



Universidade Federal da Paraíba
Centro de Tecnologia
Programa de Pós-Graduação em Engenharia Mecânica
Mestrado – Doutorado

**UMA METAHEURÍSTICA ILS APLICADA AO PROBLEMA DE
FORMAÇÃO DE CÉLULAS DE MANUFATURA**

por

Allisson Magno Rodrigues e Silva

*Dissertação de Mestrado apresentada à Universidade federal da Paraíba para
obtenção do grau de Mestre.*

João Pessoa – Paraíba

Julho, 2024

ALLISSON MAGNO RODRIGUES E SILVA

**UMA METAHEURÍSTICA ILS APLICADA AO PROBLEMA DE
FORMAÇÃO DE CÉLULAS DE MANUFATURA**

Dissertação de mestrado apresentada ao curso de
Pós-Graduação em Engenharia Mecânica da
Universidade Federal da Paraíba, em cumprimento
às exigências para obtenção do Grau de Mestre

Orientador: Prof. Dr. Lucidio dos Anjos Formiga Cabral

Coorientador: Prof. Dr. Sandro Marden Torres

Catálogo na publicação
Seção de Catalogação e Classificação

S586m Silva, Allisson Magno Rodrigues e.

Uma metaheurística ILS aplicada ao problema de formação de células de manufatura / Allisson Magno Rodrigues e Silva. - João Pessoa, 2024.

49 f. : il.

Orientação: Lucidio dos Anjos Formiga Cabral.

Coorientação: Sandro Marden Torres.

Dissertação (Mestrado) - UFPB/CT.

1. Células de manufatura. 2. Metaheurística. 3. ILS (Interated Local Search). 4. VND (Variable Neighborhood Descent). I. Cabral, Lucidio dos Anjos Formiga. II. Torres, Sandro Marden. III. Título.

UFPB/BC

CDU 658.51(043)

UMA METAHEURÍSTICA ILS APLICADA AO PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MANUFATURA

por

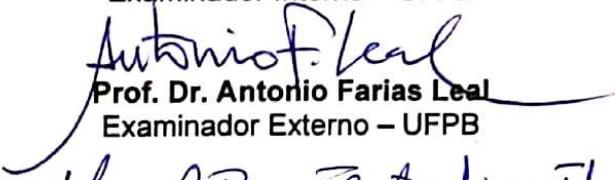
ALLISSON MAGNO RODRIGUES E SILVA

Dissertação aprovada em 30 de julho de 2024


Prof. Dr. Lucídio dos Anjos Formiga Cabral
Orientador – UFPB


Prof. Dra. Kelly Cristiane Gomes da Silva
Examinador Interno – UFPB


Prof. Dr. Sandro Marden Torres
Examinador Interno – UFPB


Prof. Dr. Antonio Farias Leal
Examinador Externo – UFPB


Prof. Dr. Marçal Rosas F Lima Filho
Examinador Externo – UFPB

AGRADECIMENTOS

Agradeço primeiramente a Deus, que me guiou e sustentou em cada momento dessa jornada, dando-me forças e serenidade para superar os desafios. À minha mãe, Regina Celi, dedico minha gratidão pelo amor incondicional, pelo apoio constante e pela paciência inabalável ao longo desse percurso tão importante. Aos meus filhos, Paulo e Vinícius, agradeço pela compreensão e pela inspiração diária que me motivaram a persistir e a acreditar na realização deste sonho.

Quero expressar minha profunda gratidão ao Prof. Lucídio Cabral, orientador, amigo e conselheiro, que esteve ao meu lado nos momentos mais difíceis, sempre com palavras de sabedoria, bondade e encorajamento. Sua dedicação e caráter exemplar foram fundamentais para o sucesso deste trabalho.

Um agradecimento especial ao Prof. Sandro Marden, cuja amizade e incentivo constantes impediram que eu desistisse deste projeto. Sua confiança em mim e suas palavras de motivação foram peças-chave para que eu continuasse firme na busca por este objetivo. Este trabalho é resultado do apoio e dedicação de todos vocês. Minha eterna gratidão a cada um.

