UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO

DIANA MARIA CHAUVIN ALVEAR

APLICAÇÃO DE ALGORITMOS GENÉTICOS NA OTIMIZAÇÃO DA
TOPOLOGIA E GEOMETRIA DO LAYOUT DE UM ESTALEIRO

São Carlos

2018

DIANA MARIA CHAUVIN ALVEAR

**APLICAÇÃO DE ALGORITMOS GENÉTICOS NA OTIMIZAÇÃO DA
TOPOLOGIA E GEOMETRIA DO LAYOUT DE UM ESTALEIRO**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como requisito para a obtenção do Título de Mestre em Engenharia de Produção.

Área de Concentração: Gestão de Operações

Orientador: Prof. Dr. Walther Azzolini Júnior

São Carlos

2018

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

CC511a
a Chauvin Alvear, Diana Maria
APLICAÇÃO DE ALGORITMOS GENÉTICOS NA OTIMIZAÇÃO DA
TOPOLOGIA E GEOMETRIA DO LAYOUT DE UM ESTALEIRO / Diana
Maria Chauvin Alvear; orientador Walther Azzolini
Júnior. São Carlos, 2018.

Dissertação (Mestrado) - Programa de Pós-Graduação
em Engenharia de Produção e Área de Concentração em
Processos e Gestão de Operações -- Escola de Engenharia
de São Carlos da Universidade de São Paulo, 2018.

1. Layout. 2. Estaleiros. 3. Algoritmos Genéticos.
4. Otimização. 5. Topologia. 6. Geometria. 7. FLP. 8.
MHC. I. Título.

Eduardo Graziosi Silva - CRB - 8/8907

FOLHA DE JULGAMENTO

Candidata: Engenheira **DIANA MARIA CHAUVIN ALVEAR**

Título da dissertação: "Aplicação de algoritmos genéticos na otimização da topologia e geometria do *layout* de um estaleiro"

Data da defesa: 10/12/2018

Comissão Julgadora:

Resultado:

Prof. Titular **Henrique Rozenfeld**

(Presidente)

(Escola de Engenharia de São Carlos/EESC-USP)

aprovado

Prof. Titular **Alexandre Cláudio Botazzo Delbem**

(Instituto de Ciências Matemáticas e de Computação/ICMC-USP)

aprovado

Prof. Dr. **Guilherme Luz Tortorella**

(Universidade Federal de Santa Catarina/UFSC)

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia de Produção:

Prof. Dr. **Marcelo Seido Nagano**

Presidente da Comissão de Pós-Graduação:

Prof. Titular **Luís Fernando Costa Alberto**

À minha tia Tere, sempre no meu coração.

Aos meus amados pais, Flavio e Silvana.

Aos meus queridos sogros, Enrique e Nora.

Ao meu presentinho da vida, Naty.

Ao meu grande amor, Leo.

AGRADECIMENTOS

O meu profundo agradecimento ao Prof. Luiz César Ribeiro Carpinetti. O senhor é, nas palavras de Bertold Brecht, daqueles homens imprescindíveis. *Infinitas Gracias*.

À Marina, minha irmã da vida. Obrigada pelo colo, pela sua mão firme, pela luta diária e pelo amor. Obrigada por não me deixar cair.

Agradeço a motivação e sabedoria de Hugo, Tati e Ximena.

À minha querida amiga, Nice, pelos cuidados e bem-querer.

À Dona Marlene, por me acolher com tanto afeto e me fazer sentir em casa.

À Jéssyca e Prof. Mateus, por me mostrarem o caminho e me darem força. À Lili, pela parceria e solidariedade.

À Diego, pela força de ontem e hoje, e pelo carinho contingente e necessário de sempre.

Aos professores que durante esses cinco semestres souberam demonstrar ética, conhecimento real, comprometimento e responsabilidade, tanto no ensino quanto na pesquisa, deixando em alto o nome da USP e do Brasil.

*Nada se pierde,
Todo se transforma.*

(Jorge Drexler)

RESUMO

CHAUVIN, ALVEAR D. M. **Aplicação de algoritmos genéticos na otimização da topologia e geometria do layout de um estaleiro**. 2018. 111 f. Dissertação (Mestrado em Engenharia de Produção) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2018.

Apesar de o problema de layout de fábrica (FLP) ser amplamente estudado, esse esforço não tem se refletido no estudo de layout para estaleiros; pelo contrário, na prática, a maioria de layouts de estaleiros foi projetada com base nas experiências adquiridas pelos especialistas desta indústria, não seguindo um método analítico e sistematizado. Por essa razão, é significativo o trabalho dos autores Choi et al (2017), no sentido de propor um método quantitativo baseado nas técnicas de planejamento de layout de fábrica (FLP) e de layout arquitetônico para o problema de layout de estaleiros (ShLP); consistindo na otimização da topologia e na otimização da geometria do layout do estaleiro. Nesta dissertação, replicou-se e adequou-se o problema proposto pelos autores, aplicando um algoritmo genético baseado em ordem para a primeira etapa, e a replicação do modelo para a segunda etapa. Além disso, propõe-se também a incorporação de uma terceira etapa para a otimização geométrica, baseado em um algoritmo genético geral, no intuito de atingir melhorias no layout geométrico final. Os testes executados apontaram uma redução de 6% nos custos de manuseio de materiais, MHC. Verificou-se também a qualidade dos layouts obtidos mediante a utilização de índices de desempenho de layout de fábrica propostos pela literatura, com a finalidade de obter uma comparação correta entre o layout final reportado pelos autores e o obtido nesta dissertação.

Palavras-chave: Layout de estaleiros. Algoritmos genéticos. Otimização da topologia. Otimização da geometria.

ABSTRACT

CHAUVIN, ALVEAR D. M. **Aplicação de algoritmos genéticos na otimização da topologia e geometria do layout de um estaleiro.** 2018. 111 f. Dissertação (Mestrado em Engenharia de Produção) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2018.

Although the facility layout problem (FLP) has been extensively studied, this effort has not been reflected in the layout study for shipyards layout; on the contrary, in practice the majority of shipyard layouts were designed based on the experiences acquired by the specialists in this industry, and not following an analytical and systematized methodology. For this reason, the work of the authors Choi et al (2017) is significant in order to propose a quantitative method based on the techniques of facility layout planning methodology (FLP) and architectural layout focused for the shipyard layout problem (ShLP); consisting of optimization of the topology and geometry of a shipyard. In this study, the problem proposed by the authors was replicated and adapted, applying a genetic algorithm based on order for the first step; and the replication of the model to the second. In addition, it is also proposed the incorporation of a third step for the geometric optimization, based on a general genetic algorithm, in order to achieve significant improvements in the final geometric layout. The tests carried out indicated a 6% reduction in materials handling costs, MHC. Moreover, it was verified the quality of the layouts obtained through the use of the factory layout performance indexes proposed by the literature, in order to obtain a correct comparison between the final layout reported by the authors and the one obtained in this study.

Keywords: Shipyard layout. Genetic algorithms. Topology optimization. Geometry optimization.

LISTA DE ILUSTRAÇÕES

Figura 1 – Adaptação proposta para a resolução do problema.....	21
Figura 2 – Modelo de pesquisa quantitativa axiomática prescritiva	22
Figura 3 – Referencial teórico: temas abordados	25
Figura 4 – Abordagens do planejamento de instalações de fábrica.....	26
Figura 5 – Classificação do arranjo físico tradicional	27
Figura 6 – Classificação das abordagens do problema de layout de fábrica – FLP	30
Figura 7 – Exemplo de Navio logístico de apoio rápido (FDL).....	36
Figura 8 – Classificação das abordagens metodológicas do problema de layout.....	41
Figura 9 – Representação inteira do cromossomo ou possível solução	43
Figura 10 – Função de Rosenbrock.....	44
Figura 12 – Representação dos pesos para cada indivíduo na roleta viciada	45
Figura 13 – Operador <i>crossover</i> de um ponto para obter o filho F1	47
Figura 14 – Operador <i>crossover</i> uniforme para obter o filho F1	47
Figura 15 – <i>Crossover</i> de um ponto-linha aplicado a matrizes.	47
Figura 16 – Operador de Mutação <i>Swapping</i> e <i>Shift</i>	48
Figura 17 – <i>Crossover</i> PMX aplicado para representação de problemas de permutação	50
Figura 18 – Fluxo de Funcionamento de um GA e principais opções em cada etapa.....	51
Figura 19 – Grade retangular para os espaços <i>a</i> , <i>b</i> , <i>c</i> e <i>d</i>	54
Figura 20 – Relações topológicas entre os espaços <i>a</i> , <i>b</i> , <i>c</i> e <i>d</i>	55
Figura 21 – Representação de espaço plano utilizando unidades retangulares	55
Figura 22 – Representação do componente retangular	56
Figura 23 – Posições <i>N</i> , <i>S</i> , <i>E</i> , e <i>W</i> , alocado na grade - Iterações.....	57
Figura 24 – Configuração do layout inicial.....	59
Figura 25 – Representação da distância euclidiana entre os departamentos	62
Figura 26 – Matriz 4x5 para representar os departamentos na otimização topológica	63
Figura 27 – Definição das restrições de adjacência e alinhamento	63
Figura 28 – Matriz unitária para simbolizar a locação dos departamentos	64
Figura 29 – Posições possíveis para a restrição topológica de adjacência.	64
Figura 30 – Posições possíveis para a restrição topológica de alinhamento	65
Figura 31 – Fluxograma do GA para a primeira etapa – Otimização da topologia.....	66
Figura 32 – Exemplo de indivíduo aleatório que forma parte da população inicial.....	66
Figura 33 – Posições de alinhamento e adjacência no processo de preenchimento no vetor.	67
Figura 34 – Grade de 46x22 células e centroide dos departamentos de.....	70
Figura 35 – Processo de crescimento por iteração do departamento 3.....	71
Figura 36 – Fluxo do algoritmo de crescimento estocástico para a otimização geométrica	72
Figura 37 – Orientação horizontal e vertical dos departamentos	74
Figura 38 – Razão de aspecto requerida vs. atingida	74
Figura 39 – Os departamentos se aproximam pela distância Chebyshev.....	75
Figura 40 – Exemplo do departamento 13, RA atingida/ RA requerida	76
Figura 41 – Definição de unidade do espaço plano.....	76
Figura 42 – Operador PUSH ou empurrar	77
Figura 43 – Valores dos operadores <i>l</i> e <i>s</i> para cada direção de crescimento.....	78
Figura 44 – Exemplo de escolha da direção de crescimento entre os departamentos <i>a</i> e <i>b</i>	78
Figura 45 – Loop do crescimento estocástico dos departamentos do layout.....	80
Figura 46 – Exemplo layout A.....	81

Figura 47 – Exemplo layout B	82
Figura 48 – Recombinação dos departamentos 1, 3 e 4 com <i>crossover</i> ponto-linha	82
Figura 49 – Swapping do departamento 4.....	83
Figura 50 – Fluxograma do GA da terceira etapa proposta	84
Figura 51 – Valor da função <i>fitness</i> de cada teste para a primeira etapa.....	85
Figura 52 – Melhor layout solução - Teste 51 – atingimento de área requerida	90
Figura 53 – Melhor layout solução - Teste 28 – atingimento de razão de aspecto requerida	91
Figura 54 – Melhor layout solução - Teste 51 – atingimento de alinhamento e adjacência	92
Figura 55 – Layout obtido com 500 iterações para a sequência de Choi et al.	92
Figura 56 – Melhor filho encontrado no GA da terceira etapa.....	93
Figura 57 – Layout obtido pelos autores Choi et al.	94
Figura 58 – Espaços livres do layout de Choi et al.	95
Figura 59 – Espaços livres do layout obtido no teste 52 neste trabalho.....	96

LISTA DE QUADROS

Quadro 1 – Trabalhos referentes a layout de estaleiro	35
Quadro 2 – Indivíduos da população inicial e seus pesos correspondentes	45
Quadro 3 – Departamentos que conformam o estaleiro	59
Quadro 4 – Fluxo de materiais entre os departamentos	60
Quadro 5 – Requerimentos de alinhamento e adjacência para os departamentos	60
Quadro 6 – Probabilidades para escolha da direção de crescimento.....	74
Quadro 7 – Probabilidades para escolha da direção de encolhimento	75
Quadro 8 – Probabilidade de escolha da direção (2) maior que.....	79
Quadro 9 – Intervalo de confiança – testes da primeira etapa.....	86
Quadro 10 – Melhores soluções obtidas - primeira etapa e solução de Choi et al.	86
Quadro 11 – Sequência obtida 03	89
Quadro 12 – Valores da Função Fitness obtida nos testes	89
Quadro 13 – Percentagem de área atingida por cada teste	90
Quadro 14 – Percentagem de razão de aspecto atingida por cada teste	91
Quadro 15 – Sequência de Choi et al.	92
Quadro 16 – Intervalo de confiança do GA da terceira etapa	93
Quadro 17 – Índice de utilização de espaço layout Choi et al (2017).....	95

LISTA DE TABELAS

Tabela 1 – Classificação dos trabalhos sobre Layout de estaleiros.....	43
--	----

LISTA DE ABREVIATURAS E SIGLAS

ACO - Ant Colony Optimization
ADE - Differential Evolution Algorithm
AGVs - Auto Guided Vehicles
AHP - Analytic Hierarchy Process
ANP - Analytic Network Process
B&B - Branch and Bound
BC - Branch and Cut
BRKGA - Biased Random-key Genetic Algorithm
BSA - Backtracking Search Algorithm
CAD - Computer-Aided Design
CIP - Constraint Integer Programming
CMAES - Covariance Matrix Adaptation Evolution Strategy
CSAs - Clonal Selection Algorithm
CX - Cycle Crossover
DM - Digital Manufacturing
DP - Dynamic Programming
DRLP - Double Row Layout Problem
DWT - Deadweight Tonnage
FDL - Fast Deployment Logistic Vessel
FLP - Facility Layout Problem
GA - Genetic Algorithms
GRASP - Greedy Randomized Adaptive Search
GT - Graph Theoretic Problem
ICA - Imperialist Competitive Algorithm
IP - Integer Programming
LAA - Location-allocation algorithm
LLP - Loop Layout Problem
LP - Linear Programming
LR - Lagrangian relaxation
MFLP - Multi-Floor Layout Problem
MHC - Material Handling Costs
MIP - Mixed Integer Programming Problem
MOVDO - Vibration Damping Optimization
MRLP - Multiple Row Layout Problem
NP - Nonlinear Programming

OBX - Order-Based Crossover
OFLP - Open Field Layout Problem
OX - Order Crossover
PBX - Position-Based Crossover
PLM - Product Lifecycle Management
PMBGAs - Probabilistic Model-Building Genetic Algorithms
PMX - Partially-Mapped Crossover
PPR-S - Product, Process, Resource and Schedule
PR - Path-Relinking
PROP - Parallel Row Ordering
PSO - Particle Swarm Optimization
QAP - Quadratic Assignment Problem
QSP - Set Covering Problem
SA - Simulated Annealing
SDP - Semidefinite Programming
SE - System Engineering
SGA- Stochastic Growth Algorithm
ShLP - Shipyard Layout Problem
SLP - Systematic Layout Planning
SQP - Sequential quadratic programming
SRLP - Single Row Layout Problem
SS - Scatter Search
TS - Tabu Search
VLSI - Very Large-Scale Integration
VNS - Variable Neighborhood Search

SUMÁRIO

1. INTRODUÇÃO	19
1.1 CONTEXTUALIZAÇÃO	19
1.2 OBJETIVO.....	21
1.3 MÉTODOS PARA O DESENVOLVIMENTO DO ESTUDO	21
1.4 ESTRUTURA DO TEXTO	23
1.5 DELIMITAÇÕES DO TRABALHO.....	24
2. REFERENCIAL TEÓRICO.....	25
2.1 PLANEJAMENTO DE INSTALAÇÕES DE FÁBRICA	25
2.1.1 QAP - PROBLEMA QUADRÁTICO DE ALOCAÇÃO	28
2.1.2 CLASSIFICAÇÃO DO FLP.....	30
2.2 LAYOUT DE ESTALEIROS	35
2.3 ALGORITMOS GENÉTICOS	43
2.3.1 FUNÇÃO DE AVALIAÇÃO OU FUNÇÃO FITNESS	43
2.3.2 POPULAÇÃO INICIAL.....	44
2.3.3 SELEÇÃO.....	44
2.3.4 REPRODUÇÃO	46
2.3.5 CRITÉRIO DE PARADA.....	50
2.3.6 ESTRUTURA GERAL DO GA	51
2.4 TOPOLOGIA E GEOMETRIA DE ESPAÇOS PLANOS.....	53
2.4.1 SOLUÇÃO TOPOLÓGICA	54
2.4.2 SOLUÇÃO GEOMÉTRICA	56
3 APLICAÇÃO DE ALGORITMOS GENÉTICOS PARA A OTIMIZAÇÃO DA TOPOLOGIA E GEOMETRIA DO LAYOUT DE UM ESTALEIRO.....	59
3.1 OTIMIZAÇÃO TOPOLÓGICA – PRIMEIRA ETAPA.....	61
3.1.1 MODELO MATEMÁTICO DE CHOI ET AL., DA PRIMEIRA ETAPA	61
3.1.2 DESCRIÇÃO DA OTIMIZAÇÃO TOPOLÓGICA	62
3.1.3 MODELAGEM COMPUTACIONAL DA PRIMEIRA ETAPA	65
3.2 OTIMIZAÇÃO GEOMÉTRICA – SEGUNDA ETAPA	68
3.2.1 MODELO MATEMÁTICO DE CHOI ET AL DA SEGUNDA ETAPA	68
3.2.2 DESCRIÇÃO DA OTIMIZAÇÃO GEOMÉTRICA.....	70
3.2.3 MODELAGEM COMPUTACIONAL DA SEGUNDA ETAPA	71
3.3 TERCEIRA ETAPA PROPOSTA - COMPLEMENTO PARA OTIMIZAÇÃO GEOMÉTRICA	81

3.3.1	DESCRIÇÃO DA TERCEIRA ETAPA PROPOSTA	81
3.3.2	MODELAGEM DA TERCEIRA ETAPA PROPOSTA.....	83
4	RESULTADOS OBTIDOS.....	85
4.1	RESULTADOS DA PRIMEIRA ETAPA – OTIMIZAÇÃO TOPOLÓGICA	85
4.2	RESULTADOS DA SEGUNDA ETAPA – OTIMIZAÇÃO GEOMÉTRICA	89
4.3	RESULTADOS DA TERCEIRA ETAPA – ALCANCE OTIMIZAÇÃO GEOMÉTRICA	93
4.4	DISCUSSÃO	94
5	CONCLUSÃO	99
5.1	SUGESTÕES DE TRABALHOS FUTUROS	99
5.2	LIMITAÇÕES DA PESQUISA	100
	REFERÊNCIAS	102
	APÊNDICE A – CÓDIGO DO PRIMERIO ESTÁGIO	106
	APÊNDICE B - TESTES COMPUTACIONAIS DA PRIMEIRA ETAPA	109
	APÊNDICE C – ATINGIMENTO DOS CRITÉRIOS AVALIADOS – 2da ETAPA	112
	APÊNDICE D - MELHORES GEOMETRIAS OBTIDAS NA 2da ETAPA.....	115

1. INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Na indústria naval, os altos custos de manuseio de materiais (MHC) têm origem num layout inapropriado, uma vez que as matérias primas e produtos intermediários próprios dessa indústria possuem grandes dimensões e peso (CHOI; KIM; CHUNG, 2017). Além do mais, Matulja et al. (2009) e Song et al. (2010), afirmam que a produtividade da indústria naval fundamentalmente deve-se a um layout de fábrica bem projetado e adequado às necessidades particulares dessa indústria.

O problema de layout de instalações de fábrica (*Facility Layout Problem*, FLP) tem sido extensamente abordado pela literatura durante as últimas décadas (HOSSEINI-NASAB et al., 2018). São variados e múltiplos os estudos desenvolvidos a partir da primeira abordagem do FLP, introduzida por Koopmans e Beckmann, em 1957. O objeto de estudo do FLP é a otimização dos denominados custos de manuseio de, considerados de vital importância por representar em média entre 30% a 40% dos custos de produção totais (MATSON; MELLICHAMPH; SWAMINATHAN, 1992) podendo chegar até 70% em algumas indústrias (TOMPKINS et al., 2013).

Agora bem, a literatura tem representado o problema de layout de instalações de fábrica como o modelado matemático de “Problema Quadrático de Alocação” (*quadratic assignment problem*, QAP) (KOOPMANS; BECKMANN, 1957), e posteriormente classificado por Sahni e González (1976) como um problema do tipo NP-completo (RODRIGUES VIANA, 1998). A literatura do FLP destaca técnicas várias heurísticas e meta-heurísticas que se aproximam razoavelmente da solução exata (MAVRIDOU; PARDALOS, 1997). Um exemplo destas meta-heurística aplicadas são os Algoritmos Genéticos (GA), os quais empregam os conceitos de evolução da espécie para “evoluir” as soluções encontradas na exploração do espaço de soluções. Neste estudo, utiliza-se a meta-heurística de GA para um problema de layout de estaleiros (ShLP).

Poucos trabalhos acadêmicos foram dedicados até o momento ao estudo de otimização do FLP aplicado a estaleiros. Na prática, os projetos de layout de estaleiros foram desenvolvidos exclusivamente com base na experiência acumulada pelos especialistas do setor (SHIN et al., 2009), não seguindo uma metodologia analítica e sistematizada do projeto.

Sendo assim, destaca-se o trabalho apresentado pelos autores Choi et al. (2017), pois trata-se de uma abordagem atual (publicada no ano 2017) e puramente quantitativa, no

esforço de propor um método sistematizado para o ShLP, aplicando meta-heurísticas presentes na literatura de FLP, como é o caso dos algoritmos genéticos.

A abordagem de Choi et al. está inspirada nas metodologias aplicadas na arquitetura naval para o projeto das cabines do arranjo geral do convés de navios proposta por E. Nick (NICK, 2008). O método de E. Nick foi estendido e adaptado pelos autores Choi et al (2017) para ser aplicado ao layout de um estaleiro. A primeira etapa desta metodologia consiste na otimização da topologia dos departamentos que conformam o estaleiro, tratando-se de um problema de sequenciamento onde a ordem dos departamentos é importante devido as condições de alinhamento e adjacência requeridas no problema; sendo assim, aplica-se um GA baseados na ordem (com operadores genéricos apropriados) para conseguir explorar melhor o espaço de soluções pulando os mínimos locais e encontrado melhores soluções.

Na segunda etapa, focaliza na otimização da forma física e orientação dos departamentos (geometria) por meio de uma grade que representa o tamanho e forma real do terreno do estaleiro. No método de Choi et al., (2017), a sequência solução obtida na primeira etapa é inserida na grade representando cada posição x e y nos quadros o centroide dos departamentos. A partir daí, aplica-se um algoritmo de crescimento estocástico (SGA) (NICK, 2008; CHOI et al., 2017) para os departamentos começarem a crescer até o algoritmo atingir um determinado número de iterações. O resultado final é aquele obtido quando o algoritmo parar.

