Tabela 7.15 – Índice de semelhança: número de grupos variável

Módulo	M1	M2	M3	M4	M5	M6	M7	M8
M1	-							
M2	0,67	-						
M3	0,33	1	-					
M4	0	0	0	-				
M5	0,67	0,33	0,33	0	-			
M6	0,33	0	0	0	0,67	-		
M7	0,33	0,5	0,33	0	0,33	0,5	-	
M8	0,33	0,33	0,33	0	0,67	0,67	0,67	-

O limitante V (*threshold*) estabelecido para que possa agregar dois módulos foi o mesmo da seção 7.2, ou seja, $V = 0,8$. Dessa forma, os módulos M2 e M3 da tabela 7.15 foram agregados. A tabela 7.16 apresenta os módulos básicos de *layout* agregados.

Tabela 7.16 – Módulos básicos de *layout* agregados

Id. Módulo	Módulo de <i>Layout</i> Agregado
M1	$1 \rightarrow 4 \rightarrow 7$
M9 = M2 + M3	$1 \rightarrow 7 \leftrightarrow 11 \leftrightarrow 10$ $\searrow \downarrow$ 12
M4	$3 \rightarrow 5$
M5	$4 \leftrightarrow 7$ $\downarrow$ 8
M6	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$
M7	$6 \rightarrow 10 \leftrightarrow 7 \rightarrow 9$
M8	$7 \rightarrow 8 \rightarrow 9$

Na tabela 7.16, os módulos M2 e M3 foram agregados, e o módulo resultante passou a ser denominado M9. Deste ponto do trabalho em diante, a identificação M2 e M3 para número de grupos variável não será mais usada.

De acordo com a etapa 6 do item 2.3.5.2, as seqüências de operações originais (tabela 7.1), são designadas aos módulos de *layout* agregados, apresentados na tabela 7.16. A tabela 7.17 apresenta esta designação.

Tabela 7.17 – Seqüência de operações designadas aos módulos: número de grupos variável

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	1, 4, 8, 9	M1 → M6
2	1, 4, 7, 4, 8, 7	M1 → M5 → 7
3	1, 2, 4, 7, 8, 9	M1 → 2 → M1 → M8
4	1, 4, 7, 9	M1 → M7
5	1, 6, 10, 7, 9	M1 → M7
6	6, 10, 7, 8, 9	M7 → M8
7	6, 4, 8, 9	M6
8	3, 5, 2, 6, 4, 8, 9	M4 → 2 → M6
9	3, 5, 6, 4, 8, 9	M4 → M6
10	4, 7, 4, 8	M5
11	6	M6
12	11, 7, 12	M9
13	11, 12	M9
14	11, 7, 10	M9 → 10
15	1, 7, 11, 10, 11, 12	M9
16	1, 7, 11, 10, 11, 12	M9
17	11, 7, 12	M9
18	6, 7, 10	6 → M7
19	12	M9

A tabela 7.17 mostra que a peça 1 é feita nos módulos M1 e M6. A peça 2 nos módulos M1, M5 e na máquina 7, chamada operação residual. Dessa forma, ainda na etapa 6, após a designação das seqüências aos módulos, realiza-se o ajuste dos módulos através da absorção de operações residuais. A tabela 7.18 mostra os módulos de arranjo físico ajustados.

Tabela 7.18 – Módulos *layout* ajustados: número de grupos variável

Id. Módulo	Módulo de <i>Layout</i> Ajustado
M1	$1 \rightarrow 4 \rightarrow 7$
M9	$ \begin{array}{c} \overline{1 \rightarrow 7} \leftrightarrow 11 \leftrightarrow 10 \\ \searrow \downarrow \\ 12 \end{array} $
M4	$3 \rightarrow 5$
M5	$ \begin{array}{c} 4 \leftrightarrow 7 \\ \downarrow \nearrow \\ 8 \end{array} $
M6	$6 \rightarrow 4 \rightarrow 8 \rightarrow 9$
M7	$ \overline{6 \rightarrow 10} \leftrightarrow 7 \rightarrow 9 $
M8	$7 \rightarrow 8 \rightarrow 9$

A diferença dos módulos da tabela 7.18 e da tabela 7.16 está na inclusão de operações residuais devido ao ajuste nos seguintes módulos:

- M9: inclusão da operação consecutiva $7 \rightarrow 10$
- M5: inclusão da operação consecutiva $8 \rightarrow 7$
- M7: inclusão da operação consecutiva $6 \rightarrow 7$

A tabela 7.19 apresenta as seqüências de operações após o ajuste dos módulos. Por exemplo, para a peça 1, a primeira operação é feita máquina 1 do módulo M1, a segunda, terceira e quarta operações são feitas respectivamente nas máquinas 4, 8 e 9 do módulo M6.

Tabela 7.19 – Seqüência de operações ajustadas aos módulos: número de grupos variável

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	1, 4, 8, 9	M1 $\rightarrow$ M6 1 $\rightarrow$ 4, 8, 9
2	1, 4, 7, 4, 8, 7	M1 $\rightarrow$ M5 1, 4, 7 $\rightarrow$ 4, 8, 7
3	1, 2, 4, 7, 8, 9	M1 $\rightarrow$ 2 $\rightarrow$ M1 $\rightarrow$ M8 1 $\rightarrow$ 2 $\rightarrow$ 4 $\rightarrow$ 7, 8, 9
4	1, 4, 7, 9	M1 $\rightarrow$ M7 1, 4 $\rightarrow$ 7, 9
5	1, 6, 10, 7, 9	M1 $\rightarrow$ M7 1 $\rightarrow$ 6, 10, 7, 9
6	6, 10, 7, 8, 9	M7 $\rightarrow$ M8 6, 10, 7 $\rightarrow$ 8, 9
7	6, 4, 8, 9	M6 6, 4, 8, 9
8	3, 5, 2, 6, 4, 8, 9	M4 $\rightarrow$ 2 $\rightarrow$ M6 3, 5 $\rightarrow$ 2 $\rightarrow$ 6, 4, 8, 9
9	3, 5, 6, 4, 8, 9	M4 $\rightarrow$ M6 3, 5 $\rightarrow$ 6, 4, 8, 9
10	4, 7, 4, 8	M5 4, 7, 4, 8
11	6	M6 6
12	11, 7, 12	M9 11, 7, 12
13	11, 12	M9 11, 12
14	11, 7, 10	M9 11, 7, 10
15	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12
16	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12
17	11, 7, 12	M9 11, 7, 12
18	6, 7, 10	M7 6, 7, 10
19	12	M9 12

Na tabela 7.19, a máquina 2, necessária às peças 3 e 8 não foi absorvida em módulos, visto que esta máquina não está presente em nenhum deles.

A partir das tabelas 7.18 e 7.19 é feito um diagrama do arranjo físico baseado na sequência de módulos ajustados do conjunto de peças. A figura 7.7 apresenta este diagrama.

Figura 7.7 – Diagrama de fluxo do arranjo físico modular: número de grupos variável

O diagrama da figura 7.7 mostra o fluxo inter e intra-módulo. A máquina 2 é a máquina residual.

De acordo com os tipos de módulos de arranjo físico apresentados na figura 2.16 do capítulo 2, os módulos da figura 7.7 podem ser classificados em:

- M1, M4, M6 e M8: módulos do tipo *flowline* (em linha). Todo movimento entre duas operações consecutivas é sempre para frente, em sequência ou saltando máquinas se necessário. Isso pode ser verificado na tabela 7.19. O módulo M1 é usado pelas peças 1, 2, 3, 4 e 5. O módulo M4 é usado pelas peças 8 e 9. O módulo M6 pelas peças 1, 7, 8, 9 e 11,

A penúltima coluna da tabela 7.20 mostra o número necessário de cada tipo de máquina. Por exemplo, para cobrir a carga de máquina, seriam necessárias 2 máquinas do tipo 1 no módulo M1, e uma no módulo M9, totalizando 3 máquinas. Entretanto, segundo Huang (2003), baixa carga de máquina pode não justificar sua compra. Dessa forma, na tabela 7.20, a carga excedente de 0,094 da máquina 1 no módulo M1 poderia ser coberta pela máquina 1 do módulo M9. Com isso haveria a necessidade de apenas uma máquina 1 no módulo M1 e outra em M9. Para isso, conforme tabela 7.19, a peça 1, ao invés de ser processada nos módulos M1 e M6, a seria nos módulos M9 e M6.

Da mesma forma, a carga de 0,052 da máquina 7 no módulo M1 poderia ser executada no módulo M5, visto que este já possui uma máquina 7 com baixa carga. Isso não alteraria a sequência de módulos, pois conforme a tabela 7.19, a única peça que usa a máquina 7 do módulo M1 é a peça 2, operação que poderia ser feita no módulo M5.

A tabela 7.20 mostra ainda que, para processamento do conjunto das 19 peças (tabela 7.1) são necessárias 30 máquinas sendo 7 módulos de arranjo físico e uma máquina residual. Como existem apenas 23 máquinas disponíveis (última coluna da tabela 7.20), 7 máquinas deveriam ser duplicadas, as máquinas 4, 7, 8, 9 e 10, sendo duas cópias das máquinas 7 e 8.

Após o cálculo das cargas das máquinas, conforme etapa 8 do procedimento para geração de arranjo físico modular (item 2.3.5.2) deve ser gerado o arranjo físico de bloco. Neste trabalho, um possível arranjo de bloco foi elaborado a partir da aplicação da heurística MST (*maximum spanning tree*), apresentada no Apêndice C. A figura 7.8 mostra o arranjo físico de bloco.

Figura 7.8 – Arranjo físico de bloco: número de grupos variável

A figura 7.8 mostra o arranjo físico para número de grupos variável, onde são necessárias 30 máquinas e 7 módulos. A máquina 1 é compartilhada entre os módulos M9 e M6, e a máquina 7 entre os módulos M1 e M5. O item 7.3.2 adiante trata de alternativas à duplicação de máquinas. A seguir é feita a análise da distância percorrida para número de grupos variável.

7.3.1 Análise da distância percorrida: número de grupos variável

As máquinas do arranjo físico da figura 7.8 foram designadas a um *grid* apresentado na figura 7.9.

Figura 7.9 – *Grid* de arranjo físico: número de grupos variável

As seqüências de operações do conjunto das 19 peças foram designadas ao *grid* de arranjo físico da figura 7.9, e calculada a distância percorrida. As mesmas considerações sobre a distância feitas na seção 7.2.1 são aplicadas aqui. A distância percorrida por cada peça é apresentada na tabela 7.21.

Tabela 7.21 – Distância percorrida por cada peça: número de grupos variável

Peça (P)	Seqüência de Operações	Seqüência aos Módulos	Distância Percorrida
1	1, 4, 8, 9	M9 → M6 1 → 4, 8, 9	3
2	1, 4, 7, 4, 8, 7	M1 → M5 1, 4 → 7 4, 8, 7	6
3	1, 2, 4, 7, 8, 9	M1 → 2 → M1 → M8 1 → 2 → 4 → 7, 8, 9	7
4	1, 4, 7, 9	M1 → M7 1, 4 → 7, 9	4
5	1, 6, 10, 7, 9	M1 → M7 1 → 6, 10, 7, 9	4
6	6, 10, 7, 8, 9	M7 → M8 6, 10, 7 → 8, 9	4
7	6, 4, 8, 9	M6 6, 4, 8, 9	3
8	3, 5, 2, 6, 4, 8, 9	M4 → 2 → M6 3, 5 → 2 → 6, 4, 8, 9	7
9	3, 5, 6, 4, 8, 9	M4 → M6 3, 5 → 6, 4, 8, 9	5
10	4, 7, 4, 8	M5 4, 7, 4, 8	3
11	6	M6 6	0
12	11, 7, 12	M9 11, 7, 12	2
13	11, 12	M9 11, 12	1
14	11, 7, 10	M9 11, 7, 10	2
15	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12	9
16	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12	9
17	11, 7, 12	M9 11, 7, 12	2
18	6, 7, 10	M7 6, 7, 10	3
19	12	M9 12	0
Distância Total			74

A tabela 7.21 mostra que a peça 1 percorre 3 unidades de distância, a peça 2, 6 unidades, e assim por diante. Na tabela 7.21 já foi considerada a alteração da sequência de módulos das peças 1 e 2, discutida na seção 7.3.

Portanto, para número de grupos variável, a distância total percorrida pelo conjunto de peças (tabela 7.21) é de 74 unidades. Para isso são necessárias 30 máquinas distribuídas em 7 módulos de arranjo físico.

Quando se trabalhou com número de grupos fixo, ou seja, quando no algoritmo genético de agrupamento (AGA) foram definidos *a priori* pelo usuário 4 módulos de arranjo físico, foram necessárias 27 máquinas (figura 7.2) com uma distância total percorrida 76 unidades (tabela 7.9). Portanto, para o problema com número de grupos variável, ou seja, quando no AGA não foi especificado o número de módulos desejado, a duplicação de 3 máquinas pode não se justificar, pois houve redução de apenas 2 unidades de distância.

Isso mostra que, 7 é um número excessivo de módulos, apesar de facilitar a programação da fábrica. Dessa forma, há necessidade da intervenção manual do projetista, com o objetivo de aproximar os módulos de modo a repartir melhor o fluxo. A seção a seguir mostra algumas formas de intervenção.