UMA METAHEURÍSTICA ILS APLICADA AO PROBLEMA DE FORMAÇÃO DE CÉLULAS DE MANUFATURA

RESUMO

Neste trabalho foi desenvolvida uma abordagem aproximativa baseada na metaheurística ILS (Interated Local Search) combinada com a metaheurística VND (Variable Neighborhood Descent) para o problema de formação de célula de manufaturas que formam a espinha dorsal de sistema de célula de manufatura. A solução proposta explora diversas estruturas de vizinhanças e a capacidade de diversificação que o mecanismo de perturbação agrega ao ILS. Resultados computacionais demonstram que o algoritmo ILS-VND-RD proposto tem ótimo desempenho e alcança algumas das melhores soluções existentes na literatura de formação de células de manufatura.

Palavras chaves - Formação de células de manufatura, Metaheurística, ILS (Interated Local Search), VND (Variable Neighborhood Descent)

A METAHEURISTIC ILS APPLIED TO THE PROBLEM OF MANUFACTURING CELL FORMATION

ABSTRACT

In this work, an approximative approach was developed based on the ILS (Iterated Local Search) metaheuristic combined with the Variable Neighborhood Descent (VND) metaheuristic for the cell formation problem in manufacturing systems, which form the backbone of manufacturing cell systems. The proposed solution explores various neighborhood structures and the diversification capability that the perturbation mechanism adds to ILS. Computational results demonstrate that the proposed ILS-VND-RD algorithm performs excellently and achieves some the best solutions existing in the manufacturing cell formation literature.

Keywords - Manufacturing cell formation, Metaheuristics, ILS (Iterated Local Search), VND (Variable Neighborhood Descent)

SUMÁRIO

CAPÍTULO I.....	11
1. INTRODUÇÃO.....	11
1.1 CONCEITOS CHAVES.....	13
1.3 ESCOPO DA PESQUISA.....	14
1.3 OBJETIVO GERAL.....	14
1.4 OBJETIVOS ESPECÍFICOS.....	15
1.5 DESCRIÇÃO DE UMA INSTÂNCIA.....	15
1.6 JUSTIFICATIVA.....	16
CAPÍTULO II.....	19
2. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS.....	19
2.1 FORMULAÇÃO MATEMÁTICA.....	22
2.1 MODELO MILP.....	24
CAPÍTULO III.....	26
3. ABORDAGEM PROPOSTA BASEADA EM ILS E VND-RD.....	26
3.1 METAHEURÍSTICA ILS.....	26
3.2 DESCIDA EM VIZINHAÇA VARIÁVEL.....	28
3.3 METAHEURÍSTICA HÍBRIDA ILS-VND-RD.....	31
3.3.1 Estruturas de Vizinhaça usadas na Busca Local.....	32
3.3.2 Estruturas de Vizinhaça com mecanismos de Perturbação.....	34
CAPÍTULO IV.....	37
4. RESULTADOS E DISCUSSÕES.....	37
4.1 AMBIENTE DE TESTES E INSTÂNCIAS.....	37
4.2 RESULTADOS FINAIS.....	40
4.3 COMPARAÇÃO COM O ESTADO DA ARTE.....	43
CAPÍTULO V.....	46
5. CONCLUSÕES.....	46
REFERÊNCIAS.....	47

LISTA DE FIGURAS

Figura 1.1 – Matriz de incidência inicial peça máquina.....	12
Figura 1.2 – Matriz de agrupamento de máquinas e peças em forma de cluster.....	13
Figura 1.3 – Sistema de produção representado por uma matriz máquina-parte	15
Figura 1.4 – Matriz que representa um SCM obtida a partir da Figura 1.3.....	16
Figura 3.1 – Pseudocódigo ILS	27
Figura 3.2: Representação esquemática do funcionamento do ILS.....	28
Figura 3.3 - Algoritmo VND	29
Figura 3.4 - Algoritmo VND usando descida randômica	30
Figura 3.5 - Algoritmo ILS-VND-RD	32
Figura 3.6 - Movimento MOVER_PARTE	33
Figura 3.7 - Movimento MOVER_MÁQUINA	33
Figura 3.8 - Movimento SWAP_PARTES	33
Figura 3.9 - Movimento SWAP_MÁQUINAS	33
Figura 3.10 - Movimento UNIR_CLUSTERS	34
Figura 3.11 - Movimento SPLIT_CLUSTER	35
Figura 3.12 - Estruturas de vizinhança Cadeia de Ejeção de PARTES	36
Figura 3.13 - Estruturas de vizinhança Cadeia de Ejeção de MÁQUINAS	36
Figura 4.1 – Variação da Solução conjunto A	41
Figura 4.2 – Variação da Solução conjunto B	43