Observa-se no trabalho dos autores supracitados, a ausência de detalhes sobre o GA aplicado na primeira etapa, visto que não se apresenta o tipo de operadores genéticos aplicados ao problema (*crossover* e mutação), além do que, não se reporta o valor quantitativo atingido para a função objetivo (custos MHC) correspondente à solução obtida. Nesta dissertação esse valor foi calculado manualmente, sendo de 11.816 (ver seção 4.1).

Enquanto a segunda etapa, repara-se que o processo de crescimento dos departamentos finaliza quando se atinge um determinado número de iterações, e não quando se alcança uma área ou razão de aspecto determinada, sendo isto uma limitação do processo de otimização, dado que o layout resultante pode ou não ter atingido as áreas requeridas, mas o algoritmo parou. A aplicação de um GA geral resolveria teoricamente essa limitação, uma vez que por definição, um GA conduz a exploração do espaço de soluções a partir da solução existente, e encontra melhores soluções a partir dela. É por isto que nesta dissertação propõe-se uma terceira etapa como complemento para atingir melhores soluções a partir da solução da segunda etapa.

1.2 OBJETIVO

O objetivo da presente pesquisa é replicar o problema e o método dos autores Choi et al., (2017) e encontrar uma melhor solução (melhor mínimo e um melhor layout) para o layout estudado, através da aplicando de potenciais melhorias na abordagem da solução do problema, e acrescentando uma terceira etapa, no intuito de atingir ainda melhores mínimos para o layout solução. A proposta é ilustrada na Figura 1.

Figura 1 – Adaptação proposta para a resolução do problema

Fonte: Adaptado de Nick (2008).

A proposta da inclusão de uma terceira etapa aplicando um GA geral, está baseada na natureza própria do GA, no qual a partir de uma solução sub-ótima conhecida, no caso, a solução da segunda etapa, poderiam obter-se soluções ainda melhores baixo o critério de “evolução” da espécie dos GAs.

1.3 MÉTODOS PARA O DESENVOLVIMENTO DO ESTUDO

Este estudo poderia ser caracterizado como uma pesquisa quantitativa axiomática prescritiva. Chama-se de axiomática por ser fundamentada em um modelo matemático definido na literatura, no caso o modelo matemático de Choi et al (2017); e se propor para ele a obtenção de novas soluções, tanto para a primeira etapa do problema quando para a segunda, a partir da identificação na literatura de potenciais melhorias que poderiam ser validadas e incorporadas no método. Diz-se prescritiva axiomática devido ao fato de que a procura de uma resolução para o modelo definido é o foco central do processo de pesquisa (BERTRAND; FRANSOO, 2009). A Figura 2 apresenta o ciclo do modelo de pesquisa axiomática prescritiva.

Figura 2 – Modelo de pesquisa quantitativa axiomática prescritiva

Fonte: BERTRAND; FRANSOO (2009) apud MITROFF et al. (1974).

A presente pesquisa foi dividida em três partes: a revisão da literatura e identificação e validação das potenciais melhorias para o problema; a implementação das melhorias no método proposto por Choi et al., (2017) e obtenção de resultados; e finalmente, a análise e discussão sobre as limitações e potencialidades a partir dos resultados alcançados.

A revisão da literatura, tem como objetivo identificar na literatura potenciais melhorias que poderiam ser aplicadas ao método dos autores Choi et al (2017). Inicia-se esta etapa com a revisão exploratória na literatura sobre o problema de layout de fábrica geral (FLP). Depois, continua-se com uma revisão completa da literatura sobre layout de estaleiros (quinze trabalhos publicados) destacando nesta busca que o artigo dos autores Choi et al., (2017), artigo base desta dissertação, é a abordagem mais atual encontrada na busca. Em seguida, efetuou-se uma revisão exploratória sobre a aplicação de algoritmos genéticos ao ShLP, e finaliza-se com a revisão exploratória sobre trabalhos de otimização de espaços planos aplicados para a topologia e geometria na arquitetura.

A busca realizou-se por meio de acesso às bases de dados *Scopus* e *Web of Science* e ao buscador *Schoolar Google*, utilizando o *string*: “*facility layout problem shipyard*” AND *optimization* para o tema de FLP; *shipyard* AND *layout* para o tema de layout de estaleiros; “*genetic algorithms*” AND *optimization* AND *shipyards*, e, finalmente, “*topology optimization*” AND “*floor plane spaces*” para o último tema supracitado. Nesse levantamento bibliográfico, foram identificados no total oitenta e oito documentos, dos quais apenas quinze trataram o tema de layout de estaleiros.

Identificou-se nesta revisão o uso do crossover PMX, aplicado para problemas de layout onde a ordem dos departamentos é importante (LIU; MENG; LIU, 2013); o uso da mutação swapping para evitar a convergência prematura (TUSON, 1995), e a aplicação da distância Chebyshev como medida de similaridade para calcular os MHC.

A segunda parte da pesquisa, foi a incorporação das potenciais melhorias encontradas na literatura, no próprio método dos autores Choi et al., (2017), por meio da implementação de um GA baseado na ordem para resolver a primeira etapa do problema, sendo aplicado como operador genético o *crossover* PMX e a mutação do tipo *swapping* já mencionados. Para a segunda etapa, replicou-se o método proposto pelos autores aplicando o mesmo algoritmo de crescimento estocástico, porém aplicando a distância Chebyshev como medida de similaridade entre os departamentos, uma vez que os autores não descrevem com maior detalhe a maneira na qual os departamentos crescem o decrescem na direção certa.

Já a incorporação da terceira etapa, trata-se de uma proposta da autora deste trabalho no intuito de atingir melhores soluções a partir da solução da segunda etapa aplicando um GA geral. Finalmente, contempla-se a comparação e discussão dos resultados obtidos aplicando índices de desempenho para layout de fábrica no intuito de determinar a qualidade do layout solução obtido.

1.4 ESTRUTURA DO TEXTO

Este trabalho está organizado da seguinte maneira: No capítulo 1, apresenta-se a contextualização do ShLP; em seguida, são apresentados o objetivo e os métodos para orientar o desenvolvimento da pesquisa, e finaliza-se o capítulo com a explicação da estrutura do texto.

O capítulo 2 tem caráter descritivo, ele mostra de maneira ampla e geral do FLP, isto é: a classificação dele na literatura; os tipos de arranjos no layout das indústrias tanto os clássicos (TOMPKINS et al., 2013) quanto os especiais (ARGOUD, 2007); a descrição do modelamento matemático do FLP na literatura como um problema quadrático de alocação QAP; a ampla classificação na literatura do problema de FLP, mencionando todas as técnicas e abordagens aplicadas na literatura em cada classificação. Aborda-se também neste capítulo, a descrição dos quinze artigos aplicados a layout de estaleiros, assim como a descrição teórica da técnica de otimização usando GA, os tipos de operadores genéticos utilizados e as aplicações desta técnica em problemas relacionados a layout de estaleiros. Este capítulo finaliza com uma explicação do tratamento de espaços planos na arquitetura, e sua otimização

desde o ponto de vista de lógica dos espaços ou topologia, e o dimensionamento físico dos mesmos, ou geometria.

No capítulo 3, descreve-se os dados de entrada do layout estudado pelos autores Choi et al. (2017), assim como uma descrição detalhada do modelo matemático proposto pelos autores: parâmetros, variáveis e função objetivo; inclui-se um detalhamento conceitual da aplicada pelos autores e das melhorias implementadas no método como são o crossover PMX, mutação tipo *swapping* e distância Chebyshev. Abarca-se finalmente uma descrição da modelagem computacional para cada etapa.

O Capítulo 4, expõe os resultados obtidos nas três etapas de otimização do layout e a discussão dos layouts obtidos, comparando-os com o layout final reportado pelos autores Choi et al. Destaca-se uma discussão sobre a qualidade do layout obtida mediante critérios de desempenho de layouts de fábrica propostos na literatura.

Finalmente, no Capítulo 5, apresenta-se a conclusão do trabalho, as possibilidades de estudos futuros e as limitações da pesquisa.

1.5 DELIMITAÇÕES DO TRABALHO

Delimita-se a presente pesquisa à réplica do problema e da metodologia de solução proposta pelos autores Choi et al (2017), e aplicando potenciais melhorias até obter melhores mínimos que as reportadas pelos autores. O detalhe das melhorias implementadas apresenta-se na seção 3.

2. REFERENCIAL TEÓRICO

Nesta seção, apresenta-se o referencial teórico organizado, como mostra a Figura 3, são abordados o problema de planejamento de layout de instalações de fábrica (FLP), planejamento do layout de estaleiros (ShLP), os algoritmos genéticos (GA) e as técnicas empregadas para otimização de layout arquitetônico (topologia e geometria).

Figura 3 – Referencial teórico: temas abordados

Fonte: Própria autoria.

2.1 PLANEJAMENTO DE INSTALAÇÕES DE FÁBRICA

A primeira revisão da literatura do problema de layout, foi publicada no ano 1987 (KUSIAK; HERAGU, 1987) contemplando 101 artigos desde 1957. Posteriormente, no ano 2007 os autores Drira et al., publicaram uma segunda revisão e classificação da literatura do FLP considerando 116 artigos até então. A última revisão publicada foi em 2018, e identificando 205 artigos desde 1957 até 2017 (HOSSEINI-NASAB et al., 2018). Pode-se observar a grande quantidade de estudos publicados que tratam o problema de FLP.

Como observado, o problema de layout vem sendo estudado desde a década de 50 e sua otimização é conhecida por ter um impacto significativo na redução dos custos e no aumento da produtividade de uma planta manufatureira (DRIRA; PIERREVAL; HAJRI-GABOUJ, 2013).

O problema de layout pode ser abordado de duas maneiras: a localização da planta e o estudo do projeto da fábrica como mostrado na Figura 05 (TOMPKINS et al., 2013). A sua vez, o projeto do arranjo físico da planta compreende três componentes: o sistema de instalações (iluminação, eletricidade, comunicações, saneamento etc.); o arranjo físico que engloba o layout das áreas de produção (departamentos da fábrica, disposição da maquinaria e equipamentos); e manuseio de materiais (equipamento e veículos – AGV – para o manuseio de materiais para suporte à produção) (ver Figura 04).

Figura 4 – Abordagens do planejamento de instalações de fábrica

Fonte: TOMPKINS (2013, p. 7).

O arranjo físico é classificado em dependência do volume de produção e a variedade dos produtos, sendo que uma fábrica pode apresentar os seguintes tipos tradicionais de arranjo (ver Figura 05) (TOMPKINS et al., 2013).

Arranjo por produto: demanda estável e produto padronizado, organizado por linha de produção; por exemplo, a produção de um bloco de motor.

Arranjo por processo ou funcional: o layout da fábrica organiza-se por agrupamento de processos similares; por exemplo: área de corte, fresado, conformado, torneado.

Arranjo por tecnologia de grupo ou celular: demanda média e quantidade moderada de peças similares que constituem uma família de componentes ou organizados por grupos tecnológicos.

Arranjo posicional: demanda baixa e alta variedade de peças, produto pesado e difícil movimentação; por exemplo, produção de navios, aeronaves, atividades de manutenção.

Figura 5 – Classificação do arranjo físico tradicional

Fonte: Adaptado de Tomkins (2013, p. 78) e Argoud (2007, p. 20).

Existem na literatura outros tipos de layouts não tradicionais, mas orientados para problemas de planejamento de layout de máquinas, tais como: arranjo ágil, arranjo distribuído, arranjo celular híbrido, arranjo fractal e arranjo modular (ARGOUD, 2007).

No estudo quantitativo do FLP, a literatura tem representado o problema com os seguintes modelos: Problema quadrático de alocação (QAP) - Koopmans e Beckmann (1957); Problema de cobertura de conjuntos (QSP) - Bazaraa MS (1975); Programação linear (LP); Programação inteira (IP) - Lawler (1963); Programação não linear (NP); Programação inteira mista (MIP) - Kaufman L. e Broeckx F. (1978) e Teoria de Grafos (GT) - Foulds e Robinson (1976). Observa-se que modelo QAP é amplamente estudado pela literatura, com 35% dos estudos publicados até hoje (HOSSEINI-NASAB et al., 2018). Esse modelo será abordado na subseção 2.1.1.

O modelo clássico QAP é um problema de posicionamento de modelos e foi desenvolvido pelos autores Koopmans e Beckmann, em 1957, na locação de facilidades. A partir desta formulação original Tomkins e White (1984) incluíram na formulação original, por exemplo, requisitos de área dividindo o terreno do layout em uma grade de divisões iguais e resolvendo o problema mediante o sistema CRAFT, baseado em um algoritmo de melhoramento; enquanto Amour e Buffa (1963) apresentaram uma metodologia para determinar a localização relativa de departamentos em uma fábrica de áreas desiguais.

Rosenblatt, em 1987, apresentou a primeira formulação de um problema de layout multicritério. Para a resolução do problema, propôs uma heurística de melhoramento que conseguia a minimização dos custos de movimentação e, ao mesmo tempo, a maximização da proximidade entre os departamentos. Kouvelis et al., em 1992, acrescentaram ao problema um

indicador de flexibilidade do layout. Os autores apontaram que um layout robusto é mais pertinente, porque consegue ter a flexibilidade para atender diferentes cenários de demanda, enquanto o layout ótimo matemático é discriminatório e só funciona para um cenário.

Kar Yan Tam, em 1992, tratou o FLP incluindo restrições geométricas para o qual adiciona na função objetivo uma função de punição por não atendimento dessas restrições. Enquanto que, em 1996, C-W Chen e D Y Sha propuseram uma abordagem multiobjetivo na qual, além do objeto quantitativo do problema, incorpora um tópico qualitativo: a proximidade entre os departamentos.

No ano 1998, Rajasekharan, Peters e Yang adicionam restrições de orientação para os departamentos e pontos de coleta e entrega. Também nesse ano, Atilia Islier (ISLIER, 1998) acrescentou o conceito de unidade de carga no modelamento do FLP. Yang, Taho et al. (2003) propuseram uma abordagem de análise hierárquica na avaliação do problema de layout. Logo depois, em 2002, K Y Lee et al. trataram o problema de adjacência entre os departamentos, assim como em 2003 A. Gomes et al., tratou além da adjacência, o problema de layout focando na variação da largura das passagens entre os departamentos (GÓMEZ et al., 2003).

Em 2006, G. Aiello et al. apresentaram uma análise multicritério de decisão, aplicando o método Electre para resolver o FLP. Os autores consideraram a satisfação do MHC, adjacência, distância e razão de aspecto das áreas. No mesmo ano, A. R. S. Amaral apresentou o problema de layout de linha única (*single row*), utilizando métodos exatos.

Yang e Hung, em 2007, utilizaram a técnica *fuzzy*, no intuito de classificar os layouts em ordem de preferência de soluções similares para achar a solução ideal.

Em 2012, Anjos e Liers utilizaram modelos de otimização cônica para problemas de planejamento de piso de contorno fixo, fornecendo soluções ótimas globais para até problemas com 40 departamentos. Em 2015, Gonçalves e Resende abordaram o FLP para departamentos de área desigual para o qual utilizaram um algoritmo genético de chave aleatória enviesada (*Biased Random-key Genetic Algorithm*, BRKGA).

2.1.1 QAP - PROBLEMA QUADRÁTICO DE ALOCAÇÃO

O modelo original de QAP (*Quadratic Assignment Problem*) é um problema de otimização combinatória no qual se posicionam n departamentos em n localidades distintas e procurando minimizar o custo de interação entre os pares de departamentos. (GOLDBARG; GOLDBARG; PACCA L. LUNA, 2016). O QAP foi desenvolvido pelos autores Koopmans e

Beckmann em 1957, nesse modelo, minimiza-se a função de custo do fluxo resultante da iteração entre departamentos atribuídos a um local específico, minimizando a distância total entre eles. A distância é considerada como uma medida de similaridade entre os departamentos do layout (ARGOUD, 2007) e pode ser calculada como distância euclidiana ou manhattan, dependendo do problema.

A equação (01) apresenta a modelagem matemática aplicado ao problema, tratando-se de uma permutação $\pi \in S_n$ das matrizes de fluxo $F=(f_{i,j})$ e distância $D=(d_{i,j})$ de um número determinados de departamentos, sendo S_n o número de permutações $\{1, \dots, n\}$ GOLDBARG et al., (2016):

$$\min_{\pi \in S_n} \sum_{i=1}^n \sum_{j=1}^n f_{\pi(i)\pi(j)} \cdot d_{i,j} \quad (01)$$

A solução do problema consiste em encontrar a permutação $\pi \in S_n$ que minimize o somatório. Um layout com n departamentos teria $n!$ possíveis soluções (LIGGETT, 1980).

O QAP é um modelo que atrai a atenção dos pesquisadores por conta da sua complexidade e aplicabilidade. Ele é usado para representar e resolver problemas de otimização de posicionamento de módulos (layout de planta, hospital, zoneamento, circuitos de placas); problemas de sequenciamento (caixeiro viajante, *scheduling*, roteamento de antenas, processamento de imagens); e problemas de particionamento (particionamento de grafos, carregamento de veículos, programação de tempo de curso escolar, estrutura de conexão e navegação website) (LIGGETT, 1980).

A complexidade do problema foi demonstrada pelos autores Sahni e Gonzalez no ano 1976. Eles demonstraram que o QAP é um problema *NP-completo*, no qual o tempo que toma um algoritmo até encontrar a solução é dado por uma função de complexidade de tempo baseada no tamanho do problema a resolver. Como afirmam GOLDBARG et al., (2016) uma instancia do QAP com poucas dezenas de variáveis pode ser computacionalmente difícil de resolver.

Por esta razão, a literatura apresenta métodos heurísticos e meta-heurísticos para se chegar perto da solução exata. Essas técnicas são tratadas na seção 2.1.2, em que se apresenta também uma classificação do problema de FLP publicada pelos autores HOSSEINI-NASAB et al., em 2018.

2.1.2 CLASSIFICAÇÃO DO FLP

É importante destacar no presente trabalho, os intuitos de classificações do amplo problema de FLP na literatura. A primeira classificação e *survey* sobre o FLP foi publicada em 1987 pelos autores Kusiak e Heragu. Até 2007 não aparecia uma nova classificação do problema, quando foi publicado o trabalho dos autores Drira et al.; sendo este trabalho a base para futuras classificações. Em 2018, Hosseini-Nasab et al. apresentaram um esforço por classificar o problema de layout contemplando os artigos contidos na literatura até 2017. A continuação descreve-se uma síntese destes três trabalhos base.

O FLP está classificado em quatro focos: o primeiro foco cobre os autores que abordaram o problema de layout por evolução, isto é: consideraram layout estático, dinâmico ou flexível. O segundo foco estuda os autores que abordaram o problema pelas características da oficina (problema de fila única, problema de filas múltiplas, layout celular, etc.). O terceiro foco refere-se aos autores que modelaram o problema por meio de um modelo matemático em específico (QAP, MIP, LP, Grafos); e o quarto foco, cobre os estudos que abordaram a solução do problema de FLP por meio de diversas técnicas (solução exata, técnicas qualitativas e quantitativas, heurísticas, meta-heurísticas). A Figura 6 resume essa classificação.

Figura 6 – Classificação das abordagens do problema de layout de fábrica – FLP

Fonte: Adaptado de Hosseini-Nasab et al (2018) e Drira et al (2007).

Evolução do Layout

A diferença substancial entre esses tipos de layout é a variação do fluxo entre os departamentos em um determinado horizonte de tempo. Um layout estático estuda uma situação fixa de fluxo, enquanto um layout dinâmico considera mudanças nesse fluxo.

Os autores sublinham que, dos trabalhos estudados, 77% consideraram problemas de layout estático, enquanto 23% avaliaram layout dinâmico. Uma observação importante desse autor é que, para tornar mais realistas os problemas que atendem o layout dinâmico, devem-se incluir nas restrições do problema restrições orçamentárias, sendo que somente cinco trabalhos incluíram essas restrições.

A literatura tem abordado o estudo de layout de fábrica (FLP) por meio de diversas e variadas formulações matemáticas, mencionadas a seguir (KUSIAK; HERAGU, 1987), (HOSSEINI-NASAB et al., 2018):

- Problema quadrático de alocação (QAP) Koopmans e Beckmann (1957);
- Problema de cobertura de conjuntos (QSP) Bazaraa MS (1975);
- Programação linear (LP);
- Programação inteira (IP) Lawler (1963);
- Programação não linear (NP);
- Programação inteira mista (MIP) Kaufman L. e Broeckx F. (1978);
- Teoria de Grafos (GT) Foulds e Robinson (1976).

O modelamento QAP e MIP são os modelos mais explorados para a resolução do FLP, pois 70% de todas as publicações têm sido abordadas com essas formulações (35% MIP e 35% QAP). Nesta dissertação, trata-se da formulação QAP.

Características da Oficina

Essa classificação compreende quatro subclassificações: por sistemas de manufatura, pela forma e dimensão dos departamentos, pelo sistema de manuseio de materiais e pelo sentido do fluxo. O tipo de produto manufaturado determina o layout da fábrica (TOMPKINS et al., 2013). Em um layout por produto, os recursos da fábrica deslocam-se até o produto, que permanece em uma posição fixa, sendo esse o layout de estaleiros. Em um layout por processo, os recursos (máquinas, pessoas) são agrupados por similaridade da atividade, enquanto que em um layout celular as operações requeridas para fabricar uma parte ou peça são organizadas em uma célula de trabalho. Enquanto a forma dos departamentos, a literatura trata o problema baseada na forma dos departamentos, ou seja, se eles mantêm uma forma regular ou não. Também, são apresentadas abordagens onde os departamentos têm

largura/altura fixa ou não. Sobre o sistema de manuseio de materiais, a literatura aborda o problema fundamentada no tipo de transporte que a fábrica utiliza, podendo ser transportados os produtos por bandas, carros transportadores, guindastes, veículos autoguiados (AGVs) etc.

A configuração do layout da fábrica depende do caminho do fluxo do material, sendo: problema de fila única (*SRLP*), problema de filas múltiplas (*MRLP*), problema de fila dupla (*DRLP*), problema de ordenação de filas paralelas (*PROP*), problema de layout de circuito (*LLP*), problema de layout de campo aberto (*OFLP*), problema de layout de múltiplos andares (*MFLP*) (ANJOS; VIEIRA, 2017).

Formulação do Problema

O problema de layout foi também abordado pela literatura baseado na representação clássica representação discreta e contínua do problema. Em geral, a abordagem discreta utiliza o modelamento QAP e a abordagem contínua utiliza a programação inteira mista, MIP.

No que se refere à função objetivo, ambas as abordagens, quantitativa e qualitativa, procuram o menor MHC possível entre os departamentos com base na distância. As restrições de alinhamento e adjacência são, então, consideradas na análise qualitativa para maximizar a proximidade entre os departamentos. A escala de proximidade geralmente utilizada é a proposta pela metodologia de planejamento sistemático de layout proposta por R. Muther, em 1973.