7.3.2 Caminhos para reduzir a duplicação de máquinas: número de grupos variável

A tabela 7.20 mostra que o número necessário de máquinas em determinados módulos é muito baixo, como por exemplo, o da máquina 7 no módulo M5. Parte da carga da máquina 10 no módulo M9 poderia ser transferida para a máquina 10 do módulo M7, necessitando de apenas uma máquina 10 em M9 (tabela 7.20). Além disso, alguns módulos possuem máquinas em comum, como os módulos M5, M7 e M8 (figura 7.8). A partir dessas observações um novo arranjo físico foi desenvolvido, e é apresentado na figura 7.10.

Figura 7.10 – Arranjo físico: número de grupos variável com alternativas à duplicação

A figura 7.10 ilustra um arranjo físico em que foi reduzido o número de máquinas. Este arranjo possui 25 máquinas, ou seja, 5 máquinas a menos que o arranjo físico anterior da figura 7.8. Houve exclusão de todo o módulo M8 com as máquinas 7, 8 e 9, e ainda da máquina 7 do módulo M5 e de uma máquina 10 do módulo M9.

As máquinas do arranjo físico da figura 7.10 foram dispostas em um *grid* para cálculo da distância percorrida. O *grid* é mostrado na figura 7.11.

Figura 7.11 – *Grid* de arranjo físico: número de grupos variável com alternativas à duplicação

As seqüências de operações do conjunto das 19 peças foram designadas ao *grid* de arranjo físico da figura 7.11, e calculada a distância percorrida. As mesmas considerações sobre a distância feitas na seção 7.2.1 são aplicadas aqui. A distância percorrida por cada peça é apresentada na tabela 7.22.

Tabela 7.22 – Distância percorrida: número de grupos variável com alternativa à duplicação

Peça (P)	Seqüência de Operações	Seqüência aos Módulos	Distância Percorrida
1	1, 4, 8, 9	M9 → M6 1 → 4, 8, 9	3
2	1, 4, 7, 4, 8, 7	M1 → M7 → M5 → M7 1, 4 → 7 → 4, 8 → 7	6
3	1, 2, 4, 7, 8, 9	M1 → 2 → M1 → M7 → M5 → M7 1 → 2 → 4 → 7 → 8 → 9	6
4	1, 4, 7, 9	M1 → M7 1, 4 → 7, 9	4
5	1, 6, 10, 7, 9	M1 → M7 1 → 6, 10, 7, 9	4
6	6, 10, 7, 8, 9	M7 → M5 → M7 6, 10, 7 → 8 → 9	5
7	6, 4, 8, 9	M6 6, 4, 8, 9	3
8	3, 5, 2, 6, 4, 8, 9	M4 → 2 → M6 3, 5 → 2 → 6, 4, 8, 9	7
9	3, 5, 6, 4, 8, 9	M4 → M6 3, 5 → 6, 4, 8, 9	5
10	4, 7, 4, 8	M1 → M7 → M5 4 → 7 → 4, 8	4
11	6	M6 6	0
12	11, 7, 12	M9 11, 7, 12	2
13	11, 12	M9 11, 12	1
14	11, 7, 10	M9 11, 7, 10	2
15	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12	8
16	1, 7, 11, 10, 11, 12	M9 1, 7, 11, 10, 11, 12	8
17	11, 7, 12	M9 11, 7, 12	2
18	6, 7, 10	M7 6, 7, 10	3
19	12	M9 12	0
Distância Total			73

Na tabela 7.22, comparada à tabela 7.21, as peças 2, 3, 6 e 10 tiveram a seqüência de módulos alterada. As peças 2, 3 que anteriormente (tabela 7.21) visitavam 2 módulos,

A tabela 7.23 mostra o número necessário de cada tipo de máquina por módulo. A penúltima coluna da tabela mostra que são necessárias 25 máquinas, apenas 2 máquinas a mais que a quantidade existente.

Comparando-se a tabela 7.23 com a tabela 7.20, as seguintes melhorias foram realizadas na solução do arranjo físico com alternativas à duplicação:

- Melhor balanceamento de carga entre as máquinas do tipo 1 dos módulos M1 e M9. Não há mais carga excedente da máquina 1 em M1.
- Melhor balanceamento de carga entre as máquinas 4 dos módulos M1 e M5.
- Transferência da carga das máquinas do tipo 7 dos módulos M1, M5 e M8 para o módulo M7. Assim, ao invés de 6 máquinas do tipo 7, são necessárias apenas 4 máquinas.
- Transferência da carga da máquina 8 do módulo M8 para o módulo M5, visto que M8 foi excluído. Ao invés de 3 máquinas do tipo 8 são necessárias apenas 2.
- Transferência da carga da máquina 9 do módulo M8 para o módulo M7. O número de máquinas do tipo 9 foi reduzido de 3 para 2.
- Melhor balanceamento da carga das máquinas do tipo 10 entre os módulos M7 e M9, o que possibilitou a redução de 4 para 3 máquinas. A carga excedente de 0,005 em M7 é muito baixa e pode ser coberta, por exemplo, com hora-extra.
- Redução do número total de máquinas necessário.

Esta solução com alternativas à duplicação de máquinas possui implicações quanto à caracterização dos módulos de arranjo físico. A tabela 7.24 ilustra as alterações realizadas em termos dos módulos.

Tabela 7.24 – Alteração nos módulos de *layout*: número de grupos variável

Id. Módulo	Módulo de <i>Layout</i> Ajustado	Nova estrutura
M1	1 → 4 → 7	1 → 4
M9
M4	3 → 5	3 → 5
M5		4 → 8
M6	6 → 4 → 8 → 9	6 → 4 → 8 → 9
M7
M8	7 → 8 → 9	-

Na tabela 7.24, a segunda coluna mostra os módulos de arranjo físico ajustados, já apresentados na tabela 7.18. A terceira coluna apresenta a nova estrutura após as alterações para reduzir a duplicação de máquinas. Na nova estrutura, quanto ao tipo de máquina presente nos módulos, foram alterados os módulos M1, M5 e M8. No módulo M1 foi excluída a máquina 7. A peça 2 (tabela 7.19), única que usava esta máquina, foi desviada para a máquina 7 do módulo M7 (tabela 7.22). Dessa forma, houve o aumento do fluxo entre os módulos M1 e M7. Por outro lado, não houve aumento da distância percorrida para a peça 2.

No módulo M5 também foi excluída a máquina 7. As peças 2 e 10 (tabela 7.21) foram desviadas para a máquina 7 de M7 (tabela 7.22). Essa alteração contribuiu para o aumento do fluxo entre os módulos M5 e M7 e tornou o fluxo das peças 2 e 10 mais complexo. Para a peça 10 houve aumento de 1 unidade de distância.

Com a exclusão de todo o módulo M8, o fluxo das peças 3 e 6 também se tornou mais complexo, com maior número de módulos visitados por estas peças. Essa alteração contribuiu

ainda mais para a intensificação do fluxo entre os módulos M5 e M7. Por outro lado, não houve grande diferença em relação à distância. A distância percorrida pela peça 3 reduziu em 1 unidade, e pela peça 6 aumentou em 1 unidade.

Dessa forma, na solução com alternativas à duplicação de máquinas houve aumento do fluxo inter-modular. Devido à exclusão das máquinas 7 nos módulos M1 e M5 e à exclusão do módulo M8, pode-se dizer que os mesmos foram descaracterizados. Os módulos M4, M6, M7 e M9 permaneceram com os mesmos tipos de máquinas. Do conjunto de 19 peças, em 4 delas (peças 2, 3, 6 e 10) houve aumento da complexidade do fluxo, com maior número de módulos visitados. Isso pode ser observado comparando-se as tabelas 7.21 e 7.22. Portanto, algumas peças saíram do seu fluxo ideal e provocaram complexidade na programação dos módulos. Cabe ao projetista definir se essa descaracterização é benéfica, visto que, para a solução desenvolvida com o objetivo de reduzir a duplicação de máquinas o número de módulos diminuiu de 7 para 6, a necessidade de máquinas de 30 (tabela 7.20) para 25 (tabela 7.23), e a distância total percorrida pelo conjunto de peças de 74 para 73 unidades.

8 ESTUDO DE CASO

8.1 Considerações iniciais

Este capítulo apresenta a aplicação do algoritmo genético de agrupamento (AGA) para geração dos módulos de arranjo físico em um segundo conjunto de dados. Segundo Irani e Huang (2000b) e Huang (2003) este conjunto de dados, composto de 25 peças, foi coletado de uma fábrica de metais estampados. Primeiramente será analisada a resposta do AGA para número de módulos ou grupos fixo, ou seja, especificados *a priori* pelo usuário. Os módulos gerados pelo AGA serão comparados aos módulos apresentados por Huang (2003). Em seguida será analisada a resposta do AGA para número de grupos variável, ou seja, quando o usuário não especifica o número de módulos desejado e o AGA deve encontrar o melhor número e formação de módulos que maximiza a função de aptidão largura da silhueta média. Entretanto, devido à indisponibilidade dos tempos de processamento das operações, não será gerado o arranjo físico detalhado da fábrica. A tabela 8.1 mostra o conjunto de dados.

Tabela 8.1 – Seqüência de operações

Peça (P)	Seqüência de Operações
1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
2	1, 2, 3, 11, 4, 8, 10
3	12, 2, 13, 3, 2, 9, 10
4	12, 2, 6, 3, 10
5	12, 6, 2, 3, 2, 4, 10
6	1, 2, 8, 9, 2, 4, 10
7	2, 3, 5, 4, 6, 7, 6, 7, 10
8	2, 3, 5, 4, 6, 10
9	1, 2, 14, 4, 5, 6, 9, 10
10	1, 2, 3, 4, 5, 6, 7, 10
11	12, 2, 3, 9, 10
12	1, 2, 13, 3, 6, 5, 9, 10
13	1, 2, 3, 5, 4, 8, 6, 8, 10
14	12, 2, 3, 5, 6, 2, 10
15	1, 2, 3, 4, 5, 8, 6, 5, 7, 10
16	1, 2, 3, 4, 5, 8, 6, 5, 7, 10
17	12, 2, 3, 10
18	1, 2, 3, 5, 6, 10
19	12, 2, 3, 5, 6, 9, 10
20	12, 2, 3, 8, 10
21	1, 2, 3, 4, 5, 6, 7, 5, 10
22	1, 2, 5, 6, 4, 9, 10
23	12, 2, 10
24	12, 2, 3, 10
25	12, 2, 3, 5, 4, 6, 9, 10

Huang (2003)

Na tabela 8.1 mostra a seqüência de operações de cada peça. A peça 1 é feita nas máquinas 1 a 10, a peça 2, nas máquinas 1, 2, 3, 11, 4, 8 e 10, e assim por diante. O procedimento para gerar as subsequências comuns únicas foi apresentado no item 5.3 do capítulo 5. As subsequências comuns são apresentadas na tabela 8.2.

Tabela 8.2 – Subseqüências comuns únicas

Id.	Subseqüências comuns únicas
S1	$8 \rightarrow 9$
S2	$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$
S3	$4 \rightarrow 8$
S4	$8 \rightarrow 10$
S5	$3 \rightarrow 2$
S6	$2 \rightarrow 13 \rightarrow 3$
S7	$3 \rightarrow 10$
S8	$2 \rightarrow 4 \rightarrow 10$
S9	$6 \rightarrow 2$
S10	$2 \rightarrow 3 \rightarrow 5 \rightarrow 4 \rightarrow 6$
S11	$6 \rightarrow 7 \rightarrow 10$
S12	$6 \rightarrow 10$
S13	$5 \rightarrow 6 \rightarrow 9 \rightarrow 10$
S14	$6 \rightarrow 5$
S15	$8 \rightarrow 6$
S16	$1 \rightarrow 2 \rightarrow 3 \rightarrow 5$
S17	$12 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6$
S18	$2 \rightarrow 10$

A partir das subseqüências comuns únicas, foi gerada no algoritmo genético de agrupamento (AGA) a matriz de similaridade. Esta matriz foi calculada a partir da aplicação do coeficiente *Merger* sobre as subseqüências comuns. O coeficiente foi detalhado na seção 2.3.5.1. A matriz é apresentada na figura 8.1.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18
S1	1,00	0,00	0,50	0,50	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,40	0,00	0,50	0,00	0,00	0,00
S2		1,00	0,40	0,00	0,40	0,60	0,40	0,50	0,40	0,70	0,60	0,40	0,40	0,40	0,40	0,90	0,70	0,40
S3			1,00	0,50	0,00	0,00	0,00	0,40	0,00	0,40	0,00	0,00	0,00	0,00	0,50	0,00	0,00	0,00
S4				1,00	0,00	0,00	0,50	0,40	0,00	0,00	0,40	0,50	0,40	0,00	0,50	0,00	0,00	0,50
S5					1,00	0,40	0,50	0,40	0,50	0,40	0,00	0,00	0,00	0,00	0,00	0,40	0,40	0,50
S6						1,00	0,40	0,30	0,40	0,60	0,00	0,00	0,00	0,00	0,00	0,60	0,60	0,40
S7							1,00	0,40	0,00	0,40	0,40	0,50	0,40	0,00	0,00	0,40	0,40	0,50
S8								1,00	0,40	0,50	0,30	0,40	0,30	0,00	0,00	0,30	0,30	0,90
S9									1,00	0,40	0,40	0,50	0,40	0,50	0,50	0,40	0,40	0,50
S10										1,00	0,30	0,40	0,40	0,40	0,40	0,70	0,80	0,40
S11											1,00	0,90	0,60	0,40	0,40	0,00	0,30	0,40
S12												1,00	0,90	0,50	0,50	0,00	0,40	0,50
S13													1,00	0,40	0,40	0,20	0,50	0,40
S14														1,00	0,50	0,40	0,40	0,00
S15															1,00	0,00	0,40	0,00
S16																1,00	0,70	0,40
S17																	1,00	0,40
S18																		1,00

Figura 8.1 – Matriz de similaridade

Na figura 8.1, a similaridade entre as subsequências S1 e S2 é zero, entre S1 e S3 é 0,50, e assim por diante.