LISTA DE TABELAS

Tabela 2.1 – Resumo dos trabalhos apresentados na revisão bibliográfica.....	21
Tabela 4.1 – Descrição do Conjunto A de Instâncias teste da literatura	38
Tabela 4.2 – Descrição do Conjunto B de Instâncias teste da literatura	39
Tabela 4.3 – Resultado abordagem ILS-VND-RD conjunto A	40
Tabela 4.4 – Resultado abordagem ILS-VND-RD conjunto B	42
Tabela 4.5 – Comparativo Literatura X ILS-VND-RD conjunto A	44
Tabela 4.6 – Comparativo Literatura X ILS-VND-RD conjunto B	45

CAPÍTULO I

1. INTRODUÇÃO

O projeto de sistemas de células de manufatura (SCMs) tem recebido a atenção crescente entre pesquisadores e profissionais das áreas de engenharias e computação (SOLEYMANPOUR M, VRAT P, SHANKAR R, 2002). O SCM decompõe um sistema de manufatura em subsistemas para gerenciá-lo mais facilmente. Em tal sistema, as partes ou peças que possuem processos similares são divididas em clusters chamados famílias de partes e as máquinas correspondentes a essas famílias de partes formam células de manufatura. As vantagens de tais procedimentos de agrupamento incluem as melhorias nos tempos de configuração, trabalhos em andamento, tempos de produção, custos de transporte, qualidade do produto e agendamento (WEMMERLÖV U, HYER NL, 1989). O problema de formação de células de manufatura é o passo inicial para projeto de sistemas de manufatura baseados em células de partes (peças) e máquinas. Este problema visa minimizar os custos de transporte de partes entre células, otimizando eficiência dos clusters (células) (LEI D, WU Z, 2005). Em outras palavras, busca-se evitar que uma parte necessite utilizar máquinas que estejam em outra célula. De forma análoga deve-se evitar custos de duplicação de máquina, ou seja, mesma máquina processando em clusters diferentes.

O problema de formação de células é um problema de otimização NP-difícil, (GAREY, M. R., JOHNSON D. S., 1979) o que significa que o tempo de execução para encontrar a solução ótima aumenta exponencialmente com o tamanho do problema (BALLAKUR A, STEUDEL HJ, 1987). Como resultado, o desenvolvimento de métodos de solução eficazes tem sido foco dos pesquisadores nas últimas décadas. Na literatura atual encontramos abordagens exatas e aproximadas para a solução do problema de formação de células, tais com: programação linear inteira, heurísticas, metaheurísticas, metaheurísticas híbridas, e metodologias de inteligência artificial (PAPAIOANNOU G, WILSON JM, 2010). Devido ao excelente

desempenho de meta-heurísticas na resolução de problemas combinatórios, um grande número de tais abordagens foi desenvolvido para resolver o problema de formação de células (NOKTEHDAN A, KARIMI B, HUSSEINZADEH KASHAN A, 2010), tais como: Simulated Annealing (SA), Busca Tabu, Algoritmo Genético, otimização baseada em enxames (Swarm Intelligence) e a combinação deles (MIRZAPOUR AL-E-HASHEM SMJ, ARYANEZHAD MB, JABBARZADEH A, 2011).

Uma das meta-heurísticas populares aplicadas ao problema do SCM são as redes neurais artificiais (POTOČNIK P, BERLEC T, STARBEK M, GOVEKAR E, 2013). Trata-se de computação redes inspiradas em sistemas nervosos biológicos e tentativas para simular sua complexa arquitetura de aprendizagem. Artificial as redes neurais são capazes de resolver problemas complexos de maneira eficaz devido ao processamento paralelo e de alta velocidade, à implementação de hardware, à memória associativa, à robustez na aproximação e às vantagens computacionais das redes neurais (SOLEYMANPOUR M, VRAT P, SHANKAR R, 2002).

Na Figura 1.1 apresentamos uma instância do problema de formação de células de manufatura considerando um conjunto com 5 máquinas (M1, M2 ...M5) e um conjunto com 5 partes (P1, P2 ...P5).