Em relação às restrições, Hosseini-Nasab et al. (2018) destacam requisitos de limites de área, áreas iguais ou desiguais, relação de aspecto, orientação geográfica, não superposição das áreas, pontos de entrega e recepção (*pick-up/ drop-off*), e orçamento. Ressalta-se que nem todas essas restrições são abordadas de maneira simultânea.

Em relação ao tipo de dados, os autores adotaram abordagens determinísticas e não determinísticas. Na abordagem determinística, os dados de fluxo e custo de transporte são conhecidos a priori. No que concerne à abordagem não determinística, considera-se que os dados de fluxo e custo de transporte são desconhecidos; para o tratamento do problema são utilizadas formulações de lógica *fuzzy* e formulações estocásticas (HOSSEINI-NASAB et al., 2018).

Abordagem da Resolução do Problema

Nessa classificação, a literatura de FLP sub-classifica a abordagem da resolução do problema em: métodos exatos; metodologias gerais; heurísticas; e modelos computacionais.

Entre os métodos exatos desenvolvidos na literatura para encontrar a solução exata, destacam o algoritmo de ramificação e poda (*Branch and Bound*), e o algoritmo de plano de corte (*Cutting Plane*) (HOSSEINI-NASAB et al., 2018). Os métodos exatos se listam a continuação:

- Programação dinâmica (DP);
- Ramificação e poda (B&B);
- Ramificação e corte (BC);
- Programação inteira de restrição (CIP);
- Algoritmo de planos de corte;
- Programação semi-definida (SDP);
- Relaxamento Lagrangiano (LR).

Enquanto as metodologias gerais propostas para encontrar a solução do FLP multi-objetivo, destacam-se: a técnica de soma ponderada, o método dos conjuntos dominantes e o processo analítico por hierarquia são as metodologias mais empregadas pelos pesquisadores (HOSSEINI-NASAB et al., 2018). O total de metodologias implementadas se apresenta a seguir:

- Método de soma ponderada;
- Método de restrição ϵ ;
- Método de critério global;
- Ordenamento lexicográfico;
- Análise envoltória de dados;
- Programação *fuzzy* multiobjetivo;
- Método dos conjuntos dominantes;
- Processo analítico por hierarquia;
- Processo analítico por redes;
- TOPSIS;
- ELECTRE;
- Índice de seleção de preferência;

- Julgamento *fuzzy* de valor sintético;
- Agrupamento *c-mens* por *clusters*;
- Classificação simples de vários atributos.

Enquanto as heurísticas e modelos computacionais, distinguem-se as seguintes (HOSSEINI-NASAB et al., 2018):

Heurísticas:

- Planejamento Sistemático de Layout, SLP, R. Muther (1973);
- Algoritmos de melhoria;
- CRAFT, COFAD;
- Algoritmos construtivos;
- CORELAP, ALDEP, PLANET.

Modelos Computacionais:

- Algoritmos Genéticos (GA);
- Recozimento simulado (AS);
- Busca Tabu (TS);
- Otimização de enxame de partículas (PSO);
- Otimização de colônia de formigas (ACO);
- Busca de vizinhança variável (VNS);
- Algoritmo de seleção clonal (CSAs);
- Algoritmo guloso (*Greedy Algorithm*);
- Explorar trajetórias, conectando soluções (PR);
- Busca de dispersão (SS);
- Algoritmo de evolução diferencial (ADE);
- Algoritmo imperialista competitivo (ICA);
- Busca aleatória adaptativa gananciosa (GRASP);
- Estimação de algoritmo de distribuição (*Estimation of Distribution Algorithm*), também conhecido como Algoritmos genéticos probabilísticos para construção de modelos (EDAs ou PMBGAs);
- Otimização de amortecimento de vibrações (MOVDO);

- Algoritmo de busca por retrocesso (BSA);
- Estratégia de evolução da adaptação da matriz de covariância (CMAES).

Sendo os Algoritmos Genéticos e o Recozimento simulado os mais empregados pelos pesquisadores do problema de FLP (HOSSEINI-NASAB et al., 2018). O presente trabalho foca nos modelos computacionais motivados pelos autores Choi et al (2017), o GA e SGA são as meta-heurísticas investigadas.

2.2 LAYOUT DE ESTALEIROS

Como mencionado nas secções prévias, a revisão da bibliografia desenvolvida no presente estudo identificou quinze documentos relacionados ao tratamento do ShLP. O Quadro 1 apresenta os itens identificados nas bases de dados *Scopus e Web of Science* e no buscador *Scholar Google*, desde o ano 1946 até 2017, relacionados com o *string* “shipyard layout”

Quadro 1 – Trabalhos referentes a layout de estaleiro

Ano	Autor (es)
1946	Lin Chen-Ming e Yeh Yu-Hu
1965	Benjamin V. Andrews, Dan G. Haney
1976	Suk Chang
1981	A. K. Syrkov
1997	A. Yucel Obadaçi, Selim Alkaner, Aykut Olcer, Ned
2004	Hamid Chabane
2009	Jong Gye Shin, Young Joo Song, Dong Kun Lee, Jo
2009	Tin Matulja, Nikša Fafandjel, Albert Zamarin
2010	Young Joo Song, Dong Kun Lee, Jong Hun Woo, Jo
2013	Yang Liu, Mei Meng, Shuang Liu
2013	Dong-Kun Lee, Yong-Kuk Jeong, Jong-Gye Shin
2013	Young Joo Song, Jong Hun Woo
2014	Sanghwan Kim, Hyun Chung, Mingyu Kim
2014	Seung Jae Lee, Jing Hun Woo, Jong Gye Shin
2017	Minjoo Choi, Sang Hyun Kim, Hyun Chung

Fonte: Própria autoria.

Os primeiros estudos sobre layout de estaleiros estavam restritos ao âmbito militar. Depois da segunda guerra mundial, houve por parte dos países envolvidos o interesse na logística para movimentação massiva de recursos, tropas e suprimentos que o transporte aéreo

não conseguia cobrir (ANDREWS; HANEY, 1965). Os navios logísticos de apoio rápido (FDL) (Figura 07) atenderam a essa necessidade de fontes de suprimento massivo de equipamentos e demais recursos.

Figura 7 – Exemplo de Navio logístico de apoio rápido (FDL)

Fonte: BMT GROUP¹.

Em 1965, os estado-unidenses Benjamin V. Andrews e Dan G. Haney apresentaram o relatório “*Conceptual Design of a Mechanized Shipyard for Fast Deployment Logistics Production*”. Nesse documento, os autores enfatizaram que os estaleiros típicos não conseguiriam atender eficientemente a quantidade de navios FDL requeridos, razão pela qual propuseram o projeto de um estaleiro dedicado e altamente mecanizado para a construção de navios tipo FDL. A intenção do trabalho dos autores era a reconfiguração do pátio do estaleiro sueco Arendal para viabilizar a construção de navios tipo FDL, mediante a incorporação de técnicas avançadas de mecanizado. Trata-se de um trabalho descritivo com abordagem qualitativa.

Posteriormente, em 1976, foi publicado o primeiro artigo a respeito do ShLP. Trata-se do trabalho do sul-coreano Suk Chang “*Fundamentals of Shipyard Layout*” (CHANG, 1976). Esse artigo, escrito em idioma coreano e inglês, apresentou os fundamentos para o processo de projeto básico e detalhado para um estaleiro típico. O autor calculou o layout, representando as atividades em uma rede de nós, onde cada atividade é um nó, ligando uma atividade à outra pelo tempo de execução. Utilizando como exemplo o fluxograma de construção por blocos de um navio graneleiro, o autor projetou um layout com maior taxa de

¹ A imagem está disponível em: <<https://www.bmt.org/markets/defence/defence-platform-design-and-support/specialist-naval-platforms/>>. Acesso em: 12 março, 2018.

uso para as estações de trabalho de montagem de painéis planos. Finalmente, apresentou uma análise do tempo de construção aproximado desse layout. Considerando o ano de publicação deste artigo, destaca-se que trata-se do primeiro artigo de layout de estaleiros que aborda quantitativamente o problema modelando a função objetivo com grafos.

Em 1981, foi escrito na forma instrutivo, o documento *Handbook on the engineering design of shipyards, shipbuilding shops and sheds* (Manual de projeto de engenharia de estaleiros navais, oficinas de construção naval e galpões), do russo A. K. Syrkov (SYRKOV, 1981), no qual examinou o projeto detalhado de estaleiros utilizando como parâmetro principal os custos de MHC para a determinação das áreas de trabalho, com o objetivo de revisar e projetar estaleiros com altos índices de produtividade, baixo custo e qualidade de produção. Este documento, originalmente de uso reservado, foi aberto para acesso ao público em 2007. Trata-se do documento mais completo publicado até aquele momento relativo no à projeto de estaleiros, pois foram inseridos inúmera técnicas e cálculos qualitativos e quantitativos para todas as etapas do projeto do layout produtivo e da execução das operações de manufatura. Entre outros tantos aspectos básicos tratados no manual, é possível mencionar as análises das áreas de abastecimento, de pré-montagem, de conformação de peças, de montagem de painéis e blocos, de anodizado, de pintura e secado, dos laboratórios de controle de qualidade, do número e capacidade dos equipamentos necessários, das necessidades de fornecimento de energia etc.

Ressalta-se neste estudo o foco para mostrar a aplicabilidade técnico-econômica das carreiras de lançamento inclinadas *versus* a montagem horizontal dos navios, junto com os desafios intermediários do transporte e montagem de partes e blocos em outra situação, dado que os equipamentos necessários num estaleiro, no caso os grandes guindastes, são o principal equipamento pelo preço de aquisição e utilização de um estaleiro.

Em todo momento, as análises técnico-econômicas conduziram a diferentes amostras da melhor decisão conforme a especialização e tipo de estaleiro a ser considerado. Outro exemplo de análise foi revisar a configuração de equipamentos necessários, conforme a decisão do grau técnico a ser alcançado antes ou depois do lançamento do navio. Ainda foram tratados temas coligados às rotinas produtivas, tais como: segurança, recursos humanos, proteção ambiental etc. Sublinha-se que as técnicas e cálculos empíricos apresentados neste documento estão orientadas ao projeto do estaleiro e sua aplicação técnica-econômica, tendo assim um valor desde o ponto de vista de sua aplicação prática, porém não apresenta nenhum intuito de otimização do layout obtido para as oficinas ou a disposição das maquinarias.

Em 1997, apareceu a primeira tentativa de abordagem sistemática para projeto de estaleiros. O estudo de A. Y. Obadaçi et al. (OBADAÇI et al., 1997) consideraram o problema de avaliação do layout de dois estaleiros turcos: Marmara e Torgem. Os autores aplicaram a metodologia proposta por R. Muther (1973), Planejamento Sistemático de Layout (SLP), no intuito de converter essas instalações em estaleiros de terceira geração, sendo elas estaleiros da primeira geração.

Em 2004, Hamid Chabane (CHABANE, 2004) adotou também a metodologia delineada por R. Muther (1973) para projetar o layout inicial para um estaleiro construtor e reparador concebido para produzir entre 1000 a 5000 DWT por ano. O limitante dessa metodologia decorre da análise principalmente qualitativa, na qual o layout ajusta-se baseado em premissas sobre o fluxo de produção.

Em 2009, T. Matulja, N. Fafandjel, e A. Zamarin, inspirados também na metodologia de Muther (1973), propuseram uma nova abordagem processual de quatro fases para projetar o layout de um estaleiro. Nesse procedimento, destaca-se, primeiramente, a relação de proximidade entre os departamentos mediante a técnica de levantamento de especialidades relevantes; a avaliação dos layouts gerados, seguindo critério de proximidade; a modelagem hierárquica para escolher o layout que melhor atende aos requerimentos; e a verificação da estabilidade do layout, mediante análise de sensibilidade (MATULJA; FAFANDJEL; ZAMARIN, 2009).

Observa-se que tanto o trabalho de A. Y. Obadaçi et al., (1997), quanto os trabalhos de H. Chabane (2004) e T. Matulja et al (2009), seguem a mesma abordagem qualitativa procedural baseada na metodologia SLP de R. Muther (1973), o que chama a atenção dado que para o ano de publicação destes trabalhos, a metodologia de R. Muther era antiga, enquanto a literatura de FLP já abordava qualitativamente o problema.

Em 2009, os autores Jong Gye Shin, Young Joo Song, Dong Kun Lee, Jong Hun Woo, apresentaram uma ferramenta computacional que foi o resultado de uma pesquisa sobre o projeto sistemático de layout de estaleiros. Os autores fundamentaram-se na metodologia de engenharia de sistemas (SE) orientada pela norma ISO/ IEC 15288. O procedimento proposto considerou a estrutura de um sistema conjugado de análise de requisitos e capacidade projetada com base no fluxo das famílias de produtos, processos, recursos e a programação da produção (PPR-S) Uma vez atingido o projeto mais razoável, o procedimento culmina com a verificação e validação do layout mediante simulação digital (SHIN et al., 2009).

Em 2010, Young Joo Song, Dong Kun Lee, Jong Hun Woo e Jong Gye Shin, continuaram a linha de Shin et al., (2009) e apresentaram as melhorias alcançadas pela ferramenta computacional de projeto sistemático de estaleiros que eles criaram em 2009 (SONG et al., 2010). Os autores conseguiram incorporar nessa plataforma que o projetista do layout acessasse a informação da engenharia de produto e do processo de projeto, compatível com os conceitos de ciclo de vida dos produtos (PLM) e manufatura digital (DM) utilizada nos estaleiros da Coreia. A estrutura dessa ferramenta está baseada nos critérios de proximidade para simular o layout e na validação da capacidade de produção, mediante simulação e análise do fluxo de produção de um navio padrão.

O ano de 2013 foi muito produtivo e diferenciado na geração de literatura sobre ShLP. Nesse ano, foram publicados três artigos com abordagens diferentes, porém, incluíram um tratamento quantitativo no problema.

O primeiro trabalho foi das chinesas Yang Liu, Mei Meng e Shuang Liu (LIU; MENG; LIU, 2013). Elas abordaram o problema de layout de filas múltiplas (*multi-row*) para a oficina de processamento de aço de um estaleiro mediante um modelo matemático de programação inteira mista (MIP). Para a resolução do problema, as autoras propuseram um algoritmo genético melhorado, no qual o *crossover* tradicional foi substituído pelo *crossover* do tipo parcialmente mapeado (*partially-mapped crossover*, PMX); e a taxa de seleção era controlada pela relação de Boltzmann. Posteriormente, um modelo paramétrico CAD permitia a visualização tridimensional do problema. Finalmente, o modelo de layout ótimo é avaliado mediante o método de análise hierárquica (AHP).

Paralelamente, os coreanos Dong-Kun Lee, Yong-Kuk Jeong e Jong-Gye Shin (LEE; JEONG; SHIN, 2013) consideraram o problema de layout para um estaleiro construtor de iates. A abordagem sistemática do problema cobriu aspectos qualitativos e quantitativos. Para modelar o layout, os autores perceberam que o sistema de produção desse tipo de estaleiro é uma combinação entre as linhas de produção das indústrias de automóveis e as oficinas usadas na indústria de construção naval. Devido ao mencionado, a produção desse estaleiro foi organizada em arranjos celulares tipo *job-shop*, de forma que permitissem atingir certa flexibilidade no processo produtivo para atender às mudanças de volume de produção e produção sob encomenda. O problema foi modelado com abordagem quantitativa mediante o QAP, com o objetivo de otimizar a distribuição das oficinas. Para encontrar a solução ótima,

utilizou-se uma heurística de localização e alocação que maximiza a distribuição das oficinas, ocupando a mínima área possível.

A última abordagem desse ano foi a dos também coreanos Young Joo Song e Jong Hun Woo (SONG; WOO, 2013). Essa pesquisa foi apresentada como uma extensão das pesquisas de Shin et al. (2009) e Song et al. (2010). Os autores usaram o sistema de projeto de estaleiro proposto pelos pesquisadores anteriormente mencionados e projetaram um layout para um estaleiro seguindo um método integrado, no qual utilizaram dados reais da construção de um navio padrão e dados do processo de produção para estimar a área requerida. Para o cálculo da solução do problema de otimização, os autores utilizaram o software comercial VIP-PLANOPT.

Em 2014, os coreanos Seung Jae Lee, Jong Hun Woo e Jong Gye Shin (LEE et al., 2014) propuseram um método para o ShLP com base nos dados reais do produto e nos tempos reais da operação. Também nesse ano, Sanghwan Kim, Hyun Chung e Mingyu Kim propuseram uma metodologia adaptada do problema de layout dinâmico e layout robusto, na qual os autores substituíram os custos de MHC e o custo de rearranjo pelo custo operacional e custo adicional de construção.

Os autores concentraram a análise no setor de pré-edificação. O argumento principal foi adaptar a estratégia construtiva de “*mix*” de produtos e aplicá-la para “*mix*” de blocos a serem produzidos, considerando as características dos blocos respectivamente para cada tipo de produto. A premissa foi o mínimo investimento adicional e, portanto, identificar a melhor relação de área de pré-edificação para fabricação dos blocos de navios comerciais *versus* a área destinada à fabricação dos blocos para plataformas *off-shore*. O estudo ainda foi confirmado com simulações dos fluxos de processo em software especializado. Uma parte importante desse trabalho é a classificação das metodologias para layout de instalações industriais, apresentada na Figura 8.

Figura 8 – Classificação das abordagens metodológicas do problema de layout

Fonte: LEE et al. (2014).

Finalmente, em 2017, os coreanos Minjoo Choi, Sang Hyun Kim e Hyun Chung apresentaram uma pesquisa sobre layout de estaleiros inspirada nas metodologias aplicadas na arquitetura naval para o projeto do arranjo do convés de navios. A metodologia de otimização proposta por E. Nick (2008) para o arranjo de um navio militar, foi adaptada pelos autores e aplicada para o layout de um estaleiro. A metodologia mencionada consta de um procedimento de dois estágios. No primeiro estágio, é otimizada a topologia dos departamentos e, no segundo estágio, a geometria dos mencionados departamentos. O primeiro estágio minimiza os custos MHC, respeitando restrições de adjacência usando algoritmo genético. No segundo estágio, são otimizadas as áreas do layout por meio de um algoritmo de crescimento estocástico. Nesse passo, os autores propõem uma conversão do domínio topológico para o domínio geométrico, utilizando uma grade cujas dimensões mínimas respeitam o tamanho do bloque de maior tamanho construído pelo estaleiro.

A continuação apresenta-se na Tabela 1, a classificação dos quinze trabalhos de layout de estaleiros referidos à classificação do problema de FLP abordado na seção 2.1. Desta forma, observa-se que os artigos 10 e 12 estão relacionados com a abordagem de Choi et al (2017) e aplicam a mesma meta-heurística de GA. O artigo número 10 é justamente o trabalho em que se aplica o operador genético PMX, o mesmo que será aplicado no GA desta dissertação. Adicionalmente, é interessante observar como até o ano 2010, as abordagens dos trabalhos

eram puramente qualitativas (a exceção do artigo No. 3), mesmo que na literatura de FLP o foco era qualitativo, isto mostra quanto os trabalhos de ShLP não acompanhavam as ferramentas já disponíveis na literatura de FLP.

Tabela 01. Classificação dos trabalhos sobre Layout de estaleiros

No	Ano	Título	Abordagem	Classificação	Sub-classificação	Técnica
1	1946	Shipyards Layout	Qualitativa	Formulação do problema	Restrições	Por Orçamento
2	1965	Conceptual design of a Mechanized Shipyards for fast Deployment Logistics Production	Qualitativa	Características da oficina	Layout por produto	n/a
3	1976	Fundamentals of Shipyards Layout	Quantitativa	Formulação do problema	Grafos	n/a
4	1981	Handbook of engineering design of Shipyards, shipbuilding shops and sheds	Qualitativa	Formulação do problema	Restrições	por Áreas
5	1997	Reengineering of small and medium-sized ship production facilities: An example for Turkish Shipbuilding Industry	Qualitativa	Formulação do problema	Multi-atributo	AHP
6	2004	Design of a Small Shipyards Facility Layout Optimized for Production and Repair	Qualitativa	Formulação do problema	Multi-atributo	AHP
7	2009	A Concept and Framework for a Shipyards Layout Design Based on Simulation	Qualitativa	Formulação do problema	Multi-atributo	AHP
8	2009	Methodology for Shipyards Production Areas Optimal Layout Design	Qualitativa	Abordagem da solução	Multi-atributo	AHP
9	2010	System Development and Applications of a Shipyards Layout Design Framework	Qualitativa	Abordagem da solução	Objetivo simples	Sistema de especialistas
10	2013	Layout Design-Based Research on Optimization and Assessment Method for Shipbuilding Workshop	Quantitativa	Abordagem da solução	Modelo Computacional	GA
11	2013	Study on a Layout Design Method for Leisure Ship Production Factories using a Heuristic Location-Allocation Algorithm	Quantitativa	Abordagem da solução	Modelo Computacional	LAA
12	2013	New shipyard layout design for the preliminary phase & case study for the green field project	Quantitativa	Abordagem da solução	Modelo Computacional	GA
13	2014	Optimization of P.E. area division and arrangement based on product mix	Quantitativa	Abordagem da solução	Layout dinâmico	Simulação
14	2014	New business opportunity: Green field project with new technology	Quantitativa	Formulação do problema	Tipo de dados	Determinísticos
15	2017	Optimal shipyard facility layout planning based on a genetic algorithm and stochastic growth algorithm	Quantitativa	Abordagem da solução	Modelo Computacional	GA e SGA

Fonte: Própria autoria

2.3 ALGORITMOS GENÉTICOS

Os algoritmos genéticos (GA) são técnicas meta-heurísticas de busca e otimização global que são inspirados na teoria da evolução natural das espécies de Charles Darwin (LINDEN, 2012). Os GAs baseiam-se nos estudos de John Henry Holland (1975, 1992) e foi aperfeiçoada por seu aluno David E. Goldberg, em 1988.

Metaforicamente, os GA consistem em gerar várias soluções, chamadas de “indivíduos”, “cromossomos” ou “*string*”, com a finalidade de obter a população inicial do espaço de soluções (possíveis soluções do problema). Essas soluções podem ser geradas aleatoriamente ou baseadas no conhecimento do domínio do problema. A forma de representação dos indivíduos de uma população por ser por meio de um cromossomo ou sequência de números, binários, inteiros ou reais. A Figura 9 apresenta um exemplo representação, em que cada número inteiro é um possível valor de uma variável do problema original. Por exemplo, o número ‘19’ no início da sequência, é o valor desta solução para a variável x_1 no problema.

Figura 9 – Representação inteira do cromossomo ou possível solução

Inteira:

19	20	7	4	3	18	10	9	5	2	12	11	8	13	17	1	6	14	15	16
----	----	---	---	---	----	----	---	---	---	----	----	---	----	----	---	---	----	----	----

Fonte: Própria autoria.

2.3.1 FUNÇÃO DE AVALIAÇÃO OU FUNÇÃO FITNESS

A função de avaliação é uma função que representa e contém todos os objetivos do problema do mundo real a resolver, podendo incluir as restrições contidas nele. Essa função retorna a um valor numérico que representa o grau de qualidade da solução obtida (LINDEN, 2012).