8.2 Número de grupos fixo

O número de grupos foi especificado em 4, para que se pudesse comparar os módulos gerados pelo AGA com os apresentados Huang (2003).

Dessa forma, após gerar a matriz de similaridade foram definidos os parâmetros do algoritmo genético. A figura 8.2 mostra a tela “Parâmetros”.

Figura 8.2 – Parâmetros do AGA

Os parâmetros do algoritmo genético foram: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005; número de grupos: 4; codificação do cromossomo: R3 (aleatória); método de geração da nova população: probabilístico; taxa de cruzamento: 0,50; taxa de mutação 3: 0,50; função de aptidão: largura da silhueta média; método de seleção: roleta.

Quando o número de grupos é fixo não são aplicados os operadores de mutação 1 e 2. Os demais parâmetros foram os mesmos dos experimentos que obtiveram os melhores resultados para número de grupos fixo realizados no capítulo 6.

Após mandar executar o algoritmo, pode-se acompanhar na tela “Estatísticas”, a evolução dos valores de aptidão para o melhor e o pior indivíduo, e a evolução da aptidão média da população (figura 8.3).

Figura 8.3 – Estatísticas: valores de aptidão

Além do gráfico de valores da aptidão, a tela “Estatísticas” mostra o número de vezes em que os operadores foram aplicados, e o número e o percentual de sucesso de aplicação de cada operador. Por exemplo, a figura 8.3 mostra que foram aplicados 24638 cruzamentos. Este número equivale aproximadamente aos 50% de probabilidade da taxa de cruzamento, uma vez que o tamanho da população é de 200 indivíduos, o critério para término de 500 gerações e os cromossomos para cruzamento são tomados aos pares. Desse total (24638), 6826 (27,7%) cruzamentos foram realizados, ou seja, resultaram em indivíduos diferentes dos pais. Para a mutação 3, esse índice foi de 75,1%. As estatísticas mostram ainda que a massa da população convergiu para o melhor cromossomo (106 indivíduos).

A população final, ordenada em modo decrescente de aptidão, ou seja, com melhor indivíduo na primeira posição, é mostrada na figura 8.4.

	K	Aptidão	Grupos
1	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
2	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
3	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
4	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
5	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
6	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
7	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
8	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
9	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
10	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
11	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
12	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
13	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
14	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
15	4	0,33917	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14]]

Executar Salvar

500/500 00:00:02

Figura 8.4 – População final

A figura 8.4 mostra que para quatro grupos ($K = 4$), o valor da aptidão do melhor indivíduo foi 0,33917. O melhor cromossomo é formado pelos grupos: {S1, S3, S4, S15}, {S2, S6, S10, S16, S17}, {S5, S7, S8, S18} e {S9, S11, S12, S13, S14}.

A tabela 8.3 mostra os módulos de arranjo físico gerados pelo melhor cromossomo a partir das subsequências comuns (tabela 8.2).

Tabela 8.3 – Módulos básicos: número de grupos fixo

Id. Módulo	Grupo de subsequência comum	Módulo Básico de <i>Layout</i>
M1	S1, S3, S4, S15	<pre> graph LR 4 --> 8 8 --> 6 8 --> 9 8 --> 10 </pre>
M2	S2, S6, S10, S16, S17	<pre> graph LR 1 --> 2 2 --> 13 13 --> 3 3 --> 4 4 --> 5 5 --> 6 6 --> 7 12 --> 2 4 <--> 5 3 <--> 4 </pre>
M3	S5, S7, S8, S18	<pre> graph LR 3 --> 2 2 --> 4 4 --> 10 3 <--> 10 2 <--> 4 </pre>
M4	S9, S11, S12, S13, S14	<pre> graph LR 5 --> 6 6 --> 7 7 --> 10 6 --> 9 9 --> 10 2 --> 6 6 --> 10 5 <--> 6 6 <--> 10 </pre>

Na tabela 8.3, o módulo M1, formado pelas subsequências S1, S3, S4 e S15, é composto pelas máquinas 4, 6, 8, 9 e 10.

Após gerar os módulos básicos de arranjo físico, foi calculado o índice de semelhança entre todos os pares de módulos, mostrado na tabela 8.4.

Tabela 8.4 – Índice de semelhança: número de grupos fixo

Módulos	M1	M2	M3	M4
M1	-			
M2	0,4	-		
M3	0,5	0,75	-	
M4	0,6	0,67	0,5	-

O limitante V (*threshold*) estabelecido para que possa agregar dois módulos foi o mesmo usado por Huang (2003), $V = 0,7$. Dessa forma, pela tabela 8.4, devem ser agregados os módulos M2 e M3. A tabela 8.5 mostra os módulos de arranjo físico agregados.

Tabela 8.5 – Módulos agregados: número de grupos fixo

Id. Módulo	Módulo <i>Layout</i> Agregado
M1
M5 = M2 + M3
M4

Na tabela 8.5 os módulos M2 e M3 foram agregados, e o módulo resultante foi identificado por M5.

A tabela 8.6 apresenta as seqüências de operações e sua designação aos módulos agregados (tabela 8.5).

Tabela 8.6 – Sequência de operações designadas aos módulos: número de grupos fixo

Peça (P)	Sequência de Operações	Sequência aos Módulos
1	(1, 2, 3, 4, 5, 6, 7), (8, 9), 10	M5 → M1 → 10
2	(1, 2, 3), 11, (4, 8, 10)	M5 → 11 → M1
3	(12, 2, 13, 3, 2), (9, 10)	M5 → M4
4	(12, 2), 6, (3, 10)	M5 → 6 → M5
5	12, (6, 2), (3, 2, 4, 10)	12 → M4 → M5
6	(1, 2), (8, 9), (2, 4, 10)	M5 → M1 → M5
7	(2, 3, 5, 4, 6, 7), (6, 7, 10)	M5 → M4
8	(2, 3, 5, 4, 6), 10	M5 → 10
9	(1, 2), 14, (4, 5), (6, 9, 10)	M5 → 14 → M5 → M4
10	(1, 2, 3, 4, 5, 6, 7), 10	M5 → 10
11	(12, 2, 3), (9, 10)	M5 → M4
12	(1, 2, 13, 3), (6, 5), (9, 10)	M5 → M4 → M4
13	(1, 2, 3, 5, 4), (8, 6), (8, 10)	M5 → M1 → M1
14	(12, 2, 3, 5, 6), (2, 10)	M5 → M5
15	(1, 2, 3, 4, 5), (8, 6), 5, (7, 10)	M5 → M1 → 5 → M4
16	(1, 2, 3, 4, 5), (8, 6), 5, (7, 10)	M5 → M1 → 5 → M4
17	(12, 2, 3, 10)	M5
18	(1, 2, 3, 5, 6), 10	M5 → 10
19	(12, 2, 3, 5, 6), (9, 10)	M5 → M4
20	(12, 2, 3), (8, 10)	M5 → M1
21	(1, 2, 3, 4, 5, 6, 7), 5, 10	M5 → 5 → 10
22	(1, 2), (5, 6), 4, (9, 10)	M5 → M5 → 4 → M4
23	(12, 2, 10)	M5
24	(12, 2, 3, 10)	M5
25	(12, 2, 3, 5, 4, 6), (9, 10)	M5 → M4

A tabela 8.6 mostra que para a peça 1, as operações 1 a 7 são feitas no módulo M5, as operações 8 e 9 no módulo M1 e a última operação na máquina 10. Esta operação é chamada residual. Dessa forma, após a designação das seqüências aos módulos, realiza-se o ajuste dos módulos através da absorção de operações residuais. A tabela 8.7 mostra os módulos de arranjo físico ajustados.

Tabela 8.7 – Módulos *layout* ajustados: número de grupos fixo

Id. Módulo	Módulo de <i>Layout</i> Ajustado
M1
M5 = M2 + M3
M4

A diferença dos módulos da tabela 8.7 e da tabela 8.5 está na inclusão de operações residuais devido ao ajuste nos seguintes módulos:

- M1: inclusão das operações consecutivas $9 \rightarrow 10$ e $6 \rightarrow 8$;
- M5: inclusão das operações consecutivas $2 \rightarrow 6$, $6 \rightarrow 3$, $6 \rightarrow 10$, $7 \rightarrow 10$, $6 \rightarrow 2$, $7 \rightarrow 5$, $5 \rightarrow 10$, $2 \rightarrow 5$ e $6 \rightarrow 4$;
- M4: inclusão das operações consecutivas $5 \rightarrow 9$ e $5 \rightarrow 7$.

A tabela 8.8 apresenta as seqüências de operações após o ajuste dos módulos. Por exemplo, para a peça 1, da primeira à sétima operação são feitas no módulo M5, e da oitava à décima, no módulo M1.

Tabela 8.8 – Seqüência de operações ajustadas aos módulos: número de grupos fixo

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	(1, 2, 3, 4, 5, 6, 7), (8, 9, 10)	M5 → M1
2	(1, 2, 3), 11, (4, 8, 10)	M5 → 11 → M1
3	(12, 2, 13, 3, 2), (9, 10)	M5 → M4
4	(12, 2, 6, 3, 10)	M5
5	(12), (6, 2), (3, 2, 4, 10)	M5 → M4 → M5
6	(1, 2), (8, 9), (2, 4, 10)	M5 → M1 → M5
7	(2, 3, 5, 4, 6, 7), (6, 7, 10)	M5 → M4
8	(2, 3, 5, 4, 6, 10)	M5
9	(1, 2), 14, (4, 5), (6, 9, 10)	M5 → 14 → M5 → M4
10	(1, 2, 3, 4, 5, 6, 7, 10)	M5
11	(12, 2, 3), (9, 10)	M5 → M4
12	(1, 2, 13, 3), (6, 5, 9, 10)	M5 → M4
13	(1, 2, 3, 5, 4), (8, 6, 8, 10)	M5 → M1
14	(12, 2, 3, 5, 6, 2, 10)	M5
15	(1, 2, 3, 4, 5), (8, 6), (5, 7, 10)	M5 → M1 → M4
16	(1, 2, 3, 4, 5), (8, 6), (5, 7, 10)	M5 → M1 → M4
17	(12, 2, 3, 10)	M5
18	(1, 2, 3, 5, 6, 10)	M5
19	(12, 2, 3, 5, 6), (9, 10)	M5 → M4
20	(12, 2, 3), (8, 10)	M5 → M1
21	(1, 2, 3, 4, 5, 6, 7, 5, 10)	M5
22	(1, 2, 5, 6, 4), (9, 10)	M5 → M4
23	(12, 2, 10)	M5
24	(12, 2, 3, 10)	M5
25	(12, 2, 3, 5, 4, 6), (9, 10)	M5 → M4

Na tabela 8.8, a máquina 11, necessária à peça 2, e a máquina 14, à peça 9, não foram absorvidas, já que estas máquinas não estão presentes em nenhum módulo.

A partir das tabelas 8.7 e 8.8 é feito um diagrama do arranjo físico baseado na seqüência de módulos ajustados do conjunto de peças. A figura 8.5 apresenta este diagrama.

Figura 8.5 – Diagrama de fluxo do arranjo físico modular: número de grupos fixo

O diagrama da figura 8.5 mostra o fluxo inter e intra-módulo. As máquinas 11 e 14 são máquinas residuais.

De acordo com os tipos de módulos de arranjo físico apresentados na figura 2.16 do capítulo 2, os módulos da figura 8.5 podem ser classificados em:

- M1 e M5: módulos funcionais. O módulo M1 é usado pelas peças 1, 2, 6, 13, 15, 16 e 20 (tabela 8.7). Para a peça 13 há contra-fluxo entre as máquinas 8 e 6. O módulo M5 é usado por todas as peças. Há contra-fluxo entre várias máquinas, como por exemplo entre as máquinas 2 e 3, e entre 4 e 5.

- M4: módulo fluxo padronizado. O módulo pode ser decomposto em *flowline* e *flowline* ramificado. Este módulo é usado pelas peças 3, 5, 7, 9, 11, 12, 15, 16, 19, 22 e 25 (tabela 8.8). Neste módulo, a partir da máquina 6 o fluxo é ramificado para as máquinas 5, 7 e 9.

8.2.1 Comparação dos módulos de arranjo físico

A solução de Huang (2003), que usou procedimento baseado em análise de agrupamentos (*clusters analysis*) para geração dos módulos de arranjo físico, é apresentada na tabela 8.9.