	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	0	1	1	0	1
M3	1	0	0	1	0
M4	0	1	1	0	1
M5	1	0	1	1	0

Figura 1.1 – Matriz de incidência inicial peça máquina

Na Figura 1.2 exibimos uma solução para a instância acima, com a formação de 2 clusters (células), sendo que na primeira célula temos as máquinas (M1, M5, M3) e as partes (P1, P4), e na segunda célula temos máquinas (M4, M2) e as partes (P3, P5 P2) sendo está a solução ótima para esta instância quando consideramos uma métrica de eficiência de agrupamento máximo (MAHMOODIAN, VAHID ET AL, 2017).

	P1	P2	P3	P4	P5
M1	1	1	0	0	0
M2	1	1	0	0	0
M3	1	1	0	0	0
M4	0	0	1	1	1
M5	0	0	1	1	1

Figura 1.2 – Matriz de agrupamento de máquinas e peças em forma de cluster

1.1. CONCEITOS CHAVES

Na visão geral fornecida abaixo, são explicados os conceitos relevantes e a terminologia utilizada nesta pesquisa.

Heurística: um algoritmo é considerado um método heurístico quando não há conhecimento matemático completo sobre seu comportamento. Em outras palavras, o algoritmo visa resolver problemas complexos usando uma ideia guia, chamada heurística, e uma quantidade razoável de recursos, como tempo, para encontrar soluções satisfatórias (SUCUPIRA, I. R., 2004). Geralmente, as heurísticas são aplicadas a problemas NP-difíceis, onde não existem métodos conhecidos que forneçam garantias, mas ainda assim produzem resultados de bom desempenho.

Metaheurística: é um conjunto de conceitos que pode ser utilizado para definir métodos heurísticos aplicáveis a uma ampla gama de problemas diferentes. Em outras palavras, é uma estrutura algorítmica geral que pode ser adaptada a diferentes problemas de otimização com relativamente poucas modificações específicas (SUCUPIRA, I. R., 2004). Os exemplos mais conhecidos de metaheurísticas incluem: algoritmos evolutivos, algoritmos genéticos, GRASP, otimização por colônia de formigas, busca tabu, busca local iterativa, recozimento simulado, entre outros.

Fabricação Celular: é uma lógica de produção que envolve a formação e operação de grupos de máquinas que têm permissão para processar apenas uma determinada coleção de peças ou pedidos (XUE E OFFODILE, 2020). Para explorar todos os benefícios de um Sistema de Manufatura Celular (CMS), quatro problemas devem ser abordados (WEMMERLOV E HYER, 1986). A Programação e Controle da Produção (PCP) é um desses quatro problemas e consiste na formação das células de produção que determinam a estrutura principal do sistema de produção.

Técnicas de otimização para CFP: como o Problema de Formação de Células (CFP) é um problema de otimização, é necessário selecionar um método adequado para resolvê-lo. Existem diversas técnicas disponíveis para resolver problemas de otimização.

Em um nível mais amplo, essas técnicas podem ser divididas em duas classes: “Técnicas Convencionais” e “Técnicas Não Convencionais” (MUKHERJEE E RAY, 2006). A principal diferença entre essas duas classes é que as técnicas não convencionais utilizam métodos heurísticos ou meta-heurísticos. Embora esses métodos possam convergir para soluções quase ótimas, eles têm a desvantagem de serem menos eficientes do que as técnicas convencionais. No entanto, as técnicas de pesquisa empregadas nas técnicas não convencionais permitem resolver problemas maiores em um tempo de computação razoável (MUKHERJEE E RAY, 2006).

1.2. ESCOPO DA PESQUISA

O Problema de Formação de Células (CFP) é um desafio de otimização (PAPAIIOANNOU G, WILSON JM, 2010, 2010). Na literatura atual, existem vários fluxos de pesquisa, cada um focado em otimizar um objetivo específico. Esses objetivos incluem a minimização de gargalos (MAHDAVI et al., 2009), a redução do número de espaços vazios (BYCHKOV E BATSYN, 2018) e a diminuição dos custos totais de operação de um Sistema de Manufatura Celular (CMS) (SULE E SURESH, 2009). Alguns poucos estudos abordam diretamente a minimização dos tempos de processamento das peças em máquinas (MUKATTASH et al., 2002). Além disso, esses estudos muitas vezes consideram apenas um conjunto limitado de características de design, o que limita sua aplicabilidade prática.