Em termos práticos a função de avaliação atribui uma nota a cada indivíduo, com forma de diferenciar um indivíduo ruim de um indivíduo bom. Do ponto de vista de otimização, uma solução obtida melhor pode ser uma aproximação melhor do ótimo global.

Um exemplo simples é a maximização da função de *Rosenbrock*: $f(x) = (3 - x)^2 + 4(y - x^2)^2$, apresentada a seguir (Figura 10).

Figura 10 – Função de Rosenbrock

Fonte: SAHA (2014).

Várias possíveis soluções (x,y) para a função podem ser geradas aleatoriamente. Algumas das soluções com valores mais altos $f(x,y)$ podem estar próximas do máximo desta função.

2.3.2 POPULAÇÃO INICIAL

A população inicial é gerada de maneira aleatória, em geral, evitando duplicidade de dois indivíduos iguais (duas soluções iguais). Cada indivíduo da população inicial é avaliado pela função *fitness* (função objetivo do problema). Em geral, seleciona-se dessa população os indivíduos que obtiverem os melhores valores na função para se tornarem os pais da seguinte geração.

2.3.3 SELEÇÃO

Depois que a população inicial é formada, escolhem-se os cromossomos pais para se reproduzirem e formarem uma nova geração de indivíduos ou cromossomos filhos. O método de seleção deve simular o que acontece na natureza, onde os indivíduos mais fortes teriam mais probabilidade de gerar descendência. O método mais utilizado, conforme identificado na literatura, chama-se roleta viciada (LINDEN, 2012). Ele permite beneficiar os indivíduos com melhores condições genéticas. Nesse método, atribui-se a cada indivíduo um peso, ou probabilidade de ser escolhido, compatível com sua avaliação; sendo assim, e dependendo do número de vezes em que o sorteio virtual é feito, todos os indivíduos da população inicial

teriam certa possibilidade de serem escolhidos, não desprezando aqueles com baixa probabilidade, pois agregam diversidade genética à população.

Tomando como exemplo os indivíduos *A*, *B*, *C* e *D*, apresentados no Quadro 2, com suas respectivas codificações em notação binária e o valor da função *fitness* para cada um deles. Note-se que o indivíduo *A* é o melhor avaliado, razão pela qual teria um maior peso ou probabilidade de ser selecionado, enquanto o indivíduo *D* apresenta o menor peso.

Quadro 2 – Indivíduos da população inicial e seus pesos correspondentes

Indivíduo	Rep. Binária					<i>Fitness</i>
A	1	0	0	1	1	→ 210
B	1	1	0	1	0	→ 105
C	0	0	1	0	1	→ 84
D	0	1	0	0	1	→ 21
Total:						→ 420

Fonte: Própria autoria.

Sendo assim, se a roleta gira aleatoriamente (LINDEN, 2012), o indivíduo *A* teria mais probabilidade de ser selecionado (50%), sem desprezar o indivíduo *D*, que também teria possibilidade de ser escolhido, mesmo tendo baixa qualidade genética (Figura 12).

Figura 11 – Representação dos pesos para cada indivíduo na roleta viciada

Fonte: Adaptado de Goldberg, 1989.

Além do método da roleta viciada, R. Linden (2012) destaca vários outros métodos para selecionar os pais, como seguem:

1) Seleção por Torneio (GOLDBERG; DEB, 1991): o torneio se realiza selecionando aleatoriamente k indivíduos da população inicial (mínimo $k=2$), de tal forma que os k indivíduos concorram entre eles e somente sobrevive aquele com melhor valor *fitness*;

2) Seleção por amostragem estocástica uniforme: esse método é similar ao método da roleta viciada, a diferença é que são atribuídos segmentos lineares para cada indivíduo proporcional à sua avaliação. Para a seleção, utiliza-se aleatoriamente um ponteiro de distância uniforme e ajustado ao tamanho da população, que aponta um segmento linear conforme sua probabilidade;

3) Seleção por Ranking (MITCHELL, 1996): esse método consiste em definir uma função de ranking para avaliar os indivíduos da população inicial; posteriormente, os indivíduos são ordenados com base nessa ordenação. A vantagem apresentada por esse método é a diminuição da convergência genética; porém, precisa-se de mais tempo para achar a melhor solução (LINDEN, 2012);

4) Seleção por Truncamento (MÜHLENBEIN; SCHLIERKAMP-VOOSE, 1991): nessa técnica, de uma população inicial de N indivíduos, são escolhidos os M ($M < N$) melhores indivíduos, após a ordenação descendente do valor da função *fitness*.

2.3.4 REPRODUÇÃO

As técnicas de reprodução contemplam o uso de dois métodos chamados de operadores genéticos: *crossover* e mutação. O operador genético de recombinação, ou *crossover*, tem a finalidade de transmitir a informação genética às novas gerações e, dessa maneira, atingir cada vez melhores indivíduos ou soluções para o problema a otimizar. A literatura apresenta várias formas de recombinação e mutação, a-influenciam na qualidade das soluções obtidas (JUNQUEIRA MARTARELLI, 2016). O tipo de *crossover* aplicado deve se adequar ao tipo do problema a ser resolvido.

Crossover

O *crossover* consiste em recombinar parte do material genético dos pais e formar um ou vários filhos para a nova geração. A Figura 13 apresenta um exemplo simples desse operador: supõe-se que se precisa obter a melhor sequência para 20 tarefas realizadas em duas máquinas; então, temos a primeira solução aleatória, ou cromossomo pai, chamada de P1 (pai 1), e a segunda de P2 (pai 2). Essas soluções são geradas aleatoriamente e o cromossomo filho, chamado de F1, é gerado por meio da recombinação de um intervalo do cromossomo do primeiro pai (primeira metade do cromossomo do P1) e parte do cromossomo do segundo pai

(segunda metade do cromossomo de P2). Esse tipo de *crossover* é o chamado *crossover* de um ponto. Existem muitos outros tipos de *crossover*, sendo os mais convencionais o *crossover* de 1, 2 ou n pontos, e o *crossover* uniforme; este último apresenta-se na Figura 14. (LINDEN, 2012).

Figura 12 – Operador *crossover* de um ponto para obter o filho F1

Fonte: Própria autoria.

Figura 13 – Operador *crossover* uniforme para obter o filho F1

Fonte: Própria autoria.

O *crossover* de n pontos pode ser estendido e aplicado a uma representação bidimensional, usando matrizes para as soluções. Por exemplo, a Figura 15 representa os indivíduos usando uma matriz bidimensional (MOTTA TOLEDO et al., 2014); nesse caso, o *crossover* aplicado é o *crossover* chamado de ponto-linha vertical.

Figura 14 – *Crossover* de um ponto-linha aplicado a matrizes.

Fonte: Adaptado de Motta Toledo et al. (2014) e notas de aula *Introdução aos algoritmos evolutivos* SSC5858.

Mutação

O operador de mutação tem a finalidade de agregar diversidade à população, evitando a convergência genética prematura (TUSON, 1995). A mutação ajuda a escapar de máximos ou mínimos locais, permitindo pular entre o espaço de soluções por meio da troca aleatória das posições dos genes no cromossomo (ver Figura 18). A mutação atua com uma baixa taxa de incidência (LINDEN, 2012) pois se aplica em geral a uma taxa seguindo a relação de $1/l$, em que l é o tamanho da população, ou $1/s$, em que s é comprimento do cromossomo (EIBEN; SMITH, 2012).

A Figura 16 apresenta o mecanismo do operador de mutação aplicado no cromossomo P1; esse tipo de mutação é conhecido como mutação *swapping*. Outro tipo de mutação é a chamada mutação tipo *shift*, mostrada na Figura 12 (DE SOUZA, 2004). Nessa mutação, um intervalo do cromossomo desloca-se de acordo com o ponto de mutação escolhido.

Figura 15 – Operador de Mutação *Swapping* e *Shift*

Fonte: Adaptado de De Souza (2004).

De Lacerda e De Carvalho (2009) apresentam vários outros operadores de *crossover* que se aplicam para casos de permutação, por exemplo: *crossover* PBX, PBX, PMX, CX e OX. O *crossover* PMX é detalhado a seguir.

Crossover de mapeamento parcial, PMX:

Um tipo especial de *crossover* é o chamado *crossover* PMX (*partially mapped crossover*), geralmente utilizado em algoritmos genéticos aplicados a problemas de otimização combinatória (LINDEN, 2012), também chamados de GA para representação de permutações (EIBEN; SMITH, 2012). Nesse problema, o valor da função fitness depende principalmente da sequência dos genes no cromossomo.

Um dos operadores genéticos aplicados a esse tipo de problema de otimização é o *crossover* PMX (EIBEN; SMITH, 2012), que respeita as sequências requeridas, começando com dois pontos de corte escolhidos aleatoriamente e realizando trocas no sentido de $P1$ para $P2$, seguindo o ciclo exposto na Figura 17. Neste ciclo observa-se o primeiro pai, $P1$ e o segundo pai, $P2$ gerando se o filho F .

Se inicia o ciclo pegando aleatoriamente uma sequência de $P1$ (ou dois pontos de corte) e transmitindo os valores das posições para as posições no filho F (x_4, x_5, x_6, x_7). Posteriormente, observa-se quais são os cromossomos repetidos na sequência, sendo que os valores repetidos em $P2$ não participarão do ciclo. Depois, pega-se em $P2$ o valor correspondente à primeira posição da sequência (posição x_4) e salva-se o valor (no caso o valor de '8'). Procura-se em $P1$ o valor da mesma posição x_4 (no caso '4') e busca-se a posição desse número em $P2$, sendo encontrado na posição x_9 , verifica-se no filho F a disponibilidade dessa posição e insere-se o valor salvo de $P2$ ('8') nesta mesma posição em F (ver Figura 17).

Continua-se o ciclo com a seguinte posição da sequência: x_5 em $P2$. No caso, o valor da posição x_5 é '2' sendo esse número salvo. Verifica-se em $P1$ o valor da mesma posição sendo o valor de '5', procura-se em F esse valor sendo encontrado na posição x_7 já é parte de F , então volta-se a procurar a posição x_7 em $P1$ (no caso, o valor '7') e procura-se esse valor em $P2$, localizando-se na posição x_3 . A posição x_3 está livre em F , portanto procede-se a inserir o valor na posição x_3 de F .

Figura 16 – Crossover PMX aplicado para representação de problemas de permutação

Fonte: EIBEN; SMITH (2012).

2.3.5 CRITÉRIO DE PARADA

A literatura destaca alguns critérios de parada do processo evolutivo. De um modo geral, o processo é interrompido quando se atinge um número pré-determinado de gerações ou quando há convergência do valor da função de avaliação (DE LACERDA ESTÉFANE GM; DE CARVALHO, 1999).

2.3.6 ESTRUTURA GERAL DO GA

A Figura 18 apresenta a estrutura geral de um GA (MITCHELL, 1988), detalhada a seguir:

- 1) Gera-se a população inicial de n indivíduos ou cromossomos com k genes;
- 2) Avalia-se o valor de função *fitness* para cada indivíduo da população;
- 3) Inicia-se o ciclo da Reprodução a seguir, que compreende:
 - 3.1) Operador de seleção escolhe os pais com base no critério de seleção adotado;
 - 3.2) Taxa de *crossover* é aplicada para o grupo definido em 3.1; esses cromossomos pais irão gerar filhos seguindo o mecanismo de cruzamento escolhido (*crossover* de n pontos, uniforme etc.);
 - 3.3) Aplica-se o operador de mutação em parte da população de 3.2 (probabilidade de 0-10%) e o cromossomo resultante forma parte da nova geração;
- 4) A nova população substitui a população anterior (poderia se aplicar elitismo);
- 5) Até que o critério de parada for satisfeito, retornar ao passo 2.

Figura 17 – Fluxo de Funcionamento de um GA e principais opções em cada etapa

Fonte: Adaptado de Gen e Cheng (GEN; CHENG, 1997)

Como relatado na literatura, os GAs têm sido aplicados para a resolução do FLP. Azadivar e Wang (AZADIVAR; WANG, 2000) utilizaram GA para minimizar o tempo do ciclo da produção. Ming (2005) aplicou GA para minimizar o volume do fluxo, considerando a área e a forma dos departamentos de um layout. Aiello et al., (AIELLO; ENEA; GALANTE, 2006) resolveram o problema de layout multiobjetivo com GA para minimizar o volume de fluxo de materiais e maximizar os requisitos de adjacência. E. Nick (2008) utilizou uma abordagem de duas etapas para determinar o arranjo geral ótimo do navio, usando GA e algoritmos de crescimento. Kwon e Lee (KWON; LEE, 2015) consideraram um GA para resolver problemas de alocação de blocos para o *scheduling* de estaleiros.

Nesta dissertação, como já mencionado, aplica-se um GA baseado em ordem para resolver o a primeira etapa do problema apresentado por Choi et al. (2017). Aplica-se este algoritmo considerando que se trata de um problema de permutação no qual a ordem dos departamentos (sequência) no vetor solução é importante, e a literatura destaca que para esse tipo de abordagem aplica-se operadores evolutivos especiais (LINDEN, 2012)0; (LACERDA et al., 1999), tal como o *crossover* PMX. O *crossover* PMX foi utilizado pelas autoras Liu et al. (LIU; MENG; LIU, 2013) para a otimização do layout da oficina de aço de um estaleiro.

Como já mencionado nas secções anteriores, a terceira etapa consiste na implementação de um GA geral para atingir uma melhor solução (layout obtido na otimização geométrica) a partir da solução da segunda etapa. Aplicando um *crossover* de ponto-linha na representação matricial dos departamentos e a mutação tipo swapping, procura-se a obtenção de uma solução (filho) com melhor fitness. O GA geral proposto nesta etapa é aplicado seguindo o critério base de um GA, que é a evolução natural da espécie, com isto busca-se atingir melhores mínimos dos já obtidos utilizando como população inicial (pais) desta etapa as soluções obtidas na segunda. Os detalhes do modelo de otimização são apresentados na seção 3.

2.4 TOPOLOGIA E GEOMETRIA DE ESPAÇOS PLANOS

O planejamento de espaços planos tem sido estudado pelos arquitetos e pesquisadores desde a década de 1960. Desde então, várias heurísticas e algoritmos têm sido desenvolvidos (YOON; COYNE, 1992). Entre as técnicas para abordar a otimização de espaços planos, destacam-se as seguintes:

- Algoritmos estocásticos (SHARPE et al., 1985; JO; GERO, 1998; JAGIELSKI; GERO, 1997);
- Algoritmos evolutivos (DAMSKI; GERO, 1996; Choi et al., Ano?);
- Algoritmos híbridos SA/SQP (MICHALEK et al., 2002);
- Rede Bayesiana (MERRELL et al., 2010);
- Programação inteira binária e SA (Hwang Yi e Yun Kyu Yi, 2014).

Planejar um espaço plano consiste em encontrar o melhor arranjo espacial em relação aos objetivos topológicos e geométricos, geralmente expressos em forma de restrições (MEDJDOUB; YANNOU, 2000).

A maneira mais utilizada pelos pesquisadores para representar um espaço plano é definindo o espaço disponível como um conjunto de quadrados arranjados em uma grade retangular (ver Figura 19), e usando um algoritmo para alocar uma atividade específica a um quadrado (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002). A partir da representação dos espaços na grade, consegue-se adaptar o problema para vários outros tipos de aplicações de planejamento do layout de espaços, tais como: automação da configuração espacial, empacotamento de componentes, planejamento de rotas, layout de processos e instalações, projeto de VLSI e layout arquitetônico (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002).

Figura 18 – Grade retangular para os espaços *a*, *b*, *c* e *d*

		b	b		
		b	b		
		b	b		
		b	b		
a	a	b	b	d	d
a	a	b	b	d	d
a	a	b	b	d	d
a	a	b	b	d	d
a	a	b	b	d	d
c	c	c	c		
c	c	c	c		
c	c	c	c		

Fonte: Adaptado de Michalek, Choudhary e Papalambros (2002, p. 4262).

Como dito anteriormente, existem dois níveis de objetivos da solução para o planejamento de espaços: a solução topológica e a solução geométrica do espaço (YI; YI, 2014), (DAMSKI; GERO, 1996), apresentadas a seguir.

2.4.1 SOLUÇÃO TOPOLÓGICA

A solução topológica de espaços planos refere-se aos requisitos de hierarquia semântica para avaliar a usabilidade do layout final (YI; YI, 2014). Eles permitem especificar a adjacência, a não adjacência ou a proximidade de um espaço com outro (MEDJDOUB; YANNOU, 2000). Esses requisitos espaciais representam as relações posicionais lógicas e qualitativas dos espaços, que são invariáveis, independentemente da forma geométrica deles (DAMSKI; GERO, 1996). Considere, por exemplo, o conjunto de restrições topológicas que os espaços *a*, *b*, *c*, *d* devem cumprir, apresentadas na Figura 20: *a* toca *b*, *a* está acima de *c*, *c* está à esquerda de *d*, *d* é adjacente à *c*; essas restrições poderiam gerar múltiplas soluções para a configuração final dos espaços *a*, *b*, *c* e *d*.

Figura 19 – Relações topológicas entre os espaços *a*, *b*, *c* e *d*

Fonte: DAMSKI; GERO (1996, p. 170).

Considere outro exemplo de requisitos semânticos para o relacionamento dos espaços de uma casa (YOON; COYNE, 1992): a casa tem 5 espaços (sala, cozinha, banheiro, dormitório e corredor). Os requisitos são declarados a seguir: a sala deve estar ao lado da cozinha, o dormitório ao lado do corredor, e o banheiro adjacente ao dormitório (ver Figura 21).

Figura 20 – Representação de espaço plano utilizando unidades retangulares

Fonte: Adaptado de Damski e Gero (1997, p. 856).

Tanto os espaços da Figura 20, quanto os da Figura 21, são configurados em uma grade que geraria uma grande quantidade de possíveis soluções topológicas para os espaços adjacentes e alinhados (espaços encostados) (DAMSKI; GERO, 1996). Note-se que a restrição de adjacência não é, necessariamente, uma representação de contato físico entre os espaços; ela é aplicada geralmente para controlar a posição relativa de um espaço com outro, através do comprimento do contato ou da distância entre eles (MEDJDOUB; YANNOU, 2000). As definições topológicas restringem o espaço do desenho geométrico; por exemplo, a cozinha da Figura 21 é adjacente à sala, condição que restringe as coordenadas geométricas da cozinha em relação à sala; assim, a qualidade da solução topológica está baseada no grau de cumprimento desses requisitos topológicos; então, pode-se selecionar várias soluções topológicas para depois avaliá-las geometricamente (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002).

2.4.2 SOLUÇÃO GEOMÉTRICA

A solução geométrica refere-se ao processo para determinar a melhor posição e tamanho de cada componente retangular ortogonal da grade geométrica (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002), respeitando as restrições: limite do terreno, orientação dos espaços e as restrições de adjacência entre eles (YOON; COYNE, 1992). A geometria é dimensionada em função de requisitos estéticos e práticos, tais como boa proporção de espaço, utilização de área mínima, entre outros (MEDJDOUB; YANNOU, 2000).

O componente retangular é definido como um espaço ortogonal alocado para uma função específica (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002), sendo que um ou vários retângulos podem representar um departamento do layout (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002). O componente retangular está definido geometricamente pelo ponto do centroide no espaço (x,y) e a distância perpendicular do centro aos limites norte (N), sul (S), leste (E), e oeste (W) do espaço (ver Figura 22).

Figura 21 – Representação do componente retangular

Fonte: MICHALEK; CHOUDHARY; PAPALAMBROS (2002, p. 463).

Essa representação proposta por Michalek et al. (2002) possibilita efetuar diversas alterações na forma do espaço em função das posições N , S , E , e W , sem variar a centroide, ou alterar a centroide sem modificar as coordenadas; portanto, essa representação proporciona flexibilidade em cada iteração da otimização geométrica (ver Figura 23) (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002).

Figura 22 – Posições N, S, E, e W, alocado na grade - Iterações

Fonte: Adaptado de Michalek et al. (2002).

Onde:

N: número de retângulos até o limite norte desde o centroide do espaço;

S: número de retângulos até o limite sul desde o centroide do espaço;

E: número de retângulos até o limite leste desde o centroide do espaço;

W: número de retângulos até o limite oeste desde o centroide do espaço;

c_x : coordenada da posição do centroide no eixo horizontal;

c_y : coordenada da posição do centroide no eixo vertical.

Ressalta-se que o posicionamento absoluto inicial das centroides é necessário, devido à sobreposição da relação entre a solução topológica e geométrica; portanto, cada topologia poderia ser interpretada geometricamente de vários modos (MEDJDOUB; YANNOU, 2000). Para solucionar essa sobreposição, utilizam-se algoritmos heurísticos que têm a possibilidade de explorar vários *designs* até se atingir layouts de qualidade razoável (MEDJDOUB; YANNOU, 2000).

A solução geométrica persegue múltiplos objetivos para o layout final, que são definidos pelo projetista. Por exemplo, no layout de uma casa, minimizar a perda de calor durante os meses frios e maximizar a iluminação natural; em uma planta, maximizar o espaço utilizável do terreno, as áreas e formas dos espaços etc. Geralmente, a função da solução geométrica é uma função multiobjetivo na qual cada objetivo tem uma ponderação w adequada aos objetivos do projeto final.

3 APLICAÇÃO DE ALGORITMOS GENÉTICOS PARA A OTIMIZAÇÃO DA TOPOLOGIA E GEOMETRIA DO LAYOUT DE UM ESTALEIRO

O estaleiro estudado no trabalho dos autores Choi et al (2017) apresenta-se na Figura 24. Observa-se que o estaleiro apresenta certas restrições, como a entrada, as áreas não disponíveis, e a posição dos cais e docas flutuantes. Estes espaços devem ser considerados como restrição na representação do problema. Também, apresenta-se os seguintes dados de entrada:

- Fluxo entre os departamentos (Quadro 4)
- Área e Razão de Aspecto desejadas para cada departamento (Quadro 3)
- Restrições de alinhamento e adjacência entre os departamentos (Quadro 5).

O estaleiro está projetado para uma capacidade aproximada de processamento de 150000 t. de aço por ano.

Figura 23 – Configuração do layout inicial

Fonte: CHOI et al. (2017).

Quadro 3 – Departamentos que conformam o estaleiro

No.	Departamentos	Área Ideal	Aspecto Ideal	Topologia	Geometria
1	Escritório	9	2	Livre	Livre
2	Armazém	28	2	Livre	Livre
3	Pátio de Estocagem	15	2	Livre	Livre
4	Pré-tratamento	14	3	Livre	Livre
5	Fabricação	50	2	Livre	Livre
6	Fabricação de anteparos	12	2	Livre	Livre
7	Linha de painel	22	4	Livre	Livre
8	Montagem de peças	25	2	Livre	Livre

9	Submontagem	50	2	Livre	Livre
10	Montagem de blocos	35	2	Livre	Livre
11	Primeira pré-verticalização	30	2	Livre	Livre
12	Espaço de estocagem (blocos)	40	2	Livre	Livre
13	Pré-equipagem	8	2	Livre	Livre
14	Pintura	20	4	Livre	Livre
15	Segunda pré-verticalização	144	4	Fixa	Livre
16	Espaço de estocagem (primeiro pier)	50	2	Fixa	Fixa
17	Espaço de estocagem (segundo pier)	34	2	Fixa	Fixa
18	Departamento fixo	-	-	-	-
19	Departamento fixo	-	-	-	-
20	Departamento fixo	-	-	-	-

Fonte: CHOI et al. (2017).