Tabela 8.9 – Módulos básicos de arranjo físico

Id. Módulo	Grupo de subsequência comum	Módulo Básico de <i>Layout</i>
M1	S2, S6, S10, S16, S17	
M2	S5, S7, S8, S18	
M3	S9, S11, S12, S13, S14, S15	
M4	S1, S3, S4	

Huang (2003)

Comparando-se os módulos gerados pelo AGA (tabela 8.3) com os gerados por Huang (tabela 8.9), as seguintes análises podem ser feitas:

- Os módulos M2 e M3 do AGA são idênticos aos módulos M1 e M2 de Huang (2003) respectivamente.
- O módulo M1 do AGA difere do módulo M4 de Huang (2003) apenas pela inclusão no primeiro da subsequência S15 ($8 \rightarrow 6$). Esta subsequência possui em comum com as demais subsequências do seu módulo (S1, S3 e S4) a máquina 8, o que justifica essa inclusão pelo AGA.
- O módulo M4 do AGA equivale ao módulo M3 de Huang (2003) a menos da subsequência S15.

A solução apresentada por Huang (2003) foi encontrada pelo AGA, como mostra a figura 8.6.

	K	Aptidão	Grupos
108	4	0,32209	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7, S8, S9, S18], [S11, S12, S13, S14]]
109	4	0,32164	[[S1, S3, S4], [S2, S6, S10, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13, S14, S15]]
110	4	0,31639	[[S1, S3, S4, S15], [S2, S6, S9, S10, S16, S17], [S5, S7, S8, S18], [S11, S12, S13, S14]]
111	4	0,31639	[[S1, S3, S4, S15], [S2, S6, S9, S10, S16, S17], [S5, S7, S8, S18], [S11, S12, S13, S14]]
112	4	0,31639	[[S1, S3, S4, S15], [S2, S6, S9, S10, S16, S17], [S5, S7, S8, S18], [S11, S12, S13, S14]]
113	4	0,31372	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S8, S18], [S7, S9, S11, S12, S13, S14]]
114	4	0,31372	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S8, S18], [S7, S9, S11, S12, S13, S14]]
115	4	0,31372	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S8, S18], [S7, S9, S11, S12, S13, S14]]
116	4	0,31218	[[S1, S3, S4, S15], [S2, S6, S7, S10, S16, S17], [S5, S8, S18], [S9, S11, S12, S13, S14]]
117	4	0,31218	[[S1, S3, S4, S15], [S2, S6, S7, S10, S16, S17], [S5, S8, S18], [S9, S11, S12, S13, S14]]
118	4	0,31218	[[S1, S3, S4, S15], [S2, S6, S7, S10, S16, S17], [S5, S8, S18], [S9, S11, S12, S13, S14]]
119	4	0,31218	[[S1, S3, S4, S15], [S2, S6, S7, S10, S16, S17], [S5, S8, S18], [S9, S11, S12, S13, S14]]
120	4	0,31218	[[S1, S3, S4, S15], [S2, S6, S7, S10, S16, S17], [S5, S8, S18], [S9, S11, S12, S13, S14]]
121	4	0,30398	[[S1, S3, S4, S15], [S2, S6, S10, S14, S16, S17], [S5, S7, S8, S18], [S9, S11, S12, S13]]
122	4	0,30169	[[S1, S3, S15], [S2, S6, S10, S16, S17], [S4, S5, S7, S8, S18], [S9, S11, S12, S13, S14]]
123	4	0,30169	[[S1, S3, S15], [S2, S6, S10, S16, S17], [S4, S5, S7, S8, S18], [S9, S11, S12, S13, S14]]

Figura 8.6 – Solução de Huang (2003): cromossomo na 109ª posição

A figura 8.6 mostra que o AGA encontrou a solução apresentada por Huang (2003), entretanto, para a função de aptidão largura da silhueta média, o cromossomo com esta solução foi classificado na 109ª posição, com função de aptidão igual a 0,32164.

Portanto, para o problema com número de grupos fixo, no qual o usuário especificou em 4 o número desejado de módulos, o AGA agrupou corretamente as subsequências nos módulos, conforme a função de aptidão usada. Os módulos gerados pelo algoritmo genético de agrupamento (tabela 8.3) apresentam praticamente os mesmos tipos de máquinas que os módulos apresentados por Huang (2003) (tabela 8.9).

8.3 Número de grupos variável

O objetivo de aplicação do AGA com número de grupos variável foi saber qual o melhor número e configuração de grupos ou módulos para o qual o valor da função de aptidão largura de silhueta média é máximo.

Como o conjunto de dados é formado por 18 subsequências comuns, o número de grupos mínimo ($K_{\min}$) foi definido em 2 e o máximo ($K_{\max}$) em 17. A função largura da silhueta média opera com no mínimo dois grupos. O número máximo foi definido como o número de subsequências menos um. Se o valor de $K_{\max}$ fosse 18, o valor da silhueta seria zero, com cada subsequência em um grupo.

Os parâmetros do algoritmo genético foram: tamanho da população: 200; número de gerações: 500; taxa de transferência: 0,005; número de grupos: 2 a 17; codificação do cromossomo: R3 (aleatória); método de geração da nova população: probabilístico; taxa de

cruzamento: 0,50; taxa de mutação 1, 2 e 3: 0,05; função de aptidão: largura da silhueta média; método de seleção: roleta.

Os valores dos parâmetros foram definidos tendo como base os experimentos realizados com número de grupos variável no capítulo 6. A figura 8.7 mostra a tela “Parâmetros”.

Figura 8.7 – Parâmetros do AGA: número de grupos variável

Na tela “Estatísticas” é possível acompanhar a evolução dos valores de aptidão para o melhor e o pior indivíduo, e a evolução da aptidão média da população (figura 8.8).

Figura 8.8 – Estatísticas: valores de aptidão, número de grupos variável

Na figura 8.8 as estatísticas mostram que a massa da população convergiu para o melhor cromossomo (170 indivíduos).

A população final, ordenada em modo decrescente de aptidão, ou seja, com melhor indivíduo na primeira posição, é mostrada na figura 8.9.

	K	Aptidão	Grupos
1	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
2	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
3	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
4	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
5	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
6	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
7	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
8	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
9	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
10	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
11	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
12	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
13	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
14	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]
15	6	0,39963	[[S1, S3, S4, S15], [S2, S6, S10, S16, S17], [S5, S7], [S8, S18], [S9, S14], [S11, S12, S13]]

Figura 8.9 – População final: número de grupos variável

A figura 8.9 mostra que o melhor indivíduo apresentou seis grupos ($K = 6$) com valor de aptidão igual a 0,39963. O melhor cromossomo é formado pelos grupos: {S1, S3, S4, S15}, {S2, S6, S10, S16, S17}, {S5, S7}, {S8, S18}, {S9, S14} e {S11, S12, S13}.

A tabela 8.10 mostra os módulos de arranjo físico gerados pelo melhor cromossomo a partir das subsequências comuns (tabela 8.2).

Tabela 8.10 – Módulos básicos: número de grupos variável

Id. Módulo	Grupo de subsequência comum	Módulo Básico de <i>Layout</i>
M1	S1, S3, S4, S15	<pre> graph LR 8 --> 4 8 --> 6 8 --> 9 8 --> 10 </pre>
M2	S2, S6, S10, S16, S17	<pre> graph LR 1 --> 2 2 --> 13 13 --> 3 3 --> 4 4 --> 5 5 --> 6 6 --> 7 12 --> 2 4 <--> 5 </pre>
M3	S5, S7	<pre> graph LR 3 --> 2 3 --> 10 </pre>
M4	S8, S18	<pre> graph LR 2 --> 4 4 --> 10 </pre>
M5	S9, S14	<pre> graph LR 6 --> 2 6 --> 5 </pre>
M6	S11, S12, S13	<pre> graph LR 5 --> 6 6 --> 7 5 --> 10 </pre>

Quando se trabalha com número de grupos variável, o AGA tenta maximizar o valor da função de aptidão largura da silhueta média. Esta função maximiza a homogeneidade interna dos módulos e a heterogeneidade entre módulos distintos. Analisando a matriz de similaridade da figura 8.1, pode-se compreender o motivo da formação de seis módulos (tabela 8.10):

- No módulo M1, a subsequência S1 possui similaridade igual a 0,5 para com S3, S4 e S15. Para com as demais subsequências, a similaridade é zero, exceto para com S13 que é 0,4. Se S13 fosse incluída em M1 o valor da silhueta diminuiria. Dessa forma, foram agrupadas no módulo M1 as subsequências S1, S3, S4 e S15.
- No módulo M4, a similaridade entre S8 e S18 é muito alta (0,9), portanto estas subsequências foram agrupadas.

- No módulo M6, a similaridade entre S11 e S12 e entre S12 e S13 é 0,9. Além disso, a maior similaridade de S13 é para com S11 e vale 0,6. Dessa forma, essas três subsequências foram unidas em um módulo.
- No módulo M3, a similaridade entre S5 e S7 é 0,5. Entre S5 e S9 também é 0,5, entretanto, a similaridade entre S9 e S7 é zero, e, portanto, estas duas subsequências não poderiam estar no mesmo grupo. A similaridade entre S5 e S18 também é 0,5, entretanto S18 foi agrupada com S8, com a qual a similaridade é maior, como mencionado anteriormente. A similaridade de S5 para com as demais subsequências é menor que 0,5, e se fossem unidas, a silhueta diminuiria.
- As subsequências S9 e S14 no módulo M4 foram agrupadas pelo mesmo motivo citado anteriormente para as subsequências do módulo M3.
- No módulo M2, a similaridade entre as subsequências S2, S6, S10, S16 e S17 situa-se entre 0,6 e 0,9. Entre estas e as demais subsequências, a similaridade assume valores menores, e se unidas, também fariam reduzir a silhueta.

Após gerar os módulos básicos de arranjo físico, foi calculado o índice de semelhança entre todos os pares de módulos, mostrado na tabela 8.11.

Tabela 8.11 – Índice de semelhança: número de grupos variável

Módulo	M1	M2	M3	M4	M5	M6
M1	-					
M2	0,4	-				
M3	0,33	0,67	-			
M4	0,67	0,67	0,67	-		
M5	0,33	1	0,33	0,33	-	
M6	0,5	0,75	0,33	0,33	0,67	-

O limitante V (*threshold*) estabelecido para que possa agregar dois módulos foi o mesmo usado para número de grupos fixo, $V=0,7$. Dessa forma, pela tabela 8.11, os módulos M2, M5 e M6 devem ser agregados. A tabela 8.12 mostra os módulos agregados.

Tabela 8.12 – Módulos agregados: número de grupos variável

Id. Módulo	Módulo <i>Layout</i> Agregado
M1	<pre> graph LR 4 --> 8 8 --> 9 9 --> 6 6 --> 10 </pre>
M3	<pre> graph LR 3 --> 2 2 --> 10 </pre>
M4	<pre> graph LR 2 --> 4 4 --> 10 </pre>
M7 = M2 + M5 + M6	<pre> graph LR 1 --> 2 12 --> 2 2 --> 13 13 --> 3 3 --> 4 4 --> 5 5 --> 6 6 --> 7 6 --> 10 10 --> 3 </pre>

Na tabela 8.12, os módulos M2, M5 e M6 foram agregados, e o módulo resultante foi identificado por M7.

A tabela 8.13 apresenta as seqüências de operações e sua designação aos módulos agregados (tabela 8.12).

Tabela 8.13 – Seqüência de operações designadas aos módulos: número de grupos variável

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	(1, 2, 3, 4, 5, 6, 7), (8, 9), 10	M7 → M1 → 10
2	(1, 2, 3), 11, (4, 8, 10)	M7 → 11 → M1
3	(12, 2, 13, 3), 2, 9, 10	M7 → 2 → 9 → 10
4	(12, 2), 6, (3, 10)	M7 → 6 → M3
5	12, (6, 2, 3), (2, 4, 10)	12 → M7 → M4
6	(1, 2), (8, 9), (2, 4, 10)	M7 → M1 → M4
7	(2, 3, 5, 4, 6, 7), (6, 7, 10)	M7 → M7
8	(2, 3, 5, 4, 6, 10)	M7
9	(1, 2), 14, (4, 5, 6), 9, 10	M7 → 14 → M7 → 9 → 10
10	(1, 2, 3, 4, 5, 6, 7, 10)	M7
11	(12, 2, 3), 9, 10	M7 → 9 → 10
12	(1, 2, 13, 3), (6, 5), 9, 10	M7 → M7 → 9 → 10
13	(1, 2, 3, 5, 4), (8, 6), (8, 10)	M7 → M1 → M1
14	(12, 2, 3, 5, 6, 2), 10	M7 → 10
15	(1, 2, 3, 4, 5), (8, 6), 5, (7, 10)	M7 → M1 → 5 → M7
16	(1, 2, 3, 4, 5), (8, 6), 5, (7, 10)	M7 → M1 → 5 → M7
17	(12, 2, 3), 10	M7 → 10
18	(1, 2, 3, 5, 6, 10)	M7
19	(12, 2, 3, 5, 6), 9, 10	M7 → 9 → 10
20	(12, 2, 3), (8, 10)	M7 → M1
21	(1, 2, 3, 4, 5, 6, 7), 5, 10	M7 → 5 → 10
22	(1, 2), (5, 6), 4, 9, 10	M7 → M7 → 4 → 9 → 10
23	(12, 2), 10	M7 → 10
24	(12, 2, 3), 10	M7 → 10
25	(12, 2, 3, 5, 4, 6), 9, 10	M7 → 9 → 10

A tabela 8.13 mostra que a peça 1 é feita nos módulos M7, M1 e na máquina 10, chamada operação residual. Dessa forma, após a designação das seqüências aos módulos, realiza-se o ajuste dos módulos através da absorção de operações residuais. A tabela 8.14 mostra os módulos de arranjo físico ajustados.