Nessa dissertação, o objetivo é resolver o problema da PCP de maneira monobjetiva, buscando a maximização da eficiência de agrupamento. Para tanto serão desenvolvidos algoritmos aproximativos baseados em metaheurísticas, em particular, as abordagens da metaheurística ILS (Interated Local Search) combinada com a metaheurística VND (Variable Neighborhood Descent).

1.3. OBJETIVO GERAL

O objetivo geral deste trabalho é desenvolver algoritmos aproximativos para o CFP utilizando as metaheurísticas ILS (Interated Local Search) combinada com a metaheurística VND (Variable Neighborhood Descent) para resolver o problema de formação de células de manufatura com função monobjetiva de eficiência de agrupamento.

1.4. OBJETIVOS ESPECÍFICOS

Esta dissertação teve como objetivos específicos:

- Construir um pequeno survey com as referências que compõem o estado da arte na área dos sistemas de célula de manufatura (SCM);
- Implementar as metaheurísticas ILS e VND explorando novas estruturas de vizinhança;
- Estudar o comportamento de assintótico de convergência dos algoritmos ILS e VND propostos;
- Comparar os resultados computacionais, com os resultados da literatura utilizando um conjunto de instâncias testes padrão para esta área.

1.5. DESCRIÇÃO DE UMA INSTÂNCIA

Para exemplificar a formação de células de manufatura, consideremos um sistema de produção composto por 10 máquinas e 15 partes. Esse sistema é representado por uma matriz de adjacência binária máquinas-partes, denotada como $A(i,j)$. Cada elemento a_{ij} é igual a 1 se a máquina (i) executa uma operação de manufatura sobre a parte (j), e 0 (ou em branco) caso contrário.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
M1					1				1	1			1		
M2	1			1				1			1				
M3					1				1	1			1		
M4	1			1				1			1				
M5		1	1			1	1					1		1	1
M6					1				1	1			1		
M7		1	1	1		1	1					1			
M8					1				1	1			1		
M9					1				1	1		1	1		1
M10		1	1			1	1					1		1	

Figura 1.3 – Sistema de produção representado por uma matriz máquina-parte

A formação de células de manufatura ocorre por meio do rearranjo das linhas e colunas da matriz de adjacência, criando blocos de células/máquinas e partes/famílias ao longo da diagonal da matriz.

	P5	P9	P10	P13	P1	P4	P8	P11	P2	P3	P6	P7	P12	P14	P15
M1	1	1	1	1											
M3	1	1	1	1											
M6	1	1	1	1											
M8	1	1	1	1											
M9	1	1	1	1									1		1
M2					1	1	1	1							
M4					1	1	1	1							
M5									1	1	1	1	1	1	1
M7						1			1	1	1	1	1		
M10									1	1	1	1	1	1	

Figura 1.4 – Matriz que representa um SCM obtida a partir da Figura 1.3

Na Figura 1.4, ilustramos uma possível divisão do sistema de produção da Figura 1.3 em três clusters de células/máquinas-famílias/partes. As células de máquinas são compostas da seguinte forma: a célula 1 inclui as máquinas 1, 3, 6, 8 e 9; a célula 2, as máquinas 2 e 4; e a célula 3, as máquinas 5, 7 e 10. As famílias de partes são distribuídas da seguinte maneira: a família 1 abrange as partes 5, 9, 10 e 13; a família 2, as partes 1, 4, 8 e 11; e a família 3, as partes 2, 3, 6, 7, 12, 14 e 15.

É importante que os clusters de células de máquinas e famílias de partes sejam independentes entre si, o que significa que devem ter poucos elementos inter-clusters (elementos na matriz iguais a 1 que não pertencem a nenhum cluster) na matriz de blocos diagonais. Esses elementos indicam partes que requerem mais de uma célula de máquina para serem completamente manufaturadas, resultando em movimento de material entre diferentes clusters de produção. No exemplo da Tabela 4, identificamos três elementos inter-clusters. Além disso, é desejável que haja poucas lacunas (elementos da matriz iguais a 0) dentro dos clusters, pois elas indicam subutilização de máquinas. Na mesma figura, observamos três lacunas no terceiro cluster.

1.6. JUSTIFICATIVA

A gestão da produção é um fator crucial para o sucesso de uma indústria. Seu objetivo é organizar o ambiente de produção de forma a otimizar custos e tempo, sem comprometer a qualidade. Em indústrias com baixa variedade e alto volume de produção, é comum estruturar o ambiente em linhas de produção. Cada linha é composta por máquinas dedicadas exclusivamente à fabricação de um único produto, atendendo à alta demanda.