Quadro 4 – Fluxo de materiais entre os departamentos

De	Para	Quantidade	De	Para	Quantidade
3	4	1800t	8	10	120t
4	5	1200t	9	10	1300t
4	7	660t	10	11	700b
5	8	800t	10	13	300b
5	9	95t	11	12	600b
5	10	48t	12	14	600b
7	9	660t	13	11	300b
8	7	4t	14	15	600b
8	9	670t			

Fonte: CHOI et al.(2017).

Quadro 5 – Requerimentos de alinhamento e adjacência para os departamentos

Dep. No.	Dep. No.	Adjacência	Alinhamento
3	4	sim	sim
4	5	sim	não
4	7	sim	sim
5	9	sim	sim
7	9	sim	não
9	10	sim	sim
11	12	sim	sim

Fonte: CHOI et al. (2017).

A continuação detalha-se as duas etapas do método proposto por Choi et al., (2017).

3.1 OTIMIZAÇÃO TOPOLÓGICA – PRIMEIRA ETAPA

3.1.1 MODELO MATEMÁTICO DE CHOI ET AL., DA PRIMEIRA ETAPA

O modelo matemático proposto por Choi et al., para otimizar a topologia dos departamentos por meio da minimização do custo de manuseio de materiais MHC é descrito pela equação 02 (CHOI; KIM; CHUNG, 2017).

$$\text{Minimizar } F = \left[\sum_{i \in D} \sum_{j \in D} f_{i,j}^{tp} \cdot c_{i,j}^{tp} \cdot d_{i,j}^{tv} \right] \quad (02)$$

Sujeito a:

$$x_f^{tv} = p_f^{tp} \quad ; \quad f \in F \quad (03)$$

$$x_i^{tv} \neq x_j^{tv} \quad ; \quad i, j \in G \quad (04)$$

$$x_g^{tv} \in \{1, 2, \dots, N\} \quad ; \quad g \in G \quad (05)$$

$$d_{i,j}^{tv} = \sqrt{(cx_i^{tv} - cx_j^{tv})^2 + (cy_i^{tv} - cy_j^{tv})^2} \quad ; \quad i, j \in D \quad (06)$$

A distância $d_{i,j}^{tv}$ entre os departamentos é considerada como a distância euclidiana entre os centroides dos departamentos na matriz da Figura 25, conforme equação 06.

Os conjuntos, parâmetros e variáveis definidos pelos autores são:

Conjuntos:

D: Conjunto dos departamentos d ;

G: Genes do cromossomo g ;

F: Conjunto dos genes fixos no cromossomo f .

Parâmetros:

$f_{i,j}^{tp}$: Fluxo de materiais entre os departamentos;

$c_{i,j}^{tp}$: Custo de transporte por distância percorrida e por unidade de volume departamentos i e j ;

p_f^{tp} : Departamento predeterminado inicialmente atribuído ao gene f ;

N : Número de departamentos.

Variáveis:

- x_g^{tv} Departamento atribuído a uma posição ao gene g ;
- d_{ij}^{tv} Distância euclidiana entre os departamentos i e j ;
- cx_d^{tv} Centróide x do departamento d ;
- cy_d^{tv} Centróide y do departamento d .

As letras superiores “ tp ” e “ tv ” indicam "parâmetro de topologia" e "variável de topologia" (CHOI et al., 2017).

Note-se que os departamentos 19, 20, 18, 17, 15 e 16 têm posição fixa e são atribuídos ao conjunto dos genes fixos: x_1^{tv} , x_2^{tv} , x_6^{tv} , x_{15}^{tv} , x_{19}^{tv} , e x_{20}^{tv} , respectivamente. Os departamentos não fixos, representados pela variável x_g^{tv} , a qual indica o departamento atribuído ao g -*enésimo* gene, são então atribuídos aleatoriamente aos genes restantes quando a população inicial é gerada, como indicado anteriormente. Nesta dissertação, não foi necessária a utilização da variável binária $cl_{i,j}^{tv}$ e $cl_{i,j}^{tv}$ dos autores, já que essa verificação é feita na própria codificação do algoritmo.

Figura 24 – Representação da distância euclidiana entre os departamentos

Fonte: CHOI et al. (2017).

3.1.2 DESCRIÇÃO DA OTIMIZAÇÃO TOPOLÓGICA

A função objetivo a otimizar é a equação 2 da seção 3.1.1. A equação representa a função de avaliação ou função *fitness* no GA, para a otimização de um layout que consiga atender às restrições topológicas entre os departamentos que constituem o estaleiro. Para modelar esse problema, os autores Choi et al. (2017), empregaram uma matriz 4x5 exibida, na

Figura 26. Observa-se que nesta matriz cada posição g_i está referenciada a uma posição x_{gi} no vetor solução ou cromossomo; portanto, o cromossomo representará as possíveis permutações entre os departamentos até atingir o menor valor para a função *fitness*.

Figura 25 – Matriz 4x5 para representar os departamentos na otimização topológica

Fonte: CHOI et al. (2017).

As relações topológicas dos departamentos de um layout referem-se às relações lógicas que os departamentos entre si devem atender (MICHALEK; CHOUDHARY; PAPALAMBROS, 2002). Por exemplo, se um departamento i deve ser adjacente a um departamento j ; ou, se um departamento i deve permanecer fixo, entre outras variações. A Figura 27 exhibe as definições de adjacência e alinhamento propostas por Choi et al. (2017), para serem implementadas nessa etapa de otimização topológica.

Figura 26 – Definição das restrições de adjacência e alinhamento

Restrições Topológicas	
Adjacência	Alinhamento
<div><div>12</div><div>12</div><div>11</div></div>	<div><div>12</div><div>12</div><div>11</div></div>

Fonte: CHOI et al. (2017).

Para otimizar as relações topológicas, os autores Choi et al. (2017) utilizaram uma matriz de 4x5 com dimensão unitária e ilustrada na Figura 31 para representar a disposição dos departamentos no layout. Destaca-se que as áreas não disponíveis são tratadas como departamentos fixos; no caso, trata-se dos departamentos 18, 19 e 20, que serão representados na cor cinza nas células do canto inferior esquerdo da matriz da Figura 28. Em relação aos departamentos 16 e 17, correspondentes às áreas de estoque dos cais. Os autores posicionaram esses departamentos no canto superior direito da matriz e são considerados fixos. O departamento 15 é considerado fixo por se tratar de um requisito do problema (ver Quadro 2).

Figura 27 – Matriz unitária para simbolizar a locação dos departamentos

Fonte: Adaptado de Choi et al. (2017).

Adicionalmente, os autores utilizam essa matriz para representar as restrições topológicas de alinhamento e adjacência listadas no Quadro 5. Considerando a restrição de adjacência, eles restringem o posicionamento desses departamentos nas células marcadas em cor verde na Figura 29. Da mesma forma, considerando a restrição de alinhamento, eles restringem o posicionamento desses departamentos nas células marcadas em cor azul, como representado na Figura 30.

Figura 28 – Posições possíveis para a restrição topológica de adjacência.

Fonte: Adaptado de Choi et al. (2017).

Figura 29 – Posições possíveis para a restrição topológica de alinhamento

Fonte: Adaptado de Choi et al. (2017).

3.1.3 MODELAGEM COMPUTACIONAL DA PRIMEIRA ETAPA

Utilizando a linguagem de programação Python, implementou-se um GA baseado na ordem utilizando o *crossover* PMX e a mutação de tipo swapping para otimizar a função de custos MHC (equação 02), considerando como dados de entrada: as informações de fluxo de materiais (Quadro 4); restrições de alinhamento e adjacência (Quadro 5); além dos departamentos com posição fixa.

A Figura 31 especifica o fluxograma do GA implementado. A partir dos dados de entrada, o programa é inicializado, solicitando ao usuário o número de indivíduos da população inicial; sendo assim, gera-se a população inicial aleatória, como a apresentada na Figura 32. O cromossomo é preenchido assinando primeiramente os departamentos com localização fixa aos genes fixos do cromossomo, no caso, os departamentos 19, 20, 18, 17, 16 e 15.

Figura 30 – Fluxograma do GA para a primeira etapa – Otimização da topologia

Fonte: Própria autoria.

Figura 31 – Exemplo de indivíduo aleatório que forma parte da população inicial

Fonte: Própria autoria.

3.2 OTIMIZAÇÃO GEOMÉTRICA – SEGUNDA ETAPA

3.2.1 MODELO MATEMÁTICO DE CHOI ET AL DA SEGUNDA ETAPA

Os autores propõem uma função de otimização multiobjetivo F (equação 07), que procura otimizar os custos MHC, as áreas, a razão de aspecto e penalizar o não cumprimento dos requisitos de alinhamento e adjacência. Como dados de entrada, nesse modelo são necessários os dados de custo, fluxo de materiais, requerimentos de áreas e razão de aspecto ideais, as restrições de alinhamento e adjacência, e, finalmente, o vetor da solução ótima da primeira etapa.

O primeiro termo da equação 07 descreve o custo de manuseio de materiais MHC; o segundo representa as áreas dos departamentos; o terceiro termo corresponde à razão de aspecto (RA); o quarto reflete o requisito de adjacência (AD); e o quinto, o requisito de alinhamento (AL). Cada um desses termos é multiplicado por um peso ω_i , dado que se trata de uma otimização multiobjetivo, utilizando o método das somas ponderadas.

$$\text{Min } F^* = \omega_1 \cdot \sum \text{Fluxo} + \omega_2 \cdot \sum \text{Área atingida} + \omega_3 \cdot \sum \text{RA atingida} + \omega_4 \cdot \sum AD + \omega_5 \cdot \sum AL$$

Modelo Matemático:

$$\begin{aligned} \text{Minimizar } F = & \left[\omega_1 \cdot \sum_{i \in D} \sum_{j \in D} f_{i,j}^{tp} \cdot c_{i,j}^{tp} \cdot d_{i,j}^{tv} + \omega_2 \cdot \sum_{d \in D} \frac{|ar_d^{gp} - ar_d^{gv}|}{ar_d^{gp}} + \omega_3 \right. \\ & \cdot \sum_{d \in D} |as_d^{gp} - as_d^{gv}| + \omega_4 \cdot \sum_{i=1}^{N-1} \sum_{j=i+1}^N ad_{i,j}^{gp} (1 - ad_{i,j}^{gv}) + \omega_4 \\ & \left. \cdot \sum_{i=1}^{N-1} \sum_{j=i+1}^N al_{i,j}^{gp} (1 - al_{i,j}^{gv}) \right] \end{aligned} \quad (07)$$

Sujeito a

$$d_{i,j}^{gv} = |cx_i^{gv} - cx_j^{gv}| + |cy_i^{gv} - cy_j^{gv}| \quad ; i, j \in D \quad (08)$$

$$ar_d^{gv} = l_d^{gv} \cdot s_d^{gv} \quad ; d \in D \quad (09)$$

$$as_d^{gv} = \frac{l_d^{gv}}{s_d^{gv}} \quad ; d \in D \quad (10)$$

Onde,

Conjuntos:

D: Conjunto dos departamentos d .

Parâmetros:

$f_{i,j}^{tp}$: Fluxo de materiais entre os departamentos i e j ;

$c_{i,j}^{tp}$: Custo de transporte por distância percorrida e por unidade de volume entre os departamentos i e j ;

ar_d^{gp} : Área requerida, dada no Quadro 3.

ar_d^{gp} : Razão de aspecto requerida, dada no Quadro 3.

$aj_{i,j}^{gp}$: 1, se o departamento i tem adjacência com o departamento j , senão, 0;

$al_{i,j}^{gp}$: 1, se o departamento i tem alinhamento com o departamento j , senão, 0;

ω_1 Fator de peso para os MHC;

ω_2 Fator de peso para a satisfação de área;

ω_3 Fator de peso para a satisfação de forma (razão de aspecto);

ω_4 Fator de peso para os requerimentos de adjacência;

ω_5 Fator de peso para os requerimentos de alinhamento;

N Número de departamentos.

Variáveis:

ar_d^{gv} Área atingida pelo departamento d ;

$aj_{i,j}^{gv}$ 1, se o departamento i tem adjacência com o departamento j , senão, 0;

$al_{i,j}^{gv}$ 1, se o departamento i tem alinhamento com o departamento j , senão, 0;

$d_{i,j}^{gv}$ Distância Manhattan entre os departamentos i e j ;

l_d^{gv} Eixo maior do departamento d ;

- s_d^{gv} Eixo menor do departamento d ;
- as_d^{gv} Razão de aspecto atingida pelo departamento d ;
- cx_d^{gv} Centróide x do departamento d ;
- cy_d^{gv} Centróide y do departamento d ;
- sel_j^{gv} Probabilidade de selecionar o departamento d ;
- p^{gv} Função de penalidade por não cumprimento de adjacência ou alinhamento.

As letras superiores " gp " e " gv " indicam "parâmetro de geometria" e "variável de geometria".

3.2.2 DESCRIÇÃO DA OTIMIZAÇÃO GEOMÉTRICA

Para representar o terreno do estaleiro e proceder com a otimização geométrica dos departamentos, os autores utilizaram uma grade virtual de 46x22 células, apresentada na Figura 37. As dimensões dessa grade estariam referenciadas ao terreno real do estaleiro, sendo que a célula da grade representa uma unidade de crescimento para o modelo em cada iteração (ver Figura 34); a célula deveria conter pelo menos o tamanho do sub-bloco padrão máximo produzido pelo estaleiro. Isso reflete a preocupação dos autores com a aplicabilidade do modelo na vida real.

Figura 33 – Grade de 46x22 células e centroide dos departamentos de Choi et al.

Fonte: CHOI et al. (2017).

Na sequência, posiciona-se inicialmente o centroide de cada departamento em uma célula da grade, para depois dar início ao processo de crescimento. Dessa maneira, todos os departamentos partem da mesma área inicial correspondente a uma célula (ver Figura 37). A partir dessa distribuição dos departamentos na grade, inicia-se o processo de otimização geométrica usando um algoritmo de crescimento estocástico, no qual os departamentos crescem seguindo as probabilidades de crescimento ou decrescimento apresentadas na seção 3.2.3 (ver Figura 35).

Figura 34 – Processo de crescimento por iteração do departamento 3

Fonte: Própria autoria.

3.2.3 MODELAGEM COMPUTACIONAL DA SEGUNDA ETAPA

O fluxo do processo para essa etapa de otimização da geometria do layout é apresentado na Figura 36, comentada a seguir.

Figura 35 – Fluxo do algoritmo de crescimento estocástico para a otimização geométrica

Fonte: Própria autoria.

Loop da Geração de indivíduos:

Diferentemente da primeira etapa, onde os indivíduos da população eram gerados aleatoriamente, nessa etapa, utiliza-se um algoritmo de crescimento estocástico baseado nos autores Choi et al. (2017). Observa-se que para cada *loop* do algoritmo obtém-se uma solução de layout.

Loop estágio 01: seleção

Nesse passo, decide-se qual departamento selecionar para continuar com o processo iterativo. Adotou-se para esse fim o critério da roleta, também proposto por Choi et al. (2017). Nesse critério, seleciona-se o departamento com maior probabilidade de ser eleito (P_{sel}), indicado pela equação 11.

$$P_{sel} = \frac{AR_i - AA_i}{\sum AR_i - AA_i} \quad (11)$$

Dessa maneira, o departamento com maior área requerida teria maior probabilidade de ser eleito, já que todos os departamentos têm inicialmente a mesma área; isto é, todos os departamentos têm só uma célula atribuída.

Loop estágio 2: crescimento ou decrescimento

O estágio 2 do *loop* decide se o departamento i dever crescer ou encolher, em função da área requerida pelo departamento i (AR_i) e a área atingida pelo departamento i (AA_i), segundo o critério:

$$AR_i - AA_i > 0 \rightarrow \text{crescimento}$$

$$AR_i - AA_i < 0 \rightarrow \text{encolhimento}$$

Voltando para o exemplo do departamento 15, supondo que ele atingisse uma área de 100 unidades até o momento, sendo que precisaria de 144. Nesse caso, o código escolheria crescimento, por conta de que a diferença entre estas duas áreas é maior que zero ($144 - 100 > 0$). Pelo contrário, se o departamento 15 atingisse uma área maior que a necessária, por exemplo 200 unidades, o código escolheria encolhimento, por ter uma diferença menor que zero ($144 - 200 < 0$).

Loop estágio 3: direção de crescimento

No estágio 3 do *loop* determina-se a direção do crescimento para cada departamento. Para essa decisão, utilizou-se os critérios propostos por Choi et al. (2017) para determinar a direção do crescimento em função da relação entre a razão de aspecto requerido (RA_r) e a razão de aspecto atingido (RA_a). Observa-se que a razão de aspecto (requerido ou atingido) é sempre calculada pela relação entre a dimensão do eixo maior e o eixo menor,

independentemente da orientação vertical ou horizontal do departamento. Os números (1), (2), (3) e (4) indicam as possíveis direções de crescimento (ver Figura 37).

Figura 36 – Orientação horizontal e vertical dos departamentos

Fonte: Adaptado de Choi et al. (2017).

A Figura 38 apresenta os critérios que determinam se a relação da razão de aspecto requerida *versus* a atingida (RAr/RAa), por cada departamento, enquadra-se em algum dos intervalos *a*, *b*, *c*, *d* ou *e* apresentados nos Quadros 6 e 7.

Figura 37 – Razão de aspecto requerida vs. atingida

$$\frac{RAr}{RAa} = \left\{ \begin{array}{l} \leq 0.5 \\ < 1 \\ = 1 \\ > 1 \\ \geq 2 \end{array} \right.$$

Fonte: Adaptado de Choi et al (2017) e Nick (2008).

Quadro 6 – Probabilidades para escolha da direção de crescimento

Direção do Crescimento	RAr/RAa				
	a	b	c	d	e
	≤ 0.5	< 1	$= 1$	> 1	≥ 2
Direção (1)	(10+l) %	(20+l) %	(25+l) %	(30+l) %	(40+l) %
Direção (2)	(10-l) %	(20-l) %	(25-l) %	(30-l) %	(40-l) %
Direção (3)	(40+s) %	(30+s) %	(25+s) %	(20+s) %	(10+s) %
Direção (4)	(40-s) %	(30-s) %	(25-s) %	(20-s) %	(10-s) %

Fonte: Adaptado de Choi et al (2017) e Nick (2008).

Quadro 7 – Probabilidades para escolha da direção de encolhimento

Direção do Encolhimento	RAr/RAa				
	a	b	c	d	e
	≤ 0.5	< 1	$= 1$	> 1	≥ 2
Direção (1)	$(40+l) \%$	$(30+l) \%$	$(25+l) \%$	$(20+l) \%$	$(10+l) \%$
Direção (2)	$(40-l) \%$	$(30-l) \%$	$(25-l) \%$	$(20-l) \%$	$(10-l) \%$
Direção (3)	$(10+s) \%$	$(20+s) \%$	$(25+s) \%$	$(30+s) \%$	$(40+s) \%$
Direção (4)	$(10-s) \%$	$(20-s) \%$	$(25-s) \%$	$(30-s) \%$	$(40-s) \%$

Fonte: Adaptado de Choi et al (2017) e Nick (2008).

As probabilidades apresentadas nos Quadros 6 e 7 foram adaptadas de Nick (2008) e Choi et al., (2017). A adaptação foi feita com a finalidade de ampliar a variação das probabilidades de crescimento (ou encolhimento) e induzir essa escolha na direção que satisfaz a relação de aspecto necessária. Porém, os autores destacam que as probabilidades apresentadas nos Quadros 6 e 7 induzem o crescimento (ou encolhimento) na direção que satisfaz a relação de aspecto necessário; porém, os autores não mencionam detalhes desse procedimento. Para superar esse ponto, nesta dissertação propõe-se o uso da distância *Chebyshev*, apresentada na Figura 39, para a escolha da direção de crescimento ou encolhimento dos departamentos com requisitos de alinhamento ou adjacência.

Figura 38 – Os departamentos se aproximam pela distância Chebyshev

Fonte: Própria autoria.

Para um melhor entendimento desse passo, utilizemos como exemplo o departamento 13, que tem uma razão de aspecto requerida (RAr) igual a 4. Supondo que no processo de crescimento ele atingisse, até o momento uma razão de aspecto de 3 (RAa), a relação das formas requerida e atingida estaria dada por: $RAr_{13}/RAa_{13} = 4/3 = 1.33$. Então, esse departamento entraria no intervalo d (>1) do Quadro 6. No Quadro 6, a coluna d apresenta as

probabilidades de crescimento nas quatro direções possíveis: (1), (2), (3) e (4); mas observa-se que as direções do eixo mais longo, (1) e (2) têm mais probabilidade do que as direções do eixo mais curto, (3) e (4). Dessa maneira, o departamento 13 cresceria na direção do eixo mais longo (1) ou (2), até atingir a razão de aspecto requerida (ver Figura 40).

Figura 39 – Exemplo do departamento 13, RA atingida/ RA requerida

Fonte: Própria autoria.

Loop estágio 4: crescimento

Nesse estágio, as áreas dos departamentos do layout crescem na direção dada pelas probabilidades dos Quadros 6 e 7 e o valor das variáveis l (*long direction*) e s (*short direction*). As variáveis l e s , segundo os autores, controlam a direção do crescimento dos departamentos que possuem restrições de alinhamento e adjacência por meio dos operadores chamados *PUSH* e *ATTRACT*. Para modelar esse *loop*, nesta dissertação representa-se a área de cada departamento utilizando a representação clássica de espaços planos proposta por Micheleck et al. (2002), conforme a Figura 41.

Figura 40 – Definição de unidade do espaço plano

Fonte: MICHAELEK et al. (2002).

Observa-se nessa figura que o departamento está definido pelas coordenadas em X e Y do seu centroide na grade geométrica: cx , cy ; além disso, note-se as variáveis N_i , S_i , E_i e W_i

que determinam o limite do departamento ao norte, sul, leste e oeste, partindo sempre do centroide cx e cy .

Supondo que o departamento i e o departamento j têm requisitos de adjacência ou alinhamento, para o departamento i crescer na direção do departamento j e atingir alinhamento ou adjacência, os autores propuseram o operador *PUSH*, ou empurrar, e *ATTRACT*, ou atrair (ver Figura 42).

Figura 41 – Operador PUSH ou empurrar

Fonte: CHOI et al. (2017).