Tabela 8.14 – Módulos *layout* ajustados: número de grupos variável

Id. Módulo	Módulo de <i>Layout</i> Ajustado
M1
M3
M4
M7

A diferença dos módulos da tabela 8.14 e da tabela 8.12 está na inclusão de operações residuais devido ao ajuste nos seguintes módulos:

- M1: inclusão das operações consecutivas $9 \rightarrow 10$ e $6 \rightarrow 8$;
- M7: inclusão das operações consecutivas $3 \rightarrow 2$, $2 \rightarrow 6$, $12 \rightarrow 6$, $7 \rightarrow 6$, $3 \rightarrow 6$, $2 \rightarrow 10$, $5 \rightarrow 7$, $3 \rightarrow 10$, $7 \rightarrow 5$, $5 \rightarrow 10$, $2 \rightarrow 5$ e $6 \rightarrow 4$.

A tabela 8.15 apresenta as seqüências de operações após o ajuste dos módulos. Por exemplo, para a peça 1, da primeira à sétima operação são feitas no módulo M7, e da oitava à décima, no módulo M1.

Tabela 8.15 – Seqüência de operações ajustadas aos módulos: número de grupos variável

Peça (P)	Seqüência de Operações	Seqüência aos Módulos
1	(1, 2, 3, 4, 5, 6, 7), (8, 9, 10)	M7 → M1
2	(1, 2, 3), 11, (4, 8, 10)	M7 → 11 → M1
3	(12, 2, 13, 3, 2), (9, 10)	M7 → M1
4	(12, 2, 6), (3, 10)	M7 → M3
5	(12, 6, 2, 3), (2, 4, 10)	M7 → M4
6	(1, 2), (8, 9), (2, 4, 10)	M7 → M1 → M4
7	(2, 3, 5, 4, 6, 7, 6, 7, 10)	M7
8	(2, 3, 5, 4, 6, 10)	M7
9	(1, 2), 14, (4, 5, 6), (9, 10)	M7 → 14 → M7 → M1
10	(1, 2, 3, 4, 5, 6, 7, 10)	M7
11	(12, 2, 3), (9, 10)	M7 → M1
12	(1, 2, 13, 3, 6, 5), (9, 10)	M7 → M1
13	(1, 2, 3, 5, 4), (8, 6, 8, 10)	M7 → M1
14	(12, 2, 3, 5, 6, 2, 10)	M7
15	(1, 2, 3, 4, 5), (8, 6), (5, 7, 10)	M7 → M1 → M7
16	(1, 2, 3, 4, 5), (8, 6), (5, 7, 10)	M7 → M1 → M7
17	(12, 2, 3, 10)	M7
18	(1, 2, 3, 5, 6, 10)	M7
19	(12, 2, 3, 5, 6), (9, 10)	M7 → M1
20	(12, 2, 3), (8, 10)	M7 → M1
21	(1, 2, 3, 4, 5, 6, 7, 5, 10)	M7
22	(1, 2, 5, 6, 4), (9, 10)	M7 → M1
23	(12, 2, 10)	M7
24	(12, 2, 3, 10)	M7
25	(12, 2, 3, 5, 4, 6), (9, 10)	M7 → M1

Na tabela 8.15, a máquina 11, necessária à peça 2, e a máquina 14, à peça 9, não foram absorvidas em módulos, já que estas máquinas não estão presentes em nenhum deles.

A partir das tabelas 8.14 e 8.15 é feito um diagrama do arranjo físico baseado na seqüência de módulos ajustados do conjunto de peças. A figura 8.10 apresenta este diagrama.

Figura 8.10 – Diagrama de fluxo do arranjo físico modular: número de grupos variável

O diagrama da figura 8.10 mostra o fluxo inter e intra-módulo. As máquinas 11 e 14 são máquinas residuais.

De acordo com os tipos de módulos de arranjo físico apresentados na figura 2.16 do capítulo 2, os módulos da figura 8.10 podem ser classificados em:

- M1 e M7: módulos funcionais. O módulo M1 é usado pelas peças 1, 2, 6, 9, 11, 12, 13, 15, 16, 19, 20, 22 e 25 (tabela 8.14). Para a peça 13 há contra-fluxo entre as máquinas 8 e 6. O módulo M7 é usado por todas as peças. Há contra-fluxo entre várias máquinas, como por exemplo entre as máquinas 2 e 3, e entre 4 e 5.
- M3: módulo *flowline* ramificado. A partir da máquina 3 o fluxo é ramificado para as máquinas 2 ou 10.
- M4: módulo *flowline*. O fluxo é sempre para frente, em sequência ou saltando máquinas.

8.3.1 Comentários: número de grupos variável

Analisando a sequência de operações ajustadas aos módulos (tabela 8.15) e o diagrama de fluxo da figura 8.10, percebemos que determinadas peças poderiam seguir roteiros alternativos. O módulo M3 é usado apenas pela peça 4, e todas as máquinas deste módulo já estão presentes no módulo M7. Dessa forma, uma opção seria excluir o módulo M3. Para isso deveria ser adicionada a operação consecutiva $6 \rightarrow 3$ no módulo M7. Com isso a peça 4 seria totalmente processada dentro de M7.

As peças 14 e 23 (tabela 8.15) também poderiam seguir outros roteiros. As duas últimas operações de ambas as peças, nas máquinas 2 e 10, poderiam ser executadas no módulo M4, visto que neste módulo já existe a operação consecutiva $2 \rightarrow 10$.

Esta definição sobre roteiros alternativos cabe ao projetista. Para isso ele deve dispor de dados sobre os tempos de processamento das peças nas máquinas para calcular a carga de máquina e o número necessário de cada tipo de máquina. Com isso é possível definir qual o arranjo físico detalhado mais adequado conforme objetivo a ser otimizado, como por exemplo a minimização da distância percorrida pelo conjunto de peças.

Na aplicação do AGA ao conjunto de peças da tabela 8.1 no problema de número de grupos variável, foram gerados 6 módulos de arranjo físico (tabela 8.10). Portanto, este é o melhor número de módulos que maximiza a função de aptidão largura da silhueta média.

Nos módulos gerados pelo AGA (tabela 8.10) o fluxo é simples, direto, e, portanto, está em sintonia com um dos princípios do arranjo físico modular, a simplificação do fluxo existente na fábrica. A simplificação do fluxo possui um custo, a duplicação de máquinas. O projetista deve definir um ponto de equilíbrio entre ambos conforme as necessidades e objetivos da empresa.

9 CONCLUSÃO

Esta tese apresentou um novo procedimento em projeto de arranjo físico modular. O procedimento usa algoritmo genético de agrupamento para geração de módulos de arranjo físico. O modelo do algoritmo genético foi definido a partir das subsequências de operações comuns a um conjunto de peças. O algoritmo genético foi capaz de identificar os padrões de fluxo existentes na fábrica e gerar módulos de arranjo físico a partir do agrupamento das subsequências.

Uma característica importante do algoritmo genético de agrupamento (AGA) é sua flexibilidade. O usuário pode dirigir seu raciocínio através da seleção de diferentes medidas de similaridade para comparação de sequências de operações, escolha da forma da geração da população inicial, conforme três diferentes tipos de codificações do cromossomo, escolha da função de aptidão para avaliar a qualidade da solução e de diferentes métodos de aplicação dos operadores genéticos.

O AGA permite ao usuário especificar *a priori* o número desejado de módulos (problema de K-agrupamentos), e também trabalhar de forma que o número e formação dos módulos seja uma otimização do problema (problema de agrupamento automático).

O único procedimento encontrado na literatura em projeto de arranjo físico modular, foi proposto por Huang (2003). Neste, os módulos são gerados a partir de análise de agrupamentos (*cluster analysis*). Neste método, o número de módulos é definido *a priori* pelo usuário. A função objetivo usada foi a soma da similaridade entre subsequências. Segundo Hruschka (2001), quando se define o número de grupos, no caso módulos, o algoritmo é rodado várias vezes, para depois escolher a solução que pareça melhor, ou que tenha a interpretação mais significativa.

O uso do AGA para agrupamento das subsequências em módulos substitui a etapa 4 do procedimento para projeto de arranjo físico modular de Huang (2003), descrito no item 2.3.5.2. O princípio de um algoritmo genético é totalmente diferente da análise de agrupamentos (*cluster analysis*). Ele trabalha com a maximização da função objetivo através de um processo de evolução a partir das melhores soluções. O AGA apresenta ainda um diferencial em relação ao trabalho de Huang (2003). Se no AGA fosse usada a função soma da similaridade, tenderia a se formar um agrupamento único, com todas as subsequências, visto que o objetivo é a maximização da função. Ou seja, seria formado apenas um módulo, com todas as máquinas. Portanto, a função soma da similaridade não é indicada para projeto de arranjo físico modular quando se usa um processo de busca por melhores soluções.

Dessa forma, nesta tese foram estudadas e aplicadas outras funções de aptidão. A primeira função estudada foi a soma das similaridades médias, detalhada na seção 4.5.1 do capítulo 4. Quanto maior a similaridade entre duas subsequências, mais operações consecutivas comuns elas possuem, e como cada operação é feita em uma máquina, pode-se dizer que mais semelhantes são as suas máquinas. A média foi calculada pelo número de combinações de subsequências em cada grupo. Assim, todos os grupos formados possuem a mesma base de comparação, ou seja, nenhum grupo é privilegiado por possuir maior número de elementos.

Entretanto, os testes realizados no capítulo 5 demonstraram que a função soma das similaridades médias permite que diferentes tipos de agrupamentos apresentem o mesmo valor da função de aptidão. Este fato ocorreu porque a função soma não considera a separação entre grupos, ou seja, existem várias formas de se agrupar elementos de modo que a soma seja a máxima. Isso provou que a heterogeneidade entre diferentes grupos é uma característica importante para projeto de arranjo físico modular, e a função soma das similaridades médias

não contempla esta característica. Portanto, esta função não é indicada para geração de grupos ou módulos em projeto de arranjo físico modular.

A segunda função de aptidão usada para avaliação das soluções do algoritmo genético foi a largura da silhueta média, descrita na seção 4.5.2 do capítulo 4. Esta função, proposta por Kaufman e Rousseeuw (1990), trabalha com dissimilaridades entre elementos de um conjunto de dados. Nesta tese, com dissimilaridades entre as subsequências de operações. Os testes de validação realizados no capítulo 5 e os experimentos apresentados no capítulo 6 comprovaram a eficácia desta função na formação dos módulos de arranjo físico. Para cada valor da função existe apenas uma formação de grupos. Esta função maximiza a homogeneidade interna de cada grupo, e a heterogeneidade entre grupos distintos. Dessa forma, a largura da silhueta média é indicada para geração de módulos em projeto de arranjo físico modular.

A terceira função de aptidão aplicada foi a taxa do critério da variância, apresentada na seção 4.5.3 do capítulo 4. Esta função, proposta por Calinski e Harabasz (1974), considera o isolamento externo de um subconjunto de dados e a sua homogeneidade interna, através da medida da variabilidade dentro dos grupos e da variabilidade entre os grupos. Esta medida está associada à determinação do número correto de agrupamentos em um conjunto de dados. Portanto, a princípio, seria indicada para geração de módulos de arranjo físico. Entretanto, os testes apresentados no capítulo 5 demonstraram que não houve constância do número de módulos da melhor solução, nem dos agrupamentos formados. A função taxa do critério da variância tende a colocar no mesmo grupo subsequências cujo valor da similaridade sejam próximos, como por exemplo, todas as subsequências com similaridade igual a zero no mesmo grupo. Isso ocorreu devido à função minimizar a variação existente dentro dos grupos. Assim, pode-se concluir que a função taxa do critério da variância não pode ser usada quando

se trabalha com a variável similaridade, e, portanto, não é indicada para geração de grupos ou módulos em projeto de arranjo físico modular.

Neste trabalho foram implementados dois métodos de geração da nova população, probabilístico, como proposto por Goldberg (1986) e por faixa percentual, como proposto por Hruschka (2001). No primeiro método, como o próprio nome indica, a aplicação dos operadores de cruzamento e mutação está associada a uma probabilidade. No segundo, os operadores sempre são aplicados. Os métodos foram descritos no item 4.7 do capítulo 4. Os experimentos realizados no capítulo 6 demonstraram a convergência do AGA para ambos os métodos. Entretanto, o método faixa percentual é mais sensível à qualidade dos operadores de cruzamento e mutação, visto que neste método os operadores sempre serão aplicados.