Por outro lado, em indústrias com grande variedade e volume médio de produção, as máquinas não precisam ser exclusivas para um único produto. Nesse cenário, uma abordagem diferente é necessária. Uma estratégia é formar grupos (clusters) de máquinas com funcionalidades semelhantes (como grupos de tornos ou fresadoras), criando departamentos especializados. Assim, uma peça que requer operações em diferentes tipos de máquinas passará por todos os grupos que contêm as máquinas necessárias para sua manufatura completa. Isso gera um grande volume de material a ser movimentado entre os grupos de máquinas, aumentando o tempo de produção e dificultando o controle e a manutenção do sistema.

Nas últimas décadas, surgiu um novo modelo de organização: a manufatura celular. Nesse sistema, máquinas com diferentes funcionalidades são agrupadas em células dedicadas à produção de famílias de partes com similaridades, especialmente em relação às máquinas necessárias para fabricação. O sistema é dividido em clusters formados por células de máquinas e famílias de partes, também conhecidos como células de manufatura.

Uma das primeiras técnicas para formar um sistema de manufatura celular foi proposta por (BURBIDGE, J. L., 1971), no trabalho "Análise do Fluxo de Produção". Essa técnica é essencial para organizar sistemas de produção com volume médio e grande variedade de produtos. A Análise do Fluxo de Produção envolve etapas como levantamento dos recursos de produção (máquinas, ferramentas e profissionais), formação de células de manufatura, escalonamento das operações das partes em cada célula e o projeto do leiaute físico das células de máquinas na fábrica. A obtenção de células de manufatura é uma etapa fundamental na criação de sistemas baseados em manufatura celular.

A formação de clusters de células de máquinas e famílias de partes oferece várias vantagens para a gestão da produção:

- a) Redução do transporte de material: As máquinas necessárias para manufatura de uma parte estão agrupadas no mesmo cluster, minimizando o transporte interno.
- b) Redução do tempo de produção: O sistema se torna mais eficiente, aumentando a capacidade de produção.
- c) Simplificação do gerenciamento: O sistema é dividido em subsistemas independentes, facilitando o controle.
- d) Facilitação do escalonamento: As atividades são planejadas separadamente em cada cluster.

- e) Aumento da segurança: Menos manipulação de material no ambiente de produção.

Um sistema de manufatura celular ideal é composto por clusters sem lacunas e sem interdependência entre si. De acordo com (BAKER R. P., MAROPOULOS P. G., 1997), essa é a primeira meta a ser alcançada na implementação prática de um sistema de manufatura celular em uma indústria. No entanto, na prática, é raro encontrar sistemas de produção que consigam criar clusters totalmente independentes. Diante disso, muitos pesquisadores têm buscado resolver a interdependência entre clusters por meio da duplicação de máquinas responsáveis por elementos excepcionais (LOGENDRAN, R., PUVANUNT, V., 1997) (por exemplo, na Figura 1.4, a máquina 9 seria duplicada para o terceiro cluster e a máquina 7 para o segundo) ou pela terceirização de partes (no mesmo exemplo, as partes 4, 12 e 15 seriam terceirizadas). Em situações reais, a decisão de duplicar máquinas ou terceirizar partes envolve questões monetárias que devem ser avaliadas e decididas pelo projetista do sistema de manufatura. Além disso, existem outras considerações inerentes ao sistema de produção, como a capacidade das máquinas, o tempo de duração de cada operação de manufatura e os custos associados a cada operação. A abordagem apresentada neste trabalho para a formação de células de manufatura, no entanto, não aborda a duplicação de máquinas ou a terceirização de partes

Esta dissertação está organizada em cinco capítulos. Este primeiro capítulo apresenta uma motivação inicial sobre o trabalho, apresentando conceitos relevantes como o problema de formação de células, descrição de uma instância e são descritos o escopo da pesquisa e os objetivos do presente trabalho. No segundo capítulo é apresentada uma fundamentação teórica sobre o problema de formação de células e a formulação matemática para o cálculo da eficiência de agrupamento.