Esses operadores atuam sempre que o departamento escolhido tenha requisito de alinhamento ou adjacência com seu departamento vizinho; com isso, esse departamento cresce na direção escolhida e provoca o encolhimento dos vizinhos (CHOI; KIM; CHUNG, 2017).

Isso quer dizer que, toda vez que o código detecta que o departamento i e j tem requisito de alinhamento ou adjacência, calcula-se a distância Chebyshev entre eles como uma maneira de determinar o encostamento entre eles. Para esse fim, os operadores l e s do Quadro 6 e Quadro 7 podem assumir somente três valores possíveis para cada direção: +10, -10 e zero.

$$\text{Se } da_{ij} > 0 \rightarrow l, s = \begin{cases} -10 \\ +10 \\ 0 \end{cases}$$

Esses operadores ajudam a aumentar a possibilidade do departamento i crescer na direção do departamento j com quem tem alinhamento ou adjacência; e o código calcula a distância Chebyshev entre eles e aplica os operadores l ou s seguindo as probabilidades do Quadro 6 ou do Quadro 7. Dessa maneira, dispõe-se de uma maior probabilidade do departamento i crescer na direção do departamento j . (ver Figura 43).

Figura 42 – Valores dos operadores l e s para cada direção de crescimento

Fonte: Adaptado de Choi et al. (2017).

Para um melhor entendimento do mecanismo utilizado, considere o exemplo da Figura 44. Supõe-se nesse exemplo que os departamentos a e b têm requisito de alinhamento ou adjacência; nessa iteração, o departamento a é escolhido para crescer e ele tem uma relação de aspecto requerida igual a quatro. Como o departamento a tem uma relação de aspecto atingida igual a 3 (12 retângulos/ 4 retângulos), então a relação de razão de aspecto requerida sobre atingida é: $RAr_a/RAa_a = 4/3 = 1,33$; portanto, o valor obtido enquadra na coluna d do Quadro 6. Infere-se, então, que a direção correta para o crescimento do departamento a deve ser a direção onde se encontra b , isto é, a direção (2).

Figura 43 – Exemplo de escolha da direção de crescimento entre os departamentos a e b

Fonte: Própria autoria.

Seguindo esse critério de crescimento na direção (2), o algoritmo calcula a distância Chebyshev entre os departamentos a e b e verifica se ela é maior que zero. No exemplo mostrado na Figura 47, o valor da distância Chebyshev é maior que zero ($\text{Max}(9,3)=9$), de modo que, para valores da distância Chebyshev maiores que zero, o operador l assume um valor de -10 :

$$\text{Direção (1): } (30 + l) \% = (30 + (-10)) \% = 20\%$$

$$\text{Direção (2): } (30 - l) \% = (30 - (-10)) \% = 40\%$$

O processo somente se detém quando o valor da distância Chebyshev atinge o valor zero, o que isso significa que os departamentos se alinham. Desse modo, destaca-se que para todos os casos das colunas do Quadro 8 (réplica do Quadro 6), a probabilidade de escolha da direção (2) será sempre maior que a probabilidade de selecionar a direção (1), independentemente da coluna.

Quadro 8 – Probabilidade de escolha da direção (2) maior que a direção (1), $P(2) > P(1)$

Direção do Crescimento	RAr/RAa				
	a	b	c	d	e
	≤ 0.5	< 1	$= 1$	> 1	≥ 2
Direção (1)	0 %	10 %	15 %	20 %	30 %
Direção (2)	20 %	30 %	35 %	40 %	50 %
Direção (3)	$(40+s) \%$	$(30+s) \%$	$(25+s) \%$	$(20+s) \%$	$(10+s) \%$
Direção (4)	$(40-s) \%$	$(30-s) \%$	$(25-s) \%$	$(20-s) \%$	$(10-s) \%$

Fonte: Própria autoria.

Generalizando esse critério para as quatro direções (1), (2), (3) e (4), tem-se:

- Direção(1): *distância chebyshev* $> 0 \rightarrow l = -10$
- Direção(2): *distância chebyshev* $> 0 \rightarrow l = +10$
- Direção(3): *distância chebyshev* $> 0 \rightarrow s = -10$
- Direção(4): *distância chebyshev* $> 0 \rightarrow s = +10$

A Figura 45 resume o *loop* de seleção dos indivíduos para constituir a população inicial.

Figura 44 – Loop do crescimento estocástico dos departamentos do layout

Fonte: Própria autoria.

Figura 46 – Exemplo layout B

[illegible]

Fonte: Própria autoria.

A Figura 48 explica o mecanismo do *crossover* ponto-linha vertical ou horizontal aplicado no centroide dos departamentos 1, 3 e 4 dos layouts *A* e *B*. No caso do departamento 1, o *crossover* ponto-linha horizontal é aplicado no centroide do layout *A*, sendo o filho definido pela junção de todos os retângulos do norte de *A*, e completando com o número de retângulos em direção sul do layout *B*; sendo eliminada a fila do centroide dos dois pais, e calculado o novo centroide e os valores na direção leste e oeste para o filho. Da mesma forma, no departamento 3 aplicou-se o *crossover* ponto-linha vertical; isto é: todos os retângulos ao leste do layout pai *A* (em verde), e completando com o número de retângulos ao oeste do layout *B*, no caso dois retângulos. Depois, calcula o novo centroide da área e as coordenadas leste e oeste. O departamento 4 apresenta o *crossover* ponto-linha horizontal similar ao departamento 1.

Figura 47 – Recombinação dos departamentos 1, 3 e 4 com *crossover* ponto-linha

Dep	Pai 1: Layout A	Pai 2: Layout B	Filho																																																																															
1	<table><tr><td>N</td><td>1</td></tr><tr><td>S</td><td>1</td></tr><tr><td>E</td><td>1</td></tr><tr><td>W</td><td>1</td></tr></table> <table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>+</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	N	1	S	1	E	1	W	1	1	1	1	+	1	1	1	1	1	<table><tr><td>N</td><td>2</td></tr><tr><td>S</td><td>2</td></tr><tr><td>E</td><td>0</td></tr><tr><td>W</td><td>1</td></tr></table> <table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td></tr></table>	N	2	S	2	E	0	W	1	1	1	1	1	1	1	1	1	<table><tr><td>N</td><td>2</td></tr><tr><td>S</td><td>2</td></tr><tr><td>E</td><td>1</td></tr><tr><td>W</td><td>1</td></tr></table> <table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	N	2	S	2	E	1	W	1	1	1	1	1	1	1	1	1	1																													
N	1																																																																																	
S	1																																																																																	
E	1																																																																																	
W	1																																																																																	
1	1	1																																																																																
+	1	1																																																																																
1	1	1																																																																																
N	2																																																																																	
S	2																																																																																	
E	0																																																																																	
W	1																																																																																	
1	1																																																																																	
1	1																																																																																	
1	1																																																																																	
1	1																																																																																	
N	2																																																																																	
S	2																																																																																	
E	1																																																																																	
W	1																																																																																	
1	1	1																																																																																
1	1	1																																																																																
1	1	1																																																																																
3	<table><tr><td>N</td><td>2</td></tr><tr><td>S</td><td>1</td></tr><tr><td>E</td><td>3</td></tr><tr><td>W</td><td>2</td></tr></table> <table><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr></table>	N	2	S	1	E	3	W	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	<table><tr><td>N</td><td>1</td></tr><tr><td>S</td><td>1</td></tr><tr><td>E</td><td>2</td></tr><tr><td>W</td><td>2</td></tr></table> <table><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr></table>	N	1	S	1	E	2	W	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	<table><tr><td>N</td><td>2</td></tr><tr><td>S</td><td>2</td></tr><tr><td>E</td><td>1</td></tr><tr><td>W</td><td>2</td></tr></table> <table><tr><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td></tr><tr><td>3</td><td>3</td><td>1</td><td>3</td></tr><tr><td>3</td><td>3</td><td>3</td><td>3</td></tr></table>	N	2	S	2	E	1	W	2	3	3	3	3	3	3	3	3	3	3	1	3	3	3	3	3
N	2																																																																																	
S	1																																																																																	
E	3																																																																																	
W	2																																																																																	
3	3	3	3	3	3																																																																													
3	3	3	3	3	3																																																																													
3	3	3	3	3	3																																																																													
3	3	3	3	3	3																																																																													
N	1																																																																																	
S	1																																																																																	
E	2																																																																																	
W	2																																																																																	
3	3	3	3	3																																																																														
3	3	3	3	3																																																																														
3	3	3	3	3																																																																														
N	2																																																																																	
S	2																																																																																	
E	1																																																																																	
W	2																																																																																	
3	3	3	3																																																																															
3	3	3	3																																																																															
3	3	1	3																																																																															
3	3	3	3																																																																															
4	<table><tr><td>N</td><td>2</td></tr><tr><td>S</td><td>1</td></tr><tr><td>E</td><td>1</td></tr><tr><td>W</td><td>2</td></tr></table> <table><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr></table>	N	2	S	1	E	1	W	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	<table><tr><td>N</td><td>1</td></tr><tr><td>S</td><td>0</td></tr><tr><td>E</td><td>3</td></tr><tr><td>W</td><td>3</td></tr></table> <table><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr></table>	N	1	S	0	E	3	W	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	<table><tr><td>N</td><td>1</td></tr><tr><td>S</td><td>0</td></tr><tr><td>E</td><td>2</td></tr><tr><td>W</td><td>2</td></tr></table> <table><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>1</td><td>4</td><td>4</td></tr></table>	N	1	S	0	E	2	W	2	4	4	4	4	4	4	4	1	4	4															
N	2																																																																																	
S	1																																																																																	
E	1																																																																																	
W	2																																																																																	
4	4	4	4																																																																															
4	4	4	4																																																																															
4	4	4	4																																																																															
4	4	4	4																																																																															
N	1																																																																																	
S	0																																																																																	
E	3																																																																																	
W	3																																																																																	
4	4	4	4	4	4	4																																																																												
4	4	4	4	4	4	4																																																																												
N	1																																																																																	
S	0																																																																																	
E	2																																																																																	
W	2																																																																																	
4	4	4	4	4																																																																														
4	4	1	4	4																																																																														

Fonte: Própria autoria.

A mutação *swapping* foi aplicada intercambiando aleatoriamente os valores das coordenadas *N*, *S*, *E*, e *W* entre dois departamentos; isto é, atribuindo o valor de *N* a *E* e vice-versa, e aplicando esse operador a 10% da população (ver Figura 49).

Figura 48 – Swapping do departamento 4

Dep	Pai 1: Layout A										Filho																																								
4	<i>N</i> 2		<table><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td></tr></table>								4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	<i>N</i> 1		<table><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>4</td><td>4</td><td>4</td><td>4</td><td>4</td></tr></table>								4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	4	4	4	4																																															
	4	4	4	4																																															
	4	4	4	4																																															
	4	4	4	4																																															
4	4	4	4	4																																															
4	4	4	4	4																																															
4	4	4	4	4																																															
<i>S</i> 1										<i>S</i> 1																																									
<i>E</i> 1										<i>E</i> 2																																									
<i>W</i> 2										<i>W</i> 2																																									

Fonte: Própria autoria.

A seleção, por outro lado, é realizada escolhendo aleatoriamente dois pais da população inicial em cada geração; sendo assim, gera-se um filho por geração por meio do *crossover* descrito acima. A modelagem computacional dessa terceira etapa é apresentada na seção 3.3.2.

3.3.2 MODELAGEM DA TERCEIRA ETAPA PROPOSTA

Como mencionado no capítulo 3, nessa etapa, propõe-se implementar na linguagem Python um GA geral com os operadores do tipo ponto-linha para o *crossover* e *swapping* para a mutação, descritos em 3.3.1. A Figura 50 apresenta o fluxograma desse algoritmo, sendo o critério de parada o atingimento de 500 iterações.

O algoritmo começa com a leitura dos pais, que são os nove melhores layouts obtidos na segunda etapa e apresentados no apêndice D, além dos requisitos de população máxima e número de gerações. Posteriormente, seleciona-se dois pais seguindo o método da roleta; isto quer dizer que aqueles layouts com melhor fitness teriam mais probabilidade de serem selecionados para o cruzamento; em seguida, o cruzamento é executando gerando um filho a cada geração, sendo que o processo continua até atingir o número de gerações requerido. Ressalta-se que, quando atingir a população máxima requerida, o indivíduo da população com menor fitness é eliminado.

Figura 49 – Fluxograma do GA da terceira etapa proposta

Fonte: Própria autoria.

4 RESULTADOS OBTIDOS

4.1 RESULTADOS DA PRIMEIRA ETAPA – OTIMIZAÇÃO TOPOLÓGICA

Foram executados cem testes do GA da primeira etapa, sendo salvo o melhor indivíduo encontrado em cada teste. O apêndice B apresenta o resultado de cada teste. A Figura 51 apresenta os valores da função *fitness* para cada sequência, destacando a maneira de comparação e em vermelho, o valor da função objetivo reportada por Choi et al., (2017). O tempo de execução não foi considerado nesta dissertação por não ser parte dos experimentos computacionais descritos pelos autores.

Figura 50 – Valor da função *fitness* de cada teste para a primeira etapa

Fonte: Própria autoria.

O intervalo de confiança atingido é apresentado no Quadro 9, destacando-se que o intervalo de confiança não cobriu o valor de 11.816 obtido pelos autores.

Quadro 9 – Intervalo de confiança – testes da primeira etapa

Número de testes	100
Média	12.178
Desvio padrão	337,412
Nível de confiança	95%
Margem de erro	66,13
Limite inferior	12.111,87
Limite superior	12.244,13

Fonte: Própria autoria.

Obtiveram-se apenas onze soluções que atingiram um melhor valor da função *fitness* do que relatado pelos autores Choi et al. Essas soluções são exibidas no Quadro 10 a seguir:

Quadro 10 – Melhores soluções obtidas - primeira etapa e solução de Choi et al.

Sequência reportada por Choi et al:																			<i>Fitness</i>																				
19	20	8	11	12	18	5	9	10	13	3	4	7	14	17	2	1	6	15	16	11.816																			
<table><tr><td>2</td><td>1</td><td>6</td><td>15</td><td>16</td></tr><tr><td>3</td><td>4</td><td>7</td><td>14</td><td>17</td></tr><tr><td>18</td><td>5</td><td>9</td><td>10</td><td>13</td></tr><tr><td>19</td><td>20</td><td>8</td><td>11</td><td>12</td></tr></table>																			2	1	6	15	16	3	4	7	14	17	18	5	9	10	13	19	20	8	11	12	
2	1	6	15	16																																			
3	4	7	14	17																																			
18	5	9	10	13																																			
19	20	8	11	12																																			
Sequência obtida 01:																			<i>Fitness</i>																				
19	20	7	4	3	18	13	9	5	6	1	11	10	8	17	2	12	14	15	16	10.944																			
<table><tr><td>2</td><td>12</td><td>14</td><td>15</td><td>16</td></tr><tr><td>13</td><td>11</td><td>1</td><td>8</td><td>17</td></tr><tr><td>18</td><td>10</td><td>9</td><td>5</td><td>6</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																			2	12	14	15	16	13	11	1	8	17	18	10	9	5	6	19	20	7	4	3	
2	12	14	15	16																																			
13	11	1	8	17																																			
18	10	9	5	6																																			
19	20	7	4	3																																			
Sequência obtida 02:																			<i>Fitness</i>																				
19	20	7	4	3	18	10	9	5	1	13	11	8	6	17	2	12	14	15	16	11.074																			
<table><tr><td>2</td><td>12</td><td>14</td><td>15</td><td>16</td></tr><tr><td>13</td><td>11</td><td>8</td><td>6</td><td>17</td></tr><tr><td>18</td><td>10</td><td>9</td><td>5</td><td>1</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																			2	12	14	15	16	13	11	8	6	17	18	10	9	5	1	19	20	7	4	3	
2	12	14	15	16																																			
13	11	8	6	17																																			
18	10	9	5	1																																			
19	20	7	4	3																																			

Continua Quadro 10

Continuação Quadro 10

Sequência obtida 03:																				Fitness																				
19	20	13	11	6	18	8	10	12	1	2	5	9	14	17	3	4	7	15	16	11.109																				
<table><tr><td>3</td><td>4</td><td>7</td><td>15</td><td>16</td></tr><tr><td>2</td><td>5</td><td>9</td><td>14</td><td>17</td></tr><tr><td>18</td><td>8</td><td>10</td><td>12</td><td>1</td></tr><tr><td>19</td><td>20</td><td>13</td><td>11</td><td>6</td></tr></table>																				3	4	7	15	16	2	5	9	14	17	18	8	10	12	1	19	20	13	11	6	
3	4	7	15	16																																				
2	5	9	14	17																																				
18	8	10	12	1																																				
19	20	13	11	6																																				
Sequência obtida 04:																				Fitness																				
19	20	7	4	3	18	10	9	5	1	2	11	8	6	17	13	12	14	15	16	11.445																				
<table><tr><td>13</td><td>12</td><td>14</td><td>15</td><td>16</td></tr><tr><td>2</td><td>11</td><td>8</td><td>6</td><td>17</td></tr><tr><td>18</td><td>10</td><td>9</td><td>5</td><td>1</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																				13	12	14	15	16	2	11	8	6	17	18	10	9	5	1	19	20	7	4	3	
13	12	14	15	16																																				
2	11	8	6	17																																				
18	10	9	5	1																																				
19	20	7	4	3																																				
Sequência obtida 05:																				Fitness																				
19	20	7	4	3	18	13	9	5	8	2	11	10	6	17	1	12	14	15	16	11.484																				
<table><tr><td>1</td><td>12</td><td>14</td><td>15</td><td>16</td></tr><tr><td>2</td><td>11</td><td>10</td><td>6</td><td>17</td></tr><tr><td>18</td><td>13</td><td>9</td><td>5</td><td>8</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																				1	12	14	15	16	2	11	10	6	17	18	13	9	5	8	19	20	7	4	3	
1	12	14	15	16																																				
2	11	10	6	17																																				
18	13	9	5	8																																				
19	20	7	4	3																																				
Sequência obtida 06:																				Fitness																				
19	20	6	3	2	18	13	7	4	1	11	10	9	5	17	12	14	8	15	16	11.674																				
<table><tr><td>12</td><td>14</td><td>8</td><td>15</td><td>16</td></tr><tr><td>11</td><td>10</td><td>9</td><td>5</td><td>17</td></tr><tr><td>18</td><td>13</td><td>7</td><td>4</td><td>1</td></tr><tr><td>19</td><td>20</td><td>6</td><td>3</td><td>2</td></tr></table>																				12	14	8	15	16	11	10	9	5	17	18	13	7	4	1	19	20	6	3	2	
12	14	8	15	16																																				
11	10	9	5	17																																				
18	13	7	4	1																																				
19	20	6	3	2																																				
Sequência obtida 07:																				Fitness																				
19	20	7	4	3	18	6	9	5	1	12	11	10	8	17	2	13	14	15	16	11.685																				
<table><tr><td>2</td><td>13</td><td>14</td><td>15</td><td>16</td></tr><tr><td>12</td><td>11</td><td>10</td><td>8</td><td>17</td></tr><tr><td>18</td><td>6</td><td>9</td><td>5</td><td>1</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																				2	13	14	15	16	12	11	10	8	17	18	6	9	5	1	19	20	7	4	3	
2	13	14	15	16																																				
12	11	10	8	17																																				
18	6	9	5	1																																				
19	20	7	4	3																																				
Sequência obtida 08:																				Fitness																				
19	20	7	4	3	18	1	9	5	6	2	13	10	8	17	12	11	14	15	16	11.709																				
<table><tr><td>12</td><td>11</td><td>14</td><td>15</td><td>16</td></tr><tr><td>2</td><td>13</td><td>10</td><td>8</td><td>17</td></tr><tr><td>18</td><td>1</td><td>9</td><td>5</td><td>6</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																				12	11	14	15	16	2	13	10	8	17	18	1	9	5	6	19	20	7	4	3	
12	11	14	15	16																																				
2	13	10	8	17																																				
18	1	9	5	6																																				
19	20	7	4	3																																				

Continua Quadro 10

Continuação Quadro 10

Sequência obtida 09:																			<i>Fitness</i>																				
19	20	6	3	4	18	13	8	5	7	1	11	10	9	17	2	12	14	15	16	11.713																			
<table><tr><td>2</td><td>12</td><td>14</td><td>15</td><td>16</td></tr><tr><td>1</td><td>11</td><td>10</td><td>9</td><td>17</td></tr><tr><td>18</td><td>13</td><td>8</td><td>5</td><td>7</td></tr><tr><td>19</td><td>20</td><td>6</td><td>3</td><td>4</td></tr></table>																			2	12	14	15	16	1	11	10	9	17	18	13	8	5	7	19	20	6	3	4	
2	12	14	15	16																																			
1	11	10	9	17																																			
18	13	8	5	7																																			
19	20	6	3	4																																			
Sequência obtida 10:																			<i>Fitness</i>																				
19	20	7	4	3	18	10	9	5	8	1	11	12	6	17	2	13	14	15	16	11.780																			
<table><tr><td>2</td><td>13</td><td>14</td><td>15</td><td>16</td></tr><tr><td>1</td><td>11</td><td>12</td><td>6</td><td>17</td></tr><tr><td>18</td><td>10</td><td>9</td><td>5</td><td>8</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																			2	13	14	15	16	1	11	12	6	17	18	10	9	5	8	19	20	7	4	3	
2	13	14	15	16																																			
1	11	12	6	17																																			
18	10	9	5	8																																			
19	20	7	4	3																																			

Continua Quadro 10

Continuação Quadro 10

Sequência obtida 11:																			<i>Fitness</i>																				
19	20	7	4	3	18	11	9	5	6	1	12	10	8	17	2	13	14	15	16	11.782																			
<table><tr><td>2</td><td>13</td><td>14</td><td>15</td><td>16</td></tr><tr><td>1</td><td>12</td><td>10</td><td>8</td><td>17</td></tr><tr><td>18</td><td>11</td><td>9</td><td>5</td><td>6</td></tr><tr><td>19</td><td>20</td><td>7</td><td>4</td><td>3</td></tr></table>																			2	13	14	15	16	1	12	10	8	17	18	11	9	5	6	19	20	7	4	3	
2	13	14	15	16																																			
1	12	10	8	17																																			
18	11	9	5	6																																			
19	20	7	4	3																																			

Fonte: Própria autoria

Dessas soluções, ressaltam-se as sequências 2, 3, 4 e 9, devido ao fato de que não apresentam contra-fluxo em comparação com as outras soluções obtidas; sendo assim, escolhe-se dentre elas a sequência 03 para ser o dado de entrada da segunda etapa, destacando que qualquer delas poderia ser escolhida para gerar geometrias válidas.

4.2 RESULTADOS DA SEGUNDA ETAPA – OTIMIZAÇÃO GEOMÉTRICA

Nessa etapa, foram executados 70 testes com 500 iterações. O vetor solução da etapa 1 dado pela sequência 3 (ver Quadro 11) foi escolhido como dado de entrada dessa segunda etapa. O apêndice D apresenta os nove melhores layouts solução que serão selecionadas para serem os pais do GA da terceira etapa. O valor da função *fitness* atingido por cada teste se apresenta no Quadro 12.