Nesta tese foram desenvolvidos três tipos de operadores de cruzamento e quatro tipos de operadores de mutação. Estes operadores são detalhados no item 4.6 do capítulo 4. Os primeiros operadores desenvolvidos foram o de cruzamento denominado Recombinação e o operador de mutação *Move Direcionado*. Neste trabalho, o operador Recombinação proposto por Hruschka (2001) foi modificado e adaptado ao problema de agrupamento de subsequências de operações de peças. Neste operador também foi introduzido um mecanismo de correção dos descendentes para evitar cromossomos inválidos. Neste mecanismo, a correção dos descendentes é direcionada, ou seja, quando é preciso criar grupos ou extinguir grupos para evitar os cromossomos inválidos foi estabelecido um critério. Este critério baseia-se na dissimilaridade entre as subsequências. O operador de mutação *Move Direcionado* criado, também estabelece critério para mover uma subsequência de um grupo para outro.

Entretanto, os experimentos apresentados no item 6.2 do capítulo 6 demonstraram que o operador Recombinação permitiu pouca evolução da população e não foi capaz de encontrar a melhor solução. O operador de mutação *Move Direcionado* não foi capaz de produzir diversidade suficiente na população. Isso comprova que o direcionamento dos operadores em

algoritmos genéticos não contribui para o processo de evolução. Dessa forma, surgiu a necessidade do desenvolvimento de novos operadores.

O segundo operador de cruzamento desenvolvido chamado Recombinação com Correção Aleatória difere do primeiro pelo mecanismo correção dos descendentes. Neste, quando é preciso criar ou extinguir grupos para evitar os cromossomos inválidos, a seleção do grupo é feita aleatoriamente. Os experimentos realizados com este operador, apresentados no item 6.3 do capítulo 6, mostraram sua superioridade em relação ao operador Recombinação, devido a evolução da população para valores maiores de aptidão.

O segundo operador de mutação desenvolvido, chamado *Move* Aleatório, é semelhante ao proposto por Cole (1998), com a diferença de que, nesta tese, o grupo selecionado deve possuir no mínimo dois objetos (subseqüências). Os experimentos (item 6.3) mostraram que este operador é altamente eficaz, e, para o problema de K-agrupamentos, em que o número de módulos é pré-estabelecido, este operador consegue, sozinho, quando aplicado com alta probabilidade, encontrar a melhor solução com aptidão máxima.

O terceiro operador de cruzamento desenvolvido, denominado Ponto Único de Grupo, foi inspirado no trabalho de Gonçalves Filho e Tiberti (2006). Neste operador há a preocupação em preservar nos descendentes o mesmo número de grupos dos cromossomos pais. As principais contribuições deste operador que o fazem diferir do operador dos autores *supra* citados são detalhadas no item 4.6.1.3. O desempenho do operador Ponto Único de Grupo foi superior ao operador Recombinação, como demonstram os experimentos apresentados no item 6.3.

Na resolução do problema de K-agrupamentos, ou seja, quando o usuário define o número desejado de grupos ou módulos (K fixo), tanto o operador de cruzamento Ponto Único de Grupo como o operador Recombinação com Correção Aleatória convergiram para a melhor solução e apresentaram padrões semelhantes de solução para cada configuração de

parâmetros usada. Neste tipo de problema, os experimentos mostraram (item 6.4) que não há diferença significativa entre esses dois operadores de cruzamento.

Na resolução do problema de agrupamento automático, ou seja, quando o usuário não define o número desejado de módulos (K variável), tanto o operador Ponto Único de Grupo como o operador Recombinação com Correção Aleatória convergiram para a melhor solução e apresentaram resultados superiores com menores taxas de cruzamento (50% ao invés de 85%). Entretanto, o melhor resultado foi obtido com o operador Ponto Único de Grupo.

Os operadores de mutação *Merge* e *Break* desenvolvidos são usados quando se trabalha com número de grupos ou módulos variável. Os experimentos realizados comprovaram que esses operadores são fundamentais na resolução desses problemas, de agrupamento automático. Esses operadores são responsáveis por alterar a estrutura dos módulos de arranjo físico e com isso buscar pelo melhor número e formação de módulos que maximiza a função objetivo usada, a largura da silhueta média.

O algoritmo genético de agrupamento desenvolvido forneceu os módulos básicos de arranjo físico. Entretanto, como a função de aptidão usada no AGA é diferente da função usada por Huang (2003), foi estabelecida uma base para comparação do arranjo físico final. A comparação foi feita através da distância total percorrida pelo conjunto de peças, definida no capítulo 7.

Para o problema com número de grupos fixo (K -agrupamentos), em que o número de módulos foi pré-especificado em 4, os módulos de arranjo físico gerados pelo AGA totalizaram 27 máquinas com uma distância percorrida de 76 unidades. Na solução apresentada por Huang (2003) são necessárias 24 máquinas, entretanto, a distância percorrida é de 93 unidades. Portanto, a não duplicação de máquinas acarreta mais transporte. Dessa forma, se for possível a duplicação de máquinas, a solução gerada pelo AGA apresenta uma redução de 18,28% da distância percorrida.

No capítulo 7 foram apresentados caminhos para reduzir a duplicação de máquinas. No caminho alternativo, sem a duplicação de máquinas, a distância percorrida pelo conjunto de peças foi de 79 unidades. Ou seja, houve um aumento de 3,8% em relação à solução gerada pelo AGA, mas por outro lado, não foi necessário duplicar máquinas. Ainda assim, o caminho alternativo apresentado, comparado à solução de Huang (2003), apresentou uma redução de 15% na distância total percorrida.

Para o problema com número de grupos variável (agrupamento automático), no qual não foi especificado o número de módulos, o AGA gerou 7 módulos de arranjo físico totalizando 30 máquinas e uma distância percorrida de 74 unidades. Dessa forma, houve a redução de apenas 2 unidades de distância em relação ao problema com número de grupos fixo. Isso mostra que a solução do AGA reduziu a distância percorrida, mas em termos de duplicação de máquinas este ganho pode não se justificar. Com isso, a partir da solução gerada pelo AGA para número de grupos variável, também foram apresentados caminhos para reduzir a duplicação de máquinas. Por outro lado, a solução do AGA para o problema de agrupamento automático fez com que o fluxo de peças se tornasse mais simples e facilitou a programação da fábrica, estando assim em sintonia com o projeto de arranjo físico modular. Portanto, o projetista deve analisar o *trade-off* existente entre o custo da duplicação e o ganho pela redução da distância percorrida, e ainda as implicações quanto ao fluxo de materiais e a programação da fábrica.

9.1 Sugestões para trabalhos futuros

Como uma característica importante do AGA é a possibilidade do usuário dirigir seu próprio raciocínio, uma extensão interessante seria implementar no algoritmo genético outros processos de seleção dos indivíduos, como por exemplo, por torneio e seleção aleatória. Na seleção por torneio, escolhe-se aleatoriamente, com probabilidades iguais (segundo distribuição uniforme), dois indivíduos da população e o com melhor aptidão é selecionado para sofrer reprodução. Na seleção aleatória, um indivíduo da população é selecionado aleatoriamente (segundo distribuição uniforme) para sofrer reprodução. Esses dois processos de seleção poderiam ser comparados com o método da Roleta, já implementado.

Nos experimentos realizados nesta tese foi usado como medida de similaridade o coeficiente *Merger* proposto por Huang (2003), para que se pudessem realizar as comparações entre os dois trabalhos. Uma tarefa importante seria realizar simulações no AGA com os outros dois coeficientes baseados em comparação de seqüência de operações, o coeficiente LCS (*longest common subsequence*) de Askin e Zhou (1998) e o coeficiente baseado na distância de Levenshtein (TAM, 1990). Dessa forma se poderiam realizar comparações entre os três coeficientes para o mesmo conjunto dados.

Uma pesquisa importante seria realizar um estudo de caso prático com dados obtidos diretamente da indústria, de forma que se pudesse comparar um arranjo físico já existente com aquele resultante da aplicação do AGA na geração dos módulos de arranjo físico. Neste estudo prático seria interessante que o conjunto de dados fosse relativamente grande, com por exemplo cinquenta subseqüências comuns únicas. Com um conjunto de dados de tal dimensão a codificação do cromossomo do tipo R2, aplicada ao problema de agrupamento automático, pode ser melhor analisada, pois esta codificação realiza um pré-agrupamento dos dados para

geração da população inicial. Com isso, a codificação R2 pode ser comparada à codificação R3, na qual a população inicial é totalmente aleatória.

O algoritmo genético desenvolvido (AGA) fornece os módulos de arranjo físico. Como continuidade da pesquisa, poderia ser incorporado ao *software* as etapas 5 a 8 do procedimento para projeto de arranjo físico modular descritas no item 2.3.5.2 do capítulo 2. Assim, o arranjo físico final seria gerado no próprio AGA. Nessa extensão do AGA até o arranjo físico final, uma pesquisa interessante seria usar também algoritmos genéticos para gerar o arranjo físico de bloco. Assim, a distância percorrida poderia ser incorporada como elemento da função de aptidão do algoritmo genético de bloco.

Neste trabalho, os arranjos físicos gerados foram analisados e comparados em relação à distância total percorrida. A distância percorrida é uma medida de desempenho estática. Uma pesquisa relevante é a avaliação do arranjo físico final por medidas de desempenho dinâmicas, como por exemplo, a quantidade de estoque em processo, ou o tempo de passagem das peças pelo sistema. Neste caso, a simulação de eventos discretos com uso de medidas dinâmicas é uma ferramenta importante e indicada para análise do desempenho do arranjo físico.

O uso da simulação para avaliar a qualidade do arranjo físico poderia ocorrer de duas formas. A primeira seria aplicar a simulação após o algoritmo genético. Poderiam ser usados *softwares* comerciais de simulação como o Promodel, Arena ou Automod. Nesta aplicação os resultados do AGA seriam enviados para a simulação. Para isso deveria ser realizada a integração entre os dois *softwares*. A segunda forma seria incorporar a simulação, em linguagem de programação, dentro da função de aptidão do algoritmo. Nesta forma, a simulação guia a escolha do melhor arranjo físico.

No procedimento para projeto de arranjo físico modular as rotas são fixas, ou seja, existe somente uma seqüência de operações para cada peça. O uso de roteiros alternativos é uma pesquisa importante que pode contribuir para a melhoria do arranjo físico da fábrica.

Na etapa 5 do procedimento para projeto de arranjo físico modular, descrito na seção 2.3.5.2, é usado um limitante V , para que se agregue dois módulos de arranjo físico. Este limitante é estabelecido pelo projetista. Se o valor de V for pequeno, a tendência é de se agregar os módulos de arranjo físico, com a redução do fluxo inter-modular. Se o valor de V for muito alto, há menos agregação, e conseqüente aumento do fluxo inter-modular. O valor de V influi diretamente no *trade-off* existente entre o custo do fluxo inter-modular e a necessidade de duplicação de máquinas. Dessa forma, uma pesquisa importante é a análise da variação do valor do limitante V e suas conseqüências no arranjo físico final.

REFERÊNCIAS

ADEL EL-BAZ, M. (2004). A genetic algorithm for facility layout problems of different manufacturing environments. **Computers & Industrial Engineering**, v.47, n.2-3, p.233-246.

ASKIN, R.G., ZHOU, M. (1998). Formation of independent flow-line cells based on operation requirements and machine capabilities. **IIE Transactions**, v.30, n.4, p.319-329.

ARENA. (2006). Disponível em: <<http://www.paragon.com.br>>. Acesso em: 16 nov. 2006.

AUTOMOD. (2007). Disponível em:
<http://www.brookssoftware.com/pages/245_automod_overview.cfm>. Acesso em: 10 jan. 2007.

AYTUG, H., KHOUJA, M., VERGARA, F.E. (2003). Use of genetic algorithms to solve production and operations management problems: a review. **International Journal of Production Research**, v.41, n.17, p.3955-4009.

AZADIVAR, F. (1999). Simulation optimization methodologies. **Winter Simulation Conference Proceedings**, Phoenix, v.1, p.93-100.

AZADIVAR, F., TOMPKINS, G. (1999). Simulation optimization with qualitative variables and structural model changes: a genetic algorithm approach. **European Journal of Operational Research**, v.113, p.169-182.

AZADIVAR, F. et al. (2002). Simulation based optimization in fishery management. **Winter Simulation Conference Proceedings**, San Diego, v.1, p.525-531.

AZADIVAR, F., WANG, J. (2000). Facility layout optimization using simulation and genetic algorithms. **International Journal of Production Research**, v.38, n.17, p.4369-4383.

BAKER, R. P., MAROPOULOS, P.G. (2000). Cell design and continuous improvement. **International Journal of Computer Integrated Manufacturing**, v.13, n.6, p.522-532.

BALAKRISHNAN, J., CHENG, C.H. (2000). Genetic search and the dynamic layout problem. **Computers & Operations Research**, v.27, n.6, p.587-593.

BENJAAFAR, S. (2002). Modeling and analysis of congestion in the design of facility layouts. **Management Science**, v.48, n.5, p.679-704.

BENJAAFAR, S., HERAGU, S.S., IRANI, S.A. (2002). Next generation factory layout: research challenges and recent progress. **Interfaces**, v.32, n.6, p.58-76.

BUSSAB, W.O., MIAZAKI, E.S., ANDRADE, D.F. (1990). Introdução à análise de agrupamentos. In: SINAPE - SIMPÓSIO NACIONAL DE PROBABILIDADE E ESTATÍSTICA, 9, São Paulo. **Anais...** São Paulo: ABE – Associação Brasileira de Estatística.

CALINSKI, T., HARABASZ, J. (1974). A dendrite method for cluster analysis. **Communications in Statistics**, v.3, n.1, p.1-27.

CARLATONIO, L.M. DI. (2001). **Novas metodologias para clusterização de dados**. 148f. Dissertação (Mestrado em Ciências em Engenharia Civil) – COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2001.

CARSON, Y., MARIA, A. (1997). Simulation optimization: methods and applications. **Winter Simulation Conference Proceedings**, Atlanta, p.118-126.

CASILLAS, A., GONZÁLEZ DE LENA, M.T., MARTÍNEZ, R. (2003). Document clustering into an unknown number of clusters using a genetic algorithm. In: **Lecture Notes in Computer Science**. Berlin: Springer, 2003. v. 2807/2003, p.43-49.

CERVO, A.L., BERVIAN, P.A. (1996). **Metodologia científica**. São Paulo: Makron Books do Brasil Editora. 4ª ed.

CHAUDHRY, S.S., LUO, W. (2005). Application of genetic algorithms in production and operations management: a review. **International Journal of Production Research**, v.43, n.19, p.4083-4101.

CHWIF, L. (1994). **Uma metodologia para auxiliar o planejamento do arranjo físico em indústrias de manufatura: base para sistemas computadorizados**. Dissertação (Mestrado em Engenharia Mecânica), POLI, Universidade de São Paulo, São Paulo, 1994.

COLE, R. M. (1998). **Clustering with genetic algorithms**. Thesis (Master of Science). Department of Computer Science, University of Western Australia, 1998.

COWGIL, M.C., HARVEY, R.J., WATSON, L.T. (1999). A genetic algorithm approach to cluster analysis. **Computers & Mathematics with Applications**, v.37, n.7, p.99-108.

DAITA, S.T.S, IRANI, S.A., KOTAMRAJU, S. (1999). Algorithms for production flow analysis. **International Journal of Production Research**, v.37, n.11, p.2609-2638.

DE LIT, P., FALKENAUER, E., DELCHAMBRE, A. (2000). Grouping genetic algorithms: an efficient method to solve the cell formation problem. **Mathematics and Computers in Simulation**, v.51, p.257-271.

DÍGRAFO. (2005). Apresenta manual elaborado por PIMENTEL, G. Disponível em: <<http://www.icmsc.sc.usp.br/manuals/sce183/gfdig.html>>. Acesso em: 13 out. 2005.

EVERITT, B.S., LANDAU, S., LEESE, M. (2001). **Cluster analysis**. London: Arnold, Inc.4th ed. 237p.

FACELI, K., CARVALHO, A.C.P.L.F., SOUTO, M.C.P. (2005) **Algoritmos de agrupamento de dados**. São Carlos: ICMC-USP. 45p. (Relatório técnico, RT-SCE-249).

FERNANDES, F.C.F., DALALIO, A.G. (2002). Balanceamento e rebalanceamento de linhas de montagem operadas por grupos de trabalho autogerenciados. **Gestão e Produção**, v.7, n.3, p.378-398.

GARAI, G., CHAUDHURI, B. (2004). A novel genetic algorithm for automatic clustering. **Pattern Recognition Letters**, v.25, p.173-187.

GOLDBERG, D. E. (1989). **Genetic Algorithms in Search, Optimization, and Machine Learning**. USA: Addison-Wesley. 2nd ed.

GONÇALVES FILHO, E.V. (1988). **Computer-aided technology part family formation based on pattern recognition techniques**. 148f. Thesis (PhD in Industrial Engineering) - Department of Industrial and Management Systems Engineering, The Pennsylvania State University, 1988.

GONÇALVES FILHO, E.V. (2004a). **Projeto de Células de Fabricação**. 34p. Apostila.

GONÇALVES FILHO, E.V. (2004b). **Projeto de Arranjo Físico Fractal**. 9p. Apostila.

GONÇALVES FILHO, E.V. (2004c). **Projeto de Arranjo Físico Funcional**. 24p. Apostila.

GONÇALVES FILHO, E.V., GORGULHO JÚNIOR, J.H.C., ARGOUD, A.R.T.T. (2004). Algoritmo genético para formação de células de fabricação. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 24, 2004, Florianópolis. **Anais...** Florianópolis: UFSC. (CD-ROM).

GONÇALVES FILHO, E.V., TIBERTI, A.J. (2006). A group genetic algorithm for the machine cell formation problem. **International Journal of Production Economics**, v.102, p.1-21.

GORGULHO JÚNIOR, J.H.C. (2006). **Análise do desempenho dos arranjos físicos distribuídos em ambiente de programação de tarefas com flexibilidade de sequência de fabricação**. 210f. Exemplar de Qualificação (Doutorado em Engenharia Mecânica) – EESC, Universidade de São Paulo, São Carlos, 2006.

GRAVEL, M., NSAKANDA, A.L., PRICE, W. (1998). Efficient solutions to cell-formation problem with multiple routings via a double-loop genetic algorithm. **European Journal of Operational Research**, v.109, p.286-298.

HAMAMOTO, S., YIH, Y., SALVENDY, G. (1999). Development and validation of genetic algorithm-based facility layout - a case study in the pharmaceutical industry. **International Journal of Production Research**, v.37, n.4, p.749-768.

HRUSCHKA, E.R. (2001). **Algoritmos genéticos de agrupamento para extração de regras de redes neurais**. 139f. Tese (Doutorado em Ciências em Engenharia Civil) - COPPE, Universidade Federal do Rio de Janeiro, 2001.

HUANG, H. (2003). **Facility layout using layout modules**. Thesis (PhD in Industrial and Systems Engineering), The Ohio State University, Columbus, 2003.

HUANG, H., IRANI, S.A. (2002). Facility layout: new ideas and methods for research and practice - hybrid cellular layouts. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 22, INTERNATIONAL CONFERENCE ON INDUSTRIAL ENGINEERING AND OPERATIONS MANAGEMENT, 8, 2002, Curitiba. **Anais...** Curitiba: PUCPR. (CD-ROM).

IRANI, S.A. (1999). **Handbook of cellular manufacturing systems**. John Wiley & Sons.

IRANI, S.A., HUANG, H. (1998). Layout modules: a novel extension of hybrid cellular layouts. **American Society of Mechanical Engineers (Paper)**, Anaheim, p.1-7.

IRANI, S.A., HUANG, H. (2000a). A new approach for department planning to minimize interdepartmental material handling traffic in a custom manufacturing facility. In: INTERNATIONAL COLLOQUIUM ON MATERIAL HANDLING RESEARCH, 6, 2000, York. **Proceedings...** York: Material Handling Institute.

IRANI, S.A., HUANG, H. (2000b). Custom design of facility layouts for multiproduct facilities using layout modules. **IEEE Transactions on Robotics Automation**, v.16, n.3, p.259-267.

IRANI, S.A., HUANG, H. (2004). Computer-aided Production Flow Analysis for Implementing JobshopLean (JSLEAN) Manufacturing in Custom Forge Shops. Disponível em: < <http://cast.cse.ohio-state.edu/pfast/successStories.php>>. Acesso em: 11 set. 2006.

IRANI, S.A., HUANG, H. (2006). Cascading flowlines and layout modules: practical strategies for machine duplication in facility layouts. **International Journal of Flexible Manufacturing Systems**, v.17, n.2, p.119-149.

IRANI S.A, ZHOU, J., HUANG, H. (2003). A pattern recognition approach for manufacturing facility compaction by machining function combination using flexible manufacturing modules. **Journal of Manufacturing Science and Engineering**, Transactions of the ASME, v.125, n.4, p.740-752.

JEON, G., LEEP, H.R. (2006). Forming part families by using genetic algorithm and designing machine cells under demand changes. **Computers & Operations Research**, v.33, p.263-283.

KAUFMAN, L., ROUSSEEUW, P.J. (1990). **Finding groups in data: an introduction to cluster analysis**. New York: John Wiley & Sons. Wiley series in probability and mathematical statistics.

KOCHHAR, J.S., FOSTER, B.T., HERAGU, S.S. (1998). HOPE: A genetic algorithm for the unequal area facility layout problem. **Computers & Operations Research**, v.25, n.7-8, p.583-594.

KOCHHAR, J.S., HERAGU, S.S. (1999). Facility layout design in a changing environment. **International Journal of Production Research**, v.37, n.11, p.2429-2446.

KULTUREL-KONAK, S., SMITH, A.E., NORMAN, B.A. (2004). Layout optimization considering production uncertainty and routing flexibility. **International Journal of Production Research**, v.42, n.21, p.4475-4493.

KUMAR, N., SHANKER, K. (2000). A genetic algorithm for FMS part type selection and machine loading. **International Journal of Production Research**, v.38, n.16, p.3861-3887.

LACERDA, E.G.M., CARVALHO, A.C.P.L.F. (1999). Introdução aos algoritmos genéticos. In: GALVÃO, C.O., VALENÇA, M.J.S. **Sistemas Inteligentes: aplicações a recursos hídricos e ciências ambientais**. Porto Alegre: Editora da Universidade UFRGS. Capítulo 3, p. 99-150.

LACERDA, E.G.M., CARVALHO, A.C.P.L.F., LUDERMIR, T.B. (1999). Um tutorial sobre algoritmos genéticos. **RITA**, v. IV, n. 2, p.109-139.

LAHMAR, M., BENJAAFAR, S. (2005). Design of distributed layouts. **IIE Transactions**, v.37, p.303-318.

MARCONI, M.A., LAKATOS, E.M. (1985). **Técnicas de pesquisa**. São Paulo: Editora Atlas.

MORAD, N., ZALZALA, A.M.S. (1996). A genetic-based approach to the formation of manufacturing cells and batch scheduling. In: IEEE INTERNATIONAL CONFERENCE ON EVOLUTIONARY COMPUTATION, 1996. Nagoya. **Proceedings...** Nagoya: IEEE. p.485-490.

MONTREUIL, B., VENKATADRI, U., LEFRANÇOIS, P. (1991). Holographic layout of manufacturing systems. In: IIE SYSTEMS INTEGRATION CONFERENCE, 19, 1991, Orlando. **Proceedings...** p.1-13.

MOURA, E. C. (1994). **As sete ferramentas gerenciais da qualidade - implementando a melhoria contínua com maior eficácia**. São Paulo: Makron Books.

MULVEY, J.M., CROWDER, H.P. (1979). Cluster analysis: an application of lagrangian relaxation. **Management Science**, v.25, n.4, p.329-340.

NALDI, M.C., CARVALHO, A.C.P.L.F. (2005) **Técnicas de agrupamento e algoritmos genéticos**. São Carlos: ICMC-USP. 46p. (Relatório técnico, RT-SCE-267).

NARAYANAN, M. (2002). **Hybrid flow strategies for high variety low volume manufacturing facilities to implement flow and pull**. 98f. Dissertação (Master of Science in Manufacturing Systems Engineering) - College of Engineering, University of Kentucky, Lexington, 2002.

NORMAN, B.A., SMITH, A.E. (2006). A continuous approach to considering uncertainty in facility design. **Computers & Operations Research**, v.33, n.6, p.1760-1775.

PANDEY, P.C., JANEWITHAYAPUN, S. HASIN, M.A.A. (2000). An integrated system for capacity planning and facility layout. **Production Planning & Control**, v.11, n.8, p.742-753.

PIERREVAL, H. et. al. (2003). Evolutionary approaches to the design and organization of manufacturing systems. **Computers & Industrial Engineering**, v.44, p.339-364.

PIERREVAL, H., PARIS, J.L. (2000). Distributed evolutionary algorithms for simulation optimization. **IEEE Transactions on Systems, Man and Cibernetics**, Part A: Systems and Humans, v.30, n.1, p.15-24.

PROMODEL. (2006). Disponível em: <http://www.belge.com.br/produtos_promodel.html>. Acesso em: 16 nov. 2006.

ROTHER, M., SHOOK, J. (2002). **Aprendendo a Enxergar**. São Paulo: Lean Institute Brasil.

SINGH, N., RAJAMANI, D. (1996). **Cellular manufacturing systems: design, planning and control**. London: Chapman & Hall.

SLACK, N. et al. (1997). **Administração da produção**. São Paulo: Atlas.

SPÄTH, H. (1980). Hierarchical cluster algorithms. In: _____. **Cluster analysis algorithms for data reduction and classification of objects**. Chichester: Ellis Horwood Limited. p.155-194.

TAM, K.Y. (1990). An operation sequence based similarity coefficient for part families formations. **Journal of Manufacturing Systems**, v.9, n.1, p.55-68.

TARJAN, R. (1972). Depth-first search and linear graph algorithms. **SIAM Journal of Computing**, v.1, n.2, p.146-160.

TIBERTI, A.J. (2003). **Desenvolvimento de um software de apoio ao projeto de arranjo físico de fábrica baseado em um *framework* orientado a objeto**. 182f. Tese (Doutorado em Engenharia Mecânica), EESC, Universidade de São Paulo, São Carlos, 2003.

TOMPKINS, J.A. et al. (2003). **Facilities planning**. USA: John Wiley & Sons, Inc. 3rd ed. 750p.