Quadro 11 – Sequência obtida 03

19	20	13	11	6	18	8	10	12	1	2	5	9	14	17	3	4	7	15	16
----	----	----	----	---	----	---	----	----	---	---	---	---	----	----	---	---	---	----	----

Fonte: Própria autoria.

Quadro 12 – Valores da Função Fitness obtida nos testes

Teste	Fitness
2	720,41
28	702,37
35	957,40
50	934,42
51	535,03
52	731,92
56	709,79
58	1280,51
59	952,19

Fonte: Própria autoria.

Ressalta-se, nessa etapa, que os critérios de avaliação da função multiobjetivo da equação 07 são: custos MHC; área atingida; razão de aspecto atingido e cumprimento de alinhamento e adjacência. A Figura 52 apresenta o melhor layout obtido segundo o critério de atingimento da área requerida, sendo ele o layout solução do teste 51 (ver apêndice C). No Quadro 13 apresenta-se o resultado obtido para os nove testes para esse critério.

Figura 51 – Melhor layout solução - Teste 51 – atingimento de área requerida

Fonte: Própria autoria.

Quadro 13 – Percentagem de área atingida por cada teste

Área Atingida		
Teste	Valor	%
Teste 2	1490%	99,35%
Teste 28	1437%	95,80%
Teste 35	1424%	94,94%
Teste 50	1530%	102,00%
Teste 51	1503%	100,18%
Teste 52	1360%	90,65%
Teste 56	1403%	93,54%
Teste 58	1386%	92,39%
Teste 59	1433%	95,55%

Fonte: Própria autoria.

A Figura 53 mostra o layout solução do teste 28, sendo o melhor layout no que concerne ao atingimento de razão de aspecto requerida. O Quadro 14 detalha o valor de razão de aspecto alcançada pelos testes.

Figura 52 – Melhor layout solução - Teste 28 – atingimento de razão de aspecto requerida

Fonte: Própria autoria.

Quadro 14 – Percentagem de razão de aspecto atingida por cada teste

Razão de Aspecto atingida		
Teste	Valor	%
Teste 2	1368%	91,23%
Teste 28	1443%	96,20%
Teste 35	1821%	121,43%
Teste 50	1418%	94,50%
Teste 51	1312%	87,49%
Teste 52	1625%	108,36%
Teste 56	1358%	90,52%
Teste 58	1414%	94,26%
Teste 59	1602%	106,81%

Fonte: Própria autoria.

A Figura 54 exibe o layout solução do teste 51, sendo o teste que melhor alcançou os critérios de alinhamento e adjacência. O grau de atingimento dos critérios avaliados de área, razão de aspecto, e alinhamento e adjacência são apresentados no apêndice C.

Figura 53 – Melhor layout solução - Teste 51 – atingimento de alinhamento e adjacência

Fonte: Própria autoria.

Adicionalmente, realizou-se 35 testes com 500 iterações para a sequência obtida pelos autores Choi et al. (Quadro 15), obtendo-se o menor valor da função *fitness* no layout apresentado na Figura 55.

Quadro 15 – Sequência de Choi et al.

19	20	8	11	12	18	5	9	10	13	3	4	7	14	17	2	1	6	15	16
----	----	---	----	----	----	---	---	----	----	---	---	---	----	----	---	---	---	----	----

Fonte: CHOI et al. (2017).

Figura 54 – Layout obtido com 500 iterações para a sequência de Choi et al.

Fonte: Própria autoria.

4.3 RESULTADOS DA TERCEIRA ETAPA – ALCANCE OTIMIZAÇÃO GEOMÉTRICA

Nessa etapa foram executados 50 testes, sendo o melhor deles apresentado na Figura 56, correspondente a um valor da função fitness de 123,56. Esse valor é significativamente menor que os pais apresentados no apêndice D. O intervalo de confiança dos testes é apresentado no Quadro 16.

Figura 55 – Melhor filho encontrado no GA da terceira etapa

Quadro 16 – Intervalo de confiança do GA da terceira etapa

Número de testes	35		
Desvio padrão	176,03	Média	461,61
Nível de confiança	95%	Limite superior	519,92
Margem de erro	58,32	Limite inferior	403,29

Fonte: Própria autoria.

Sendo:

Área livre total: área sem uso do layout;

Área total: área total disponível do layout;

Valor alvo: definido pelo usuário, geralmente entre 0,8 e 0,9 (LIN; SHARP, 1999);

Razão de área: definida na equação 13;

Área livre parcial: cada setor da área livre do layout, se tiver mais de um setor livre;

Área retangular que cobre as áreas livres: retângulo que cobre todas as áreas livres;

Razão de forma da área livre: definida por (14), sendo a relação de aspecto.

Com base nos dados de área livre total obtido por Choi et al. (Figura 58), calculou-se o índice de utilização de área, apresentado no Quadro 17.

Figura 57 – Espaços livres do layout de Choi et al.

Quadro 17 – Índice de utilização de espaço layout Choi et al (2017)

Área livre total	291
Área total	756
Valor alvo	0,9
Razão de formato livre	880
Razão de área	0,33
Razão de formato	0,478
Índice	0,016
Índice ponderado	0,047

Fonte: Própria autoria.

O Quadro 18 apresenta o cálculo do índice para o layout solução do teste 52 desta dissertação (Figura 59).

Figura 58 – Espaços livres do layout obtido no teste 52 neste trabalho

Fonte: Própria autoria

Quadro 18 – Índice de utilização de espaço do layout teste 52 obtido neste trabalho

área livre total	266
área total	756
valor alvo	0,9
razão de formato livre	880
razão de área	0,30
razão de formato	0,478
Índice	0,0226
Índice ponderado	0,064

Fonte: Própria autoria.

Quanto mais o índice estiver perto de 1, mais qualidade em relação à utilização de espaço ele tem (LIN; SHARP, 1999). Consequentemente, pode-se inferir que o layout obtido nesta dissertação supera ao obtido pelos autores Choi et al., no que concerne a utilização de espaço.

Em relação ao índice que avalia o fluxo, Lin e Sharp (1999) propõem o cálculo do índice baseado na densidade do volume de material e a distância percorrida. O índice estima a densidade de movimento de material dos departamentos para suportar os requisitos de produção (LIN; SHARP, 1999). A equação 15 apresenta esse índice.

$$\text{Densidade de volume e distância} = \frac{\sum(\text{distância percorrida} * \text{volume de material})}{\text{volume de material}}$$
(15)

É interessante ressaltar que o índice da equação 15 proposto por Lin e Sharp (1999), é o mesmo conceito de custos de manuseio de materiais (MHC) da equação 2 da primeira etapa. O cálculo desse índice, tanto para o layout dos autores Choi et al., quanto para o layout obtido nesta dissertação, é apresentado no Quadro 19. Percebe-se que o resultado obtido nesta dissertação (1,00) é melhor do que obtido pelos autores (0,94), considerando que quanto maior é a densidade de volume de materiais, mais caro é o layout; consequentemente, nesse índice também se obteve vantagem.

Quadro 19 – Cálculo do índice de volume de material e distância

Dep. i	Dep. j	Volume de material ou Fluxo	distância	Valor
3	4	1860	1	1860
4	5	1200	1	1200
4	7	660	1	660
5	8	800	1,4142	1131,4
5	9	95	1	95
5	10	48	2	96
7	9	660	1	660
8	7	4	2	8
8	9	670	1	670
8	10	120	1,4142	169,71
9	10	1300	1	1300
10	11	700	1	700
10	13	300	1	300
11	12	600	1	600
12	14	600	2,2361	1341,6
13	11	300	1,4142	424,26
14	15	600	1	600
Valor Choi et al.				11816 → 0,94
Nesta dissertação				
3	4	1860	1	1860
4	5	1200	1	1200
4	7	660	1	660
5	8	800	1	800
5	9	95	1	95
5	10	48	1,41	67,882
7	9	660	1	660
8	7	4	2,24	8,9443

Continua Quadro 19

Continuação Quadro 19

8	9	670	1,41	947,52
8	10	120	1	120
9	10	1300	1	1300
10	11	700	1,41	989,95
10	13	300	1	300
11	12	600	1	600
12	14	600	1	600
13	11	300	1	300
14	15	600	1	600
Valor nesta dissertação				11109 → 1,00

Fonte: Própria autoria.

Os valores obtidos na aplicação do índice Lin and Sharp (1999) de aproveitamento de área em comparação com o layout dos autores foram uma surpresa, pois esse critério não foi considerado no modelamento do problema, mas poderia ser incorporado na função objetivo tal como se indica em 5.1. Já no índice de densidade de fluxo, esperava-se atingir sim um melhor performance que os autores supracitados, devido ao fato de ter melhorado na primeira o valor da função objetivo referida ao fluxo entre os departamentos e as distancias.

Uma vez atingida uma melhor solução na primeira etapa, obter melhores soluções nas etapas subsequentes era uma consequência, dado que o método depende dos resultados obtidos em cada etapa.

5 CONCLUSÃO

O método de dois estágios utilizado e complementado nesta dissertação, além de ser a mais moderna abordagem sobre o ShLP, é a única que propõe um modelo aplicável às particularidades dessa indústria. Como consequência, esse modelo tem potencial de se tornar a base teórica para a construção gradativa de uma metodologia específica para layout de estaleiros, inexistente até o momento, dado que como observado em 2.2 os autores abordam o tema de maneira isolada e independente.

Nesta dissertação, replicou-se e se adequou os dois estágios do modelo de Choi et al., (2017). Observou-se já na otimização da primeira etapa, uma melhora de 6% no valor da função objetivo (10.944) em comparação com o valor mínimo atingido pelos autores (11.816). Esse melhor mínimo foi alcançado a través da aplicação de um GA baseado na ordem para o problema. Verificou-se assim um destacável desempenho das adequações propostas e implementadas neste trabalho, que foram: GA baseado em ordem por meio do *crossover* PMX e mutação do tipo *swapping* para o primeiro estágio.

As soluções da segunda etapa aplicando a distância Chebyshev atingiram valores de 100% de área desejada (teste 51) e 96% de razão de aspecto desejada (teste 28); enquanto o teste 51 atingiu o melhor mínimo para a função objetivo (535,03) de entre todas as nove soluções.

Em referência à terceira etapa, constatou-se ainda o atingimento de melhores valores para a função objetivo que os alcançados no segundo estágio, sendo o menor valor atingido de 123,36. Também, para efeitos comparativos, utilizou-se um índice de qualidade de layout de fábrica, a fim de validar as soluções obtidas para o layout do estaleiro, constatando-se melhor performance do teste 52 em comparação com o layout solução dos autores.

Dessa forma, cumpre-se com o objetivo ter encontrado uma melhor solução (melhor mínimo e um melhor layout) para o layout estudado pelos autores Choi et al (2017) aplicando melhorias destacadas na literatura, como é o GA baseado na ordem com *crossover* PMX e mutação *swapping*.

5.1 SUGESTÕES DE TRABALHOS FUTUROS

O estudo do ShLP é um assunto com altíssimo potencial de desenvolvimento devido aos escassos trabalhos dedicados ao tema. Por isso, são propostas as seguintes complementações para o modelo estudado:

Primeiro Estágio:

- Definição e agregação de caminhos ou rotas de transporte no layout do estaleiro, acrescentando esses termos no modelo matemático, sendo que nos estaleiros os caminhos ou rotas apresenta limitações físicas de largura comprimento, além de permitir projetar as rotas que minimizem o fluxo.
- Definição de pontos *pick-up* e *drop-off* para cada departamento, possibilitando o cálculo da distância exata de movimentação, dado que todos os departamentos do estaleiro possuem portas de ingresso/saída de matérias;
- Explorar geometricamente as outras sequências obtidas.

Segundo Estágio:

- Sendo importante o máximo aproveitamento do terreno, seria oportuno acrescentar um sexto termo na equação multiobjetivo do segundo estágio, representando uma penalidade por área não utilizada, calculada como a diferença entre a área total da grade e a área atingida pelos departamentos em cada iteração, como apresentado a seguir:

$$MinF^* = \omega_1 \cdot \sum Fluxo + \omega_2 \cdot \sum Área\ atingida + \omega_3 \cdot \sum RA\ atingida + \omega_4 \cdot \sum AD + \omega_5 \cdot \sum AL + \omega_6 \cdot \sum (Área_{grade} - Área\ atingida)$$

- Utilizar a análise AHP nos pesos ω 's para definir uma melhor ponderação deles, incluindo talvez o critério dos funcionários do próprio estaleiro para definir o grau de importância de cada ω , segundo o ponto de vista prático.

5.2 LIMITAÇÕES DA PESQUISA

Como limitação, destaca-se que, dos resultados obtidos no primeiro estágio, utilizou-se somente uma solução de onze disponíveis para ser explorada no segundo estágio, limitando, portanto, a possibilidade de examinar mais opções para a solução geométrica. Além disso, destaca-se como limitante a falta de disponibilidade do arquivo executável do algoritmo dos autores para efetuar uma comparação válida entre os algoritmos implementados

e acrescentar o tempo computacional como um outro parâmetro de qualidade do código na comparação.

Também uma importante limitação deste trabalho é a omissão do cálculo do valor matemático exato para a função objetivo da primeira etapa, sendo que apenas 15 departamentos apresentam movimentação e os outros 5 são fixos, razão pela qual existem, para esse número de departamento, vários algoritmos e métodos exatos (*Branch and Bound*, software CEPLEX) para determinar o ótimo global e, dessa maneira, precisar o *delta* entre a solução exata e a melhor solução sub-ótima obtida pelo GA.

REFERÊNCIAS

- AIELLO, G.; ENEA, M.; GALANTE, G. A multi-objective approach to facility layout problem by genetic search algorithm and Electre method. **Robotics and Computer-Integrated Manufacturing**, 2006.
- ANDREWS, B. V; HANEY, D. G. **Conceptual design of a Mechanized Shipyard for fast Deployment Logistics Production**. [s.l: s.n.].
- ARGOUD, A. R. T. . **Procedimento para projeto de arranjo físico modular em manufatura através de algoritmo genético de agrupamento**. [s.l.] USP, 2007.
- AZADIVAR, F.; WANG, J. Facility layout optimization using simulation and genetic algorithms. **International Journal of Production Research**, v. 38, n. 17, p. 4369–4383, 2000.
- BERTRAND, J. W. M.; FRANSOO, J. C. **Researching Operations Management**. 1.ed. New York e Londres: Taylor & Francis, 2009.
- CHABANE, H. **Design of a Small Shipyard Facility Layout Optimised for Production and Repair**. Symposium International : Qualité et Maintenance au Service de l'Entreprise. **Anais...**2004
- CHANG, S. Fundamentals of Shipyard Layout. **BULLETIN OF THE SOCIETY OF NAVAL ARCHITECTS OF KOREA**, v. 13, n. 13, p. 25–43, 1976.
- CHOI, M.; KIM, S. H.; CHUNG, H. Optimal shipyard facility layout planning based on a genetic algorithm and stochastic growth algorithm. **Ships and Offshore Structures**, 2017.
- DAMSKI, J. C.; GERO, J. S. A logic-based framework for shape representation. **CAD Computer Aided Design**, 1996.
- DE LACERDA ESTÉFANE GM; DE CARVALHO, A. **Introdução aos algoritmos genéticos, Sistemas inteligentes: aplicações a recursos hídricos e ciências ambientais**. 1.ed. [s.l.] Coleção Associação Brasileira de Recursos Hídricos, 1999.
- DE SOUZA, D. **Algoritmos Genéticos Aplicados ao Planejamento do Transporte Principal de Madeira**. [s.l.] Universidade Federal de Paraná, 2004.
- DRIRA, A.; PIERREVAL, H.; HAJRI-GABOUJ, S. Facility layout problems: A survey. **Annual Reviews in Control**, 2007.
- DRIRA, A.; PIERREVAL, H.; HAJRI-GABOUJ, S. Design of a robust layout with information uncertainty increasing over time: A fuzzy evolutionary approach. **Engineering Applications of Artificial Intelligence**, 2013.
- EIBEN, A. E.; SMITH, J. E. **Introduction to Evolutionary Computing Genetic Algorithms**. [s.l.] Springer-Verlag Berlin Heidelberg, 2012.
- GEN, M.; CHENG, R. **Genetic Algorithms & Engineering Design**. [s.l.] Jhon Wiley

and Sons, 1997.

GOLDBARG, M.; GOLDBARG, E.; PACCA L. LUNA, H. **Otimização Combinatória E Meta-heurísticas**. 1.ed. Rio de Janeiro: [s.n.].

HOSSEINI-NASAB, H. et al. Classification of facility layout problems: a review study. **International Journal of Advanced Manufacturing Technology**, 2018.

JUNQUEIRA MARTARELLI, N. **Seleção de atributos em agrupamento de dados utilizando algoritmos evolutivos**. [s.l.] USP, 2016.

KOOPMANS, T. C.; BECKMANN, M. Assignment Problems and the Location of Economic Activities. **Econometrica**, 1957.

KUSIAK, A.; HERAGU, S. S. The facility layout problem. **European Journal of Operational Research**, v. 29, n. 3, p. 229–251, jun. 1987.

KWON, B.; LEE, G. M. Spatial scheduling for large assembly blocks in shipbuilding. **Computers and Industrial Engineering**, 2015.

LEE, D.-K.; JEONG, Y.-K.; SHIN, J.-G. Study on a Layout Design Method for Leisure Ship Production Factories. **Journal of the Korean Society of Marine Environment & Safety**, v. 19, n. 3, p. 277–284, 2013.

LEE, D. K. et al. 26 Simulation-Based Work Plan Verification in Shipyards. **Journal of Ship Production and Design**, v. 30, n. 2, p. 49–57, 2014.

LIGGETT, R. S. The quadratic assignment problem: an analysis of applications and solution strategies. **Environment and Planning B**, v. 7, n. 2, p. 141–162, 1980.

LIMA, M. R. C. et al. Aprendizagem cooperativa e o problema de formação de grupos. **Novas Tecnologias na Educação**, v. 3, n. 1, p. 1–10, 2005.

LIN, L. C.; SHARP, G. P. Quantitative and Qualitative indices for the plant layout evaluation problem. **European Journal of Operation Research**, v. 116, p. 100–117, 1999.

LINDEN, R. **Algoritmos Genéticos - Uma Importante Ferramenta da Inteligência Computacional**. 3. ed. Rio de Janeiro: Brasport, 2012.

LIU, Y.; MENG, M.; LIU, S. Layout design-based research on optimization and assessment method for shipbuilding workshop. **Journal of Marine Science and Application**, 2013.

MATSON, J. O.; MELLICHAMPH, J. M.; SWAMINATHAN, S. R. EXCITE: Expert consultant for in-plant transportation equipment. **International Journal of Production Research**, 1992.

MATULJA, T.; FAFANDJEL, N.; ZAMARIN, A. Methodology for shipyard production areas optimal layout design. **Brodogradnja**, 2009.

MAVRIDOU, T. D.; PARDALOS, P. M. Simulated Annealing and Genetic Algorithms for

the Facility Layout Problem : A Survey. **Computational Optimization and Applications**, v. 7, p. 111–126, 1997.

MEDJDOUB, B.; YANNOU, B. Separating topology and geometry in space planning. **CAD Computer Aided Design**, 2000.

MICHALEK, J. J.; CHOUDHARY, R.; PAPALAMBROS, P. Y. Architectural layout design optimization. **Engineering Optimization**, v. 34, n. 5, p. 461–484, 2002.

MITCHELL, M. **An Introduction to Genetic Algorithms**. 1. ed. Cambridge, Massachusetts: Cambridge, Mass MIT Press , 1988.

MOTTA TOLEDO, C. F. et al. A genetic algorithm/mathematical programming approach to solve a two-level soft drink production problem. **Computers and Operations Research**, 2014.

NICK, E. K. Fuzzy Optimal Allocation and Arrangement of Spaces in Naval Surface Ship Design. 2008.

OBADAÇI, Y. et al. Reengineering of Small and Medium-Sized Ship Production Facilities: An Example for Turkish Shipbuilding Industry. **Journal of Ship Production**, v. 13, n. 1, p. 8–15, 1997.

OMER SARACOGLU, B.; GOZLU, S. **Identification of Production Performance Criteria in Shipbuilding Industry**. The Seventeenth International Offshore and Polar Engineering. **Anais**. Lisboa, Portugal: The International Society of Offshore and Polar Engineers (ISOPE), 2007

RODRIGUES VIANA, G. V. **Meta-heurísticas e programação paralela em otimização combinatória**. 1. ed. Fortaleza: UFC, 1998.

SAHA, A. **Artificial bee colony (abc) algorithm applied to slope- stability in searching the critical surface**. Kolkata: [s.n.]. Disponível em: <<https://www.researchgate.net/publication/294087456>>. Acesso em: 15 abr. 2018.

SAHNI, S.; GONZALEZ, T. P-Complete Approximation Problems. **Journal of the ACM**, 1976.

SHIN, J. G. et al. A Concept and Framework for a Shipyard Layout Design Based on Simulation. **Journal of Ship Production**, v. 25, n. 3, p. 126–135, 2009.

SONG, Y. J. et al. System Development and Applications of a Shipyard Layout Design Framework. **Journal of Ship Production and Design**, v. 26, n. 2, p. 144–154, 2010.

SONG, Y. J.; WOO, J. H. New shipyard layout design for the preliminary phase & case study for the green field project. **International Journal of Naval Architecture and Ocean Engineering**, 2013.

SYRKOV, A. K. **Handbook on the engineering design of shipyards shipbuilding shops and sheds**. [s.l.] FBIS, 1981.

TOMPKINS, J. A. et al. **Planejamento de Instalações** - 4.ed. Rio de Janeiro: LTC, 2013.

TUSON, A. L. **Adapting Operator Probabilities In Genetic Algorithms**. [s.l.] University of Edinburgh, 1995.

YAN TAM, K. **Genetic algorithms, function optimization, and facility layout design**European Journal of Operational Research. [s.l: s.n.].

YI, H.; YI, Y. K. **Performance based architectural design optimization: automated 3d space layout using simulated annealing**. (ASHRAE/IBPSA, Ed.)Building Simulation Conference . **Anais...**Atlanta: ASHRAE/IBPSA, 2014. Disponível em: <www.ashrae.org>. Acesso em 15 abr. 2018.

YOON, K. B.; COYNE, R. D. Reasoning about spatial constraints. **Environment and Planning B: Planning and Design**, v. 19, p. 243–266, 1992.