TSENG, L., YANG, S.B. (2001). A genetic approach to the automatic clustering problem. **Pattern Recognition**, v.34, p.415-424.

UDDIN, M.K., SHANKER, K. (2002). Grouping of parts and machines in presence of alternatives process routes by genetic algorithm. **International Journal of Production Economics**, v.76, n.3, p.219-228.

UNIVERSIDADE DE SÃO PAULO. Sistema Integrado de Bibliotecas. Grupo DiTeses. (2004). **Diretrizes para apresentação de dissertações e teses da USP: documento eletrônico e impresso**. São Paulo: SIBi-USP, 2004. 110p.

VAKHARIA, A.J., WERMMERLÖV, U. (1990). Designing a cellular manufacturing systems: a materials flow approach based on operations sequences. **IIE Transactions**, v.22, n.1, p.84-97.

VENKATADRI, U., RARDIN, R.L., MONTREUIL, B. (1997). A design methodology for fractal layout organization. **IIE Transactions**, v.29, p.911-924.

VENUGOPOL, V., NARENDRAN, T.T. (1992). A genetic algorithm approach to the machine-component grouping problem with multiple objectives. **Computers and Industrial Engineering**, v.22, n.4, p.469-480.

YOLIS, E. (2003). **Algoritmos genéticos aplicados a la categorización automática de documentos**. Tesis (Grado en Ingenieria Informatica) - Facultad de Ingenieria, Universidad de Buenos Aires, Buenos Aires, 2003.

ZHOU, J., IRANI, S.A. (2000). Design of modular layouts for fabrication-based assembly facilities. In: WORLD CONGRESS ON INTELLIGENT MANUFACTURING PROCESSES & SYSTEMS, 3, 2000, Cambridge. **Proceedings...** Cambridge: MIT and CIRP. p.62-69.

APÊNDICE A – Exemplo de arquivo de entrada do AGA

/* Comentários:

Exemplo de Tabela de Seqüência de Operações. Estrutura:

- 3 colunas (peça, seqüência de operações, tempo de processamento) separadas por tabulação:
 - coluna peça: contém o identificador da peça;
 - coluna seqüência de operações: contém uma lista de identificadores das operações da peça, separadas por vírgula, na ordem de execução.
 - coluna tempo de processamento: contém uma lista de tempos das operações do lote de peças, separadas por vírgula, na ordem de execução.
- 1ª linha: nome da tabela. Deixe uma linha em branco se omitido.
- 2ª linha: rótulo das colunas. Deixe uma linha em branco se omitido.
- 3ª linha em diante: dados. Obrigatório.

*/

Seqüência de Operações Teste

Peça (P)	Seqüência de operações (Seq)	Tempo de processamento (Tp) (min/lote)
1	1, 4, 8, 9	96, 36, 36, 72
2	1, 4, 7, 4, 8, 7	36, 120, 20, 120, 24, 20
3	1, 2, 4, 7, 8, 9	96, 48, 36, 120, 36, 72
4	1, 4, 7, 9	96, 36, 120, 72
5	1, 6, 10, 7, 9	96, 72, 200, 120, 72
6	6, 10, 7, 8, 9	36, 120, 60, 24, 36
7	6, 4, 8, 9	72, 36, 48, 48
8	3, 5, 2, 6, 4, 8, 9	144, 120, 48, 72, 36, 48, 48
9	3, 5, 6, 4, 8, 9	144, 120, 72, 36, 48, 48
10	4, 7, 4, 8	120, 20, 120, 24
11	6	72
12	11, 7, 12	192, 150, 80
13	11, 12	192, 60
14	11, 7, 10	288, 180, 360
15	1, 7, 11, 10, 11, 12	15, 70, 54, 45, 54, 30
16	1, 7, 11, 10, 11, 12	15, 70, 54, 45, 54, 30
17	11, 7, 12	192, 150, 80
18	6, 7, 10	108, 180, 360
19	12	60

APÊNDICE B – Exemplo de arquivo de saída do AGA

Resultados da execução do algoritmo genético: Algoritmo Genético

[Parâmetros de execução do algoritmo genético]

Matriz de similaridade.....: Sequência de Operações Teste - Subseqüências de Operações - Matriz de Similaridade

Tamanho da população.....: 200

Número de gerações.....: 200

Taxa de transferência.....: 0,02

Método de geração.....: Faixa percentual

Taxa de cruzamento.....: 0,50

Taxa de mutação 1 (Merge).: 0

Taxa de mutação 2 (Break).: 0

Taxa de mutação 3 (Move)..: 0,50

Codificação do cromossomo.: R1

Número de grupos: 4

Função de aptidão.....: Largura da silhueta média

[Estatística de execução do algoritmo genético]

Tempo de execução(h:min:s): 00:00:01

Total de cruzamentos.....: 9800 aplicados 487 realizados

Total de mutações do tipo 1 (Merge): 0 aplicados 0 realizados

Total de mutações do tipo 2 (Break): 0 aplicados 0 realizados

Total de mutações do tipo 2 (Move).: 19600 aplicados 3338 realizados

Nro. de indivíduos iguais ao melhor na população final.....: 171

Nro. de indivíduos similares ao melhor na população final: 0

Nro. máximo de indivíduos similares na população final.....: 172

[Gráfico Convergência dos Valores de Aptidão]

Geração Menor Valor Maior Valor Valor Médio

Geração	Menor Valor	Maior Valor	Valor Médio
0	-0,0210	0,4190	0,2996
10	0,2581	0,4190	0,4046
20	0,2581	0,4190	0,4129
30	0,3224	0,4190	0,4149
40	0,2581	0,4190	0,4141
50	0,3593	0,4190	0,4161
60	0,3593	0,4190	0,4152
70	0,3593	0,4190	0,4164
80	0,3593	0,4190	0,4161
90	0,3593	0,4190	0,4168
100	0,3593	0,4190	0,4164
110	0,3593	0,4190	0,4157
120	0,3593	0,4190	0,4168
130	0,3593	0,4190	0,4157
140	0,3593	0,4190	0,4153
150	0,3593	0,4190	0,4154
160	0,3593	0,4190	0,4171
170	0,3593	0,4190	0,4163
180	0,3593	0,4190	0,4164
190	0,3593	0,4190	0,4166
200	0,3593	0,4190	0,4144

[População Inicial]

Indivíduo	K	Aptidão	Cromossomo – Grupos
1	4	0,4190	[4, 4, 3, 3, 3, 3, 2, 4, 4, 1, 1, 2, 1, 2, 1, 2, 1] - [[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]]
2	4	0,4115	[3, 3, 1, 1, 1, 1, 4, 3, 3, 2, 2, 4, 2, 1, 2, 4, 2] - [[S1, S2, S8, S9], [S3, S4, S5, S6, S14], [S7, S12, S16], [S10, S11, S13, S15, S17]]
3	4	0,4076	[1, 1, 4, 4, 4, 4, 3, 1, 1, 1, 2, 4, 2, 4, 2, 2, 2] - [[S1, S2, S8, S9, S10], [S3, S4, S5, S6, S12, S14], [S7], [S11, S13, S15, S16, S17]]

.

.

.

[População Final]

Indivíduo	K	Aptidão	Cromossomo - Grupos
1	4	0,4190	[4, 4, 3, 3, 3, 3, 2, 4, 4, 1, 1, 2, 1, 2, 1, 2, 1] - [[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]]
2	4	0,4190	[4, 4, 3, 3, 3, 3, 2, 4, 4, 1, 1, 2, 1, 2, 1, 2, 1] - [[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]]
3	4	0,4190	[4, 4, 3, 3, 3, 3, 2, 4, 4, 1, 1, 2, 1, 2, 1, 2, 1] - [[S1, S2, S8, S9], [S3, S4, S5, S6], [S7, S12, S14, S16], [S10, S11, S13, S15, S17]]

.

.

.

APÊNDICE C – *Maximum Spanning Tree*

Uma MST (*maximum spanning tree*) ou árvore de medida máxima é um procedimento heurístico baseado na teoria dos grafos. Segundo Baase apud Gonçalves Filho (2004c), uma MST pode ser usada para o projeto do arranjo físico desde que a direção do fluxo entre os departamentos (ou máquinas) não seja importante. Em outras palavras, o fluxo de A para B tem as mesmas características do fluxo de B para A.

Portanto, aplicada ao projeto de arranjo físico, a MST fornece a necessidade de proximidade das máquinas ou departamentos, a partir do fluxo entre eles. A principal fonte de informação é a matriz De-Para. Para tal é necessária a demanda de cada peça.

O algoritmo para obtenção da MST é apresentado no quadro C1.

<p>Entrada: Grafo com pesos nas arestas $G(V, E, G)$ Saída: E_1, as arestas em uma MST</p> <ol style="list-style-type: none"> 1. Inicialize os conjuntos Seja x um vértice arbitrário. $V_1 \leftarrow \{x\}$; $V_2 \leftarrow \emptyset$; $V_3 \leftarrow V - \{x\}$; $E_1 \leftarrow E_2 \leftarrow \emptyset$ 2. While $V_1 \neq V$ do 3. for cada y em V_2 adjacente a x if $W(xy) > W(a \text{ aresta } e \text{ em } E_2 \text{ incidente com } y)$ then $E_2 \leftarrow E_2 - \{e\} \cup \{xy\}$ 4. for cada y em V_3 adjacente a x do $V_2 \leftarrow V_2 \cup \{y\}$; $V_3 = V_3 - \{y\}$ $E_2 \leftarrow E_2 \cup \{xy\}$ end 5. if $E_2 = \emptyset$ then return ‘não existe <i>spanning tree</i>’ 6. Determine uma aresta, e, em E_2 com máximo peso; Defina x como sendo o vértice em V_2 incidente com e. $E_1 \leftarrow E_1 \cup \{e\}$; $E_2 \leftarrow E_2 - \{e\}$ $V_1 \leftarrow V_1 \cup \{x\}$; $V_2 \leftarrow V_2 - \{x\}$ end
--

Baase apud Gonçalves Filho (2004c)

Quadro C.1 – Algoritmo MST

Os dados de demanda usados neste trabalho foram obtidos de Vakharia e Wemmerlöv (1990) e são apresentados na tabela C.1.

Tabela C.1 – Seqüência de operações e demanda

Peça (P)	Seqüência de Operações	Tempo Total de Processamento (min / lote)	Demanda (lotes/dia)
1	1, 4, 8, 9	96-36-36-72	2
2	1, 4, 7, 4, 8, 7	36-120-20-120-24-20	3
3	1, 2, 4, 7, 8, 9	96-48-36-120-36-72	1
4	1, 4, 7, 9	96-36-120-72	3
5	1, 6, 10, 7, 9	96-72-200-120-72	2
6	6, 10, 7, 8, 9	36-120-60-24-36	1
7	6, 4, 8, 9	72-36-48-48	2
8	3, 5, 2, 6, 4, 8, 9	144-120-48-72-36-48-48	1
9	3, 5, 6, 4, 8, 9	144-120-72-36-48-48	1
10	4, 7, 4, 8	120-20-120-24	2
11	6	72	3
12	11, 7, 12	192-150-80	1
13	11, 12	192-60	1
14	11, 7, 10	288-180-360	3
15	1, 7, 11, 10, 11, 12	15-70-54-45-54-30	1
16	1, 7, 11, 10, 11, 12	15-70-54-45-54-30	2
17	11, 7, 12	192-150-80	1
18	6, 7, 10	108-180-360	3
19	12	60	2

Fonte: Vakharia e Wemmerlöv (1990)

A partir dos dados da demanda e da seqüência de operações foi construída a matriz De-Para, apresentada no quadro C.2.

DE - Máquina	PARA - Máquina											
	1	2	3	4	5	6	7	8	9	10	11	12
1	*	1		8		2	3					
2		*		1		1						
3			*		2							
4				*			9	11				
5		1			*	1						
6				4		*	3			3		
7				5			*	2	5	6	3	2
8							3	*	8			
9									*			
10							3			*	3	
11							5			3	*	4
12												*

Quadro C.2 – Matriz De-Para

O quadro C.3 apresenta a matriz De-Para diagonalizada.

DE- Máquina	PARA- Máquina											
	1	2	3	4	5	6	7	8	9	10	11	12
1	*	1	0	8	0	2	3	0	0	0	0	0
2		*	0	1	1	1	0	0	0	0	0	0
3			*	0	2	0	0	0	0	0	0	0
4				*	0	4	14	11	0	0	0	0
5					*	1	0	0	0	0	0	0
6						*	3	0	0	3	0	0
7							*	5	5	9	8	2
8								*	8	0	0	0
9									*	0	0	0
10										*	6	0
11											*	4
12												*

Quadro C.3 – Matriz De-Para diagonalizada

A partir da matriz De-Para diagonalizada foi elaborado o fluxo de informações, mostrado na figura C.1.

Figura C.1 – Fluxo de informações

A figura C.1 mostra o fluxo existente entre as máquinas do conjunto de dados. O algoritmo MST foi aplicado aos conjunto de dados da tabela C.1, a partir do fluxo do informações. A figura C.2 mostra a aplicação do algoritmo MST.

Figura C.2 – Algoritmo MST aplicado ao conjunto de dados.

Figura C.2 – continuação. Algoritmo MST aplicado ao conjunto de dados