APÊNDICE A – CÓDIGO DO PRIMERIO ESTÁGIO

```

import numpy as np
import pandas as pd

class Genetico:
    listAd = [[4,5],[7,9]]
    listAli = [[3,4],[4,7],[5,9],[9,10],[11,12]]

    def __init__(self,arquivo='Fluxo_custo.xlsx'):
        dados = pd.ExcelFile(arquivo)
        self.fluxo = pd.read_excel(dados,'Fluxo')
        self.custo = pd.read_excel(dados,'Custo')
        self.fluxo = self.fluxo.set_index(np.arange(1,21))
        self.custo = self.custo.set_index(np.arange(1,21))

    def getPos(self,i):
        coluna = i%5
        if coluna == 0:
            coluna=5
        linha = int(i/5)+1
        return coluna,linha

    def calcDistancia(self,item,item2):
        x,y = self.getPos(item)
        x1,y1 = self.getPos(item2)
        return ((x-x1)**2 + (y-y1)**2)**0.5

    def calcValor(self,lista):
        valor = 0
        posItem1 = 1
        for j in lista:
            posItem2 = 1
            for i in lista:
                valorAtual = self.fluxo[j][i] * self.custo[j][i] * self.calcDistancia(posItem2,posItem1)
                valor += valorAtual
                posItem2 += 1
            posItem1 += 1
        return valor

    def inserirDadosFixos(self,lista):
        listalns = np.insert(lista, 0, 20)
        listalns = np.insert(listalns , 0, 19)
        listalns = np.insert(listalns , 5, 18)
        listalns = np.insert(listalns , len(listalns) , 15)
        listalns = np.insert(listalns , len(listalns) , 16)
        listalns = np.insert(listalns , len(listalns) - 5, 17)
        return listalns

    def gerarListaAleatoria(self):
        aleatorio = np.arange(1, 15)

```

```

np.random.shuffle(aleatorio)
# aleatorio = inserirDadosFixos(aleatorio)
return aleatorio

def crossover(self, lista1, lista2):
    metade = int(len(lista1)/2)
    pontCruz = np.random.randint(metade)
    filho = np.zeros(len(lista1))
    filho[pontCruz:pontCruz+metade] = lista1[pontCruz:pontCruz+metade]
    for i in range(len(lista2)):
        if filho[i] == 0:
            novoltem = lista2[i]
            while novoltem in filho:
                #print("Entrou", novoltem, "Outro", lista2[i], "i:", i)
                pos = np.where(filho==novoltem)[0][0]
                novoltem = lista2[pos]
            filho[i] = novoltem
    return filho

def checkItem(self, df, x):
    for i in range(len(df)):
        if x == list(df.iloc[i]):
            return True
    return False

def mutação(self, percMutacao, filho):
    numero = np.random.randint(0,100)
    if numero < 10:
        # print("mut:", numero)
        numero1 = np.random.randint(0,14)
        numero2 = np.random.randint(0,14)
        aux = filho[numero1]
        filho[numero1] = filho[numero2]
        filho[numero2] = aux
    return filho

def TestAdj(self, listAd, lista):
    for item in listAd:
        pos1 = np.where(np.array(lista) == item[0])[0]
        pos2 = np.where(np.array(lista) == item[1])[0]
        if not (self.calcDistancia(pos1, pos2) == 2 ** 0.5):
            return False
    return True

def TestAli(self, listAli, lista):
    for item in listAli:
        pos1 = np.where(np.array(lista) == item[0])[0]
        pos2 = np.where(np.array(lista) == item[1])[0]
        if not (self.calcDistancia(item[0], item[1]) == 1):
            return False
    return True

def cruzamento(self, dfPop, popMax, porMuta):
    valores = []

```

```

dfPop = dfPop.sort_values(by=1)
# pegando os pais
listPai1 = np.unique(abs(np.random.normal(0, 1, int(len(dfPop)*0.5)).astype(int)))
listPai2 = np.unique(abs(np.random.normal(0, 1, int(len(dfPop)*0.5)).astype(int)))
for i in listPai1:
    for j in listPai2:
        if j != i :
            j += 1
            pai1 = dfPop.iloc[j][0]
            pai2 = dfPop.iloc[i][0]
            # Fazendo cruzamento
            filho = self.crossover(pai1, pai2)

            #Fazendo teste de retrições
            # if not (self.TestAdj(self.listAd, self.inserirDadosFixos(filho)) or self.TestAli(self.listAli,
self.inserirDadosFixos(filho))):
                # return dfPop
                filho = self.mutação(porMuta, filho)
                valor = self.calcValor(self.inserirDadosFixos(filho))
                if not self.checkItem(dfPop[0], list(filho)):
                    # Adicionando filho a população
                    dfPop = dfPop.append([[filho, valor]], ignore_index=True)
                    dfPop = dfPop.sort_values(by=1)
                if len(dfPop) > popMax:
                    ultima = dfPop.index[len(dfPop.index) - 1]
                    dfPop = dfPop.drop(ultima)
            return dfPop
def algoritmoGenet(self, popInicial=50, popMax=1000, maxEras=1000, porMuta=0.1):
    população = []
    dfPop = pd.DataFrame()
    # Gerando população inicial
    for i in range(popInicial):
        novo = self.gerarListaAleatoria()
        while self.TestAdj(self.listAd, self.inserirDadosFixos(novo)) and self.TestAli(self.listAli,
self.inserirDadosFixos(novo)):
            novo = self.gerarListaAleatoria()
            valor = self.calcValor(self.inserirDadosFixos(novo))
            dfPop = dfPop.append([[novo, valor]], ignore_index=True)
    # A cada criando 1000 filhos
    for i in range(maxEras):
        dfPop = self.cruzamento(dfPop, popMax, porMuta)
    return dfPop

# dfPop = algoritmoGenet(50, 100, 1000)
# saida = pd.ExcelWriter('saida.xlsx')
# dfPop.to_excel(saida)
# saida.save()
# print(dfPop)

```

APÊNDICE B - TESTES COMPUTACIONAIS DA PRIMEIRA ETAPA

teste	cromossomo																				Fitness
1	19	20	1	13	6	18	9	10	12	11	7	5	8	14	17	4	3	2	15	16	12.537
2	19	20	7	4	3	18	10	9	5	8	11	12	1	2	17	13	6	14	15	16	12.389
3	19	20	8	6	2	18	9	5	3	1	13	10	7	4	17	11	12	14	15	16	12.027
4	19	20	8	7	4	18	10	9	5	3	1	13	11	12	17	6	2	14	15	16	12.255
5	19	20	7	4	6	18	9	5	3	2	1	10	8	14	17	13	11	12	15	16	11.962
6	19	20	11	12	6	18	13	10	14	1	3	5	9	2	17	4	7	8	15	16	12.483
7	19	20	7	4	1	18	9	5	3	6	13	10	8	2	17	12	11	14	15	16	12.313
8	19	20	7	4	3	18	10	9	5	1	13	12	11	8	17	2	6	14	15	16	11.959
9	19	20	7	4	3	18	2	9	5	1	12	11	10	8	17	6	13	14	15	16	11.685
10	19	20	8	7	4	18	10	9	5	3	13	11	12	1	17	2	6	14	15	16	11.841
11	19	20	7	4	3	18	13	9	5	1	2	14	10	8	17	6	12	11	15	16	12.056
12	19	20	1	13	2	18	9	10	12	11	7	5	8	14	17	4	3	6	15	16	12.537
13	19	20	8	13	1	18	5	9	10	6	4	7	12	11	17	3	2	14	15	16	12.021
14	19	20	1	11	12	18	9	10	13	6	7	5	8	14	17	4	3	2	15	16	12.620
15	19	20	7	4	3	18	13	9	5	2	1	14	10	8	17	6	12	11	15	16	12.056
16	19	20	6	3	4	18	13	8	5	7	1	11	10	9	17	2	12	14	15	16	11.713
17	19	20	7	4	3	18	2	9	5	8	6	13	10	14	17	1	12	11	15	16	12.222
18	19	20	8	6	1	18	5	9	12	14	4	7	10	11	17	3	2	13	15	16	12.630
19	19	20	6	7	4	18	2	9	5	3	13	11	10	8	17	1	12	14	15	16	11.889
20	19	20	7	4	3	18	9	5	1	2	13	10	8	14	17	6	11	12	15	16	11.962
21	19	20	7	4	3	18	10	9	5	8	12	11	13	2	17	1	6	14	15	16	12.346
22	19	20	1	4	3	18	8	5	7	2	13	10	9	14	17	6	11	12	15	16	11.962
23	19	20	13	11	6	18	8	10	12	1	2	5	9	14	17	3	4	7	15	16	11.109
24	19	20	10	13	6	18	8	9	11	12	2	7	5	14	17	1	4	3	15	16	12.255
25	19	20	7	4	3	18	8	9	5	6	1	13	10	14	17	2	12	11	15	16	12.254
26	19	20	7	4	3	18	9	5	1	6	13	10	8	2	17	12	11	14	15	16	12.313
27	19	20	8	4	3	18	1	5	7	6	13	10	9	2	17	11	12	14	15	16	12.417
28	19	20	7	4	3	18	1	9	5	8	2	6	10	14	17	13	11	12	15	16	12.269
29	19	20	8	7	4	18	10	9	5	3	2	11	13	6	17	1	12	14	15	16	11.841
30	19	20	11	12	6	18	2	10	13	14	3	5	9	1	17	4	7	8	15	16	12.873
31	19	20	8	7	4	18	1	9	5	3	2	13	10	14	17	6	11	12	15	16	12.297
32	19	20	12	6	2	18	8	11	13	1	5	9	10	14	17	7	4	3	15	16	12.587
33	19	20	7	4	3	18	11	9	5	8	6	12	10	14	17	2	13	1	15	16	12.674
34	19	20	7	4	3	18	2	9	5	1	6	12	10	8	17	14	11	13	15	16	12.558
35	19	20	13	11	2	18	9	10	12	14	7	5	8	1	17	4	3	6	15	16	12.620
36	19	20	6	3	2	18	13	7	4	1	11	10	9	5	17	12	14	8	15	16	11.674
37	19	20	7	4	3	18	2	9	5	1	6	14	10	8	17	13	12	11	15	16	12.232
38	19	20	8	7	4	18	10	9	5	3	1	11	12	14	17	6	2	13	15	16	12.212
39	19	20	13	6	1	18	9	10	11	12	8	5	7	14	17	2	3	4	15	16	12.110
40	19	20	7	4	6	18	9	5	3	2	13	10	8	14	17	1	11	12	15	16	11.962
41	19	20	7	4	3	18	13	9	5	8	6	2	10	14	17	1	12	11	15	16	12.597
42	19	20	2	6	13	18	9	10	12	11	7	5	8	14	17	4	3	1	15	16	12.660
43	19	20	1	4	3	18	13	5	7	6	2	10	9	8	17	11	12	14	15	16	12.419

44	19	20	7	4	3	18	6	9	5	1	2	14	10	8	17	13	12	11	15	16	12.232
45	19	20	1	7	4	18	10	9	5	3	13	11	2	8	17	6	12	14	15	16	11.890
46	19	20	13	6	1	18	10	11	12	2	8	9	5	14	17	7	4	3	15	16	12.281
47	19	20	5	9	8	18	4	7	10	13	6	3	12	11	17	1	2	14	15	16	12.286
48	19	20	2	7	4	18	13	9	5	3	12	11	10	8	17	6	14	1	15	16	12.562
49	19	20	9	10	13	18	7	5	11	12	3	4	8	14	17	6	2	1	15	16	12.500
50	19	20	7	4	2	18	9	5	3	6	13	10	8	14	17	1	11	12	15	16	11.962
51	19	20	6	11	2	18	9	10	12	13	7	5	8	14	17	4	3	1	15	16	12.303
52	19	20	6	1	3	18	13	8	7	4	2	10	9	5	17	11	12	14	15	16	12.377
53	19	20	10	11	13	18	5	9	12	1	4	7	8	14	17	3	6	2	15	16	12.059
54	19	20	7	4	3	18	10	9	5	8	1	11	12	6	17	2	13	14	15	16	11.780
55	19	20	4	3	13	18	5	7	11	6	8	9	10	12	17	1	2	14	15	16	12.975
56	19	20	2	7	4	18	10	9	5	3	13	11	8	1	17	6	12	14	15	16	11.846
57	19	20	11	12	1	18	8	10	14	2	3	5	9	13	17	4	7	6	15	16	12.684
58	19	20	1	7	4	18	11	9	5	3	13	12	10	8	17	2	6	14	15	16	12.552
59	19	20	8	7	4	18	10	9	5	3	13	11	12	14	17	2	6	1	15	16	11.841
60	19	20	8	7	4	18	10	9	5	3	6	11	13	14	17	1	12	2	15	16	12.582
61	19	20	6	1	3	18	2	8	7	4	11	10	9	5	17	12	13	14	15	16	12.441
62	19	20	6	4	3	18	13	5	7	1	2	10	9	8	17	11	12	14	15	16	12.419
63	19	20	5	6	1	18	8	9	4	3	13	11	10	7	17	2	12	14	15	16	12.021
64	19	20	7	4	3	18	10	9	5	1	11	12	6	8	17	13	2	14	15	16	11.905
65	19	20	1	13	2	18	8	10	11	6	3	5	9	12	17	4	7	14	15	16	11.962
66	19	20	7	4	3	18	10	9	5	8	1	11	14	6	17	13	12	2	15	16	12.472
67	19	20	7	4	1	18	9	5	3	6	13	10	8	14	17	11	12	2	15	16	12.620
68	19	20	7	4	3	18	13	9	5	8	6	11	10	2	17	1	12	14	15	16	11.484
69	19	20	13	2	1	18	9	10	11	6	7	5	8	12	17	4	3	14	15	16	11.962
70	19	20	5	8	2	18	10	9	4	3	6	11	13	7	17	1	12	14	15	16	11.861
71	19	20	5	7	2	18	8	9	4	3	1	11	10	13	17	6	12	14	15	16	12.021
72	19	20	6	7	4	18	10	9	5	3	11	13	1	8	17	12	2	14	15	16	12.655
73	19	20	10	13	1	18	5	9	11	6	4	7	8	12	17	3	2	14	15	16	12.297
74	19	20	7	4	3	18	11	9	5	6	1	12	10	8	17	2	13	14	15	16	11.782
75	19	20	7	4	3	18	2	9	5	1	11	13	10	8	17	12	14	6	15	16	12.119
76	19	20	8	7	4	18	13	9	5	3	2	11	10	6	17	1	12	14	15	16	11.883
77	19	20	7	4	3	18	10	9	5	1	11	13	2	8	17	12	14	6	15	16	11.886
78	19	20	7	4	3	18	10	9	5	8	2	11	12	14	17	6	1	13	15	16	11.975
79	19	20	1	7	4	18	10	9	5	3	6	11	13	8	17	2	12	14	15	16	11.889
80	19	20	7	4	3	18	10	9	5	1	13	11	8	6	17	2	12	14	15	16	11.074
81	19	20	7	4	3	18	13	9	5	1	6	11	10	8	17	14	12	2	15	16	12.144
82	19	20	7	4	3	18	11	9	5	2	13	12	10	8	17	6	14	1	15	16	12.134
83	19	20	6	4	3	18	13	5	7	2	11	10	9	8	17	12	14	1	15	16	12.483
84	19	20	7	4	3	18	8	9	5	1	2	11	10	6	17	13	12	14	15	16	11.883
85	19	20	7	4	3	18	11	9	5	6	13	12	10	8	17	2	14	1	15	16	12.134
86	19	20	7	4	3	18	10	9	5	6	11	12	8	14	17	1	13	2	15	16	12.264
87	19	20	7	4	3	18	13	9	5	2	1	11	10	8	17	6	12	14	15	16	10.944
88	19	20	13	2	1	18	8	10	11	6	3	5	9	12	17	4	7	14	15	16	11.962
89	19	20	7	4	3	18	10	9	5	1	6	11	8	2	17	13	12	14	15	16	11.445

90	19	20	7	4	3	18	10	9	5	2	11	12	13	8	17	1	6	14	15	16	11.959
91	19	20	7	4	6	18	9	5	3	2	13	10	8	1	17	11	12	14	15	16	11.879
92	19	20	2	3	4	18	13	8	5	7	6	11	10	9	17	1	12	14	15	16	11.713
93	19	20	7	4	3	18	1	9	5	8	2	11	10	14	17	6	12	13	15	16	12.226
94	19	20	2	6	1	18	8	10	11	12	3	5	9	14	17	4	7	13	15	16	12.508
95	19	20	7	4	3	18	1	9	5	2	6	13	10	8	17	12	11	14	15	16	11.709
96	19	20	8	7	4	18	2	9	5	3	1	13	10	14	17	6	11	12	15	16	12.297
97	19	20	13	12	2	18	9	10	11	1	7	5	8	14	17	4	3	6	15	16	12.313
98	19	20	7	4	3	18	11	9	5	6	2	12	10	8	17	1	13	14	15	16	11.782
99	19	20	7	4	1	18	9	5	3	6	13	10	8	14	17	2	11	12	15	16	11.962
100	19	20	5	8	2	18	1	9	4	3	13	11	10	7	17	6	12	14	15	16	12.065

APÊNDICE C – ATINGIMENTO DOS CRITÉRIOS AVALIADOS – 2da ETAPA

Área Atingida																			
Dpto.	Área Req.	Teste 2		Teste 28		Teste 35		Teste 50		Teste 51		Teste 52		Teste 56		Teste 58		Teste 59	
		Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%	Área. Ating.	%
1	9	9	100%	8	89%	8	89%	8	89%	9	100%	9	100%	9	100%	8	89%	9	100%
2	28	28	100%	27	96%	20	71%	25	89%	30	107%	20	71%	30	107%	24	86%	20	71%
3	15	15	100%	15	100%	12	80%	20	133%	16	107%	10	67%	15	100%	15	100%	15	100%
4	14	15	107%	15	107%	12	86%	20	143%	12	86%	10	71%	12	86%	15	107%	15	107%
5	50	50	100%	52	104%	50	100%	50	100%	42	84%	45	90%	45	90%	50	100%	48	96%
6	12	12	100%	8	67%	12	100%	12	100%	12	100%	12	100%	8	67%	9	75%	12	100%
7	22	20	91%	21	95%	24	109%	21	95%	20	91%	20	91%	21	95%	16	73%	15	68%
8	25	24	96%	24	96%	16	64%	16	64%	28	112%	18	72%	16	64%	15	60%	24	96%
9	50	50	100%	50	100%	50	100%	50	100%	50	100%	50	100%	48	96%	50	100%	50	100%
10	35	30	86%	30	86%	40	114%	40	114%	40	114%	32	91%	32	91%	32	91%	30	86%
11	30	32	107%	32	107%	30	100%	32	107%	32	107%	33	110%	32	107%	30	100%	32	107%
12	40	40	100%	40	100%	40	100%	40	100%	40	100%	44	110%	40	100%	40	100%	40	100%
13	8	8	100%	8	100%	8	100%	8	100%	8	100%	8	100%	8	100%	8	100%	8	100%
14	20	22	110%	18	90%	22	110%	21	105%	21	105%	18	90%	20	100%	21	105%	20	100%
15	144	135	94%	144	100%	145	101%	130	90%	130	90%	138	96%	144	100%	144	100%	147	102%
	Total	1490%		1437%	Total	1424%	Total	1530%	Total	1503%	Total	1360%	Total	1403%	Total	1386%	Total	1433%	

Nota: O teste 5 é o teste com melhor percentagem de atingimento de área requerida, com 1503% de 1500% possíveis.

Dpto.	RA Req.	Razão de Aspecto Atingido																	
		Teste 2		Teste 28		Teste 35		Teste 50		Teste 51		Teste 52		Teste 56		Teste 58		Teste 59	
		RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%	RA Ating.	%
1	2	2,0	50%	2,0	100%	2,0	100%	2,0	100%	1,0	50%	1,0	50%	1,0	50%	2,0	100%	1,0	50%
2	2	1,0	50%	3,0	150%	5,0	250%	1,0	50%	1,2	60%	5,0	250%	1,2	60%	1,5	75%	5,0	250%
3	2	1,8	88%	3,0	150%	5,0	250%	1,0	50%	1,2	60%	5,0	250%	1,2	60%	1,5	75%	5,0	250%
4	3	1,7	83%	1,7	83%	1,7	83%	1,3	63%	1,3	63%	2,5	125%	1,7	83%	1,7	83%	1,7	83%
5	2	1,7	56%	1,7	56%	1,7	56%	1,3	42%	1,3	44%	2,5	83%	1,3	44%	1,7	56%	1,7	56%
6	2	2,0	100%	3,3	163%	2,0	100%	2,0	100%	4,7	233%	1,8	90%	1,8	90%	2,0	100%	3,0	150%
7	4	1,3	67%	2,0	100%	1,3	67%	1,3	67%	1,3	67%	1,3	67%	2,0	100%	1,3	67%	3,0	150%
8	2	1,3	31%	2,3	58%	2,7	67%	2,3	58%	1,3	31%	1,3	31%	2,3	58%	4,0	100%	1,7	42%
9	2	1,5	75%	2,7	133%	4,0	200%	4,0	200%	1,8	88%	2,0	100%	4,0	200%	1,7	83%	1,5	75%
10	2	2,0	100%	1,8	90%	2,0	100%	2,5	125%	2,0	100%	2,0	100%	1,3	67%	2,0	100%	2,0	100%
11	2	3,3	167%	1,6	80%	2,5	125%	2,5	125%	2,5	125%	2,0	100%	2,0	100%	2,0	100%	3,3	167%
12	2	2,0	100%	2,0	100%	3,3	167%	2,0	100%	1,2	60%	3,7	183%	2,0	100%	3,3	167%	2,0	100%
13	2	1,6	80%	1,6	80%	2,5	125%	2,0	100%	2,0	100%	2,8	138%	1,6	80%	2,5	125%	1,6	80%
14	4	2,0	100%	2,0	100%	2,0	100%	2,0	100%	2,0	100%	2,0	100%	2,0	100%	2,0	100%	2,0	100%
15	4	5,5	138%	2,0	50%	5,5	138%	2,3	58%	4,0	100%	4,5	113%	5,0	125%	2,3	58%	5,0	125%
		5,4	135%	4,0	100%	5,8	145%	5,2	130%	3,7	92%	3,8	96%	4,0	100%	4,0	100%	3,0	75%
		Total	1368%	Total	1443%	Total	1821%	Total	1418%	Total	1312%	Total	1625%	Total	1358%	Total	1414%	Total	1602%

Nota: O teste 28 é o teste com melhor percentagem de atingimento de razão de aspecto requerida, com 1443% de 1500% possíveis.

		Penalização por Alinhamento e Adjacência								
Dpto.	Dpto.	Teste 2	Teste 28	Teste 35	Teste 50	Teste 51	Teste 52	Teste 56	Teste 58	Teste 59
3	4	0	0	0	0	0	0	0	0	0
4	5	1	0	0	0	0	0	0	0	0
4	7	0	0	1	1	0	0	0	1	0
5	9	1	1	1	1	0	1	1	1	1
7	9	0	1	0	0	0	1	1	0	0
9	10	0	0	0	0	1	0	0	1	0
11	12	0	0	0	0	0	0	0	0	1
	Total	2	2	2	2	1	2	2	3	2

Nota: O teste 51 é o teste com menor numero de penalização (uma) por incumprimento dos requisitos de alinhamento e adjacência.

Teste 50

Valor: 934,4246

[illegible]

Teste 51

Valor: 535,0315

[illegible]Teste 52

Valor: 732,6149

[illegible]

[illegible][illegible]