O terceiro capítulo apresenta a fundamentação teórica sobre a metaheurística utilizada para resolução de problemas de formação de células, em especial a metaheurística ILS-VND-RD. No quarto capítulo é apresentada as instâncias de trabalho, e os resultados obtidos com a metaheurística utilizada. No quinto capítulo são apresentadas as considerações finais desse trabalho e descritas algumas propostas de trabalhos futuros. Por fim, são apresentadas as referências bibliográficas utilizadas na composição desse trabalho.

CAPÍTULO II

2. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

A formação de células de máquinas e peças (*Cell Formation Problem*)(CFP) é um dos principais desafios de otimização na manufatura. O CFP busca agrupar máquinas e peças em células de produção de forma a maximizar a eficiência e reduzir os custos operacionais, principalmente minimizando a movimentação de peças entre células. Este problema é classicamente formulado através de uma matriz binária, onde as linhas representam máquinas e as colunas representam peças. O objetivo central é aumentar as operações dentro das células e diminuir os movimentos entre células, conforme descrito por (KUMAR, K. R., & CHANDRASEKHARAN, M. P. 1990).

Diversos modelos foram desenvolvidos para abordar o CFP, variando desde simples matrizes binárias até complexas formulações de Programação Linear Inteira Mista (MILP). O modelo MILP proposto por (BYCHKOV, I., BATSYN, M., SUKHOV, P., PARDALOS, P.M., 2013) se destaca por resolver a maioria das instâncias de problemas de forma otimizada. A eficácia do agrupamento, uma medida popular para o CFP, é calculada utilizando fórmulas que maximizam operações intra-células e minimizam operações inter-células.

A biclusterização tem sido explorada como uma técnica promissora para resolver o problema de agrupamento de peças e máquinas. Em seu trabalho (BOUTSINAS, B., 2013) discute a aplicação da biclusterização, que permite a clusterização simultânea de linhas e colunas de uma matriz de dados, criando submatrizes compostas por subconjuntos de linhas e colunas que são mais similares entre si. Esta abordagem se destaca por mitigar problemas comuns como movimentos intercelulares excessivos e subutilização de máquinas

O CF-GGA, um algoritmo genético especializado, é apresentado como uma solução eficaz para o CFP. Os autores (BROWN, E.C., SUMICHRAS, R.T., 2008) destacam que

este algoritmo se foca na diagonalização do bloco da matriz de incidência máquina-peça, mostrando melhorias significativas em comparação com outros métodos documentados. Já (GONÇALVES, J.F., RESENDE, M.G.C., 2004) também propõem uma abordagem combinando algoritmos genéticos com heurísticas de busca local, resultando em soluções com eficácia de agrupamento superior.

Quando (ELBENANI, B., FERLAND, J.A., 2012) introduzem a resolução exata do CFP utilizando o algoritmo de Dinkelbach, transformando o problema em uma programação fracionária linear binária e resolvendo-o com o otimizador IBM ILOG CPLEX 12.2. Este método validou que as melhores soluções conhecidas são iguais às soluções ótimas para 27 dos 35 problemas de benchmark.

Ao discutir as formulações alternativas do problema de formação de grupos de máquinas e peças (BOCTOR, F., 1991), propondo uma nova formulação linear zero-um que elimina a maioria das desvantagens observadas em outros modelos. Já (BRUSCO, M.J., 2015) apresenta um algoritmo exato para maximizar a eficácia de agrupamento, utilizando uma combinação de heurísticas e técnicas de branch-and-bound.

O artigo de (ASKIN, R.G., SUBRAMANIAN, S., 1987) propõe uma abordagem heurística para a configuração CFP, considerando custos de inventário, manuseio de materiais, configuração e processamento. A metodologia envolve três etapas: reordenar tipos de peças com base na similaridade de roteamento, combinar tipos de peças adjacentes para reduzir a necessidade de máquinas e agrupar onde os benefícios econômicos de utilização superam os custos

Estudos empíricos têm demonstrado a eficiência das técnicas propostas. (ADIL, G.K., RAJAMANI, D., STRONG, D., 1996) desenvolvem um modelo que minimiza a soma ponderada de vazios e elementos excepcionais, utilizando um algoritmo de resfriamento simulado para resolver problemas maiores. (KUMAR, K.R., KUSIAK, A., VANNELLI, A., 1986) abordam a problemática do agrupamento de peças e componentes em sistemas de manufatura flexíveis, propondo algoritmos que simplificam significativamente os processos de planejamento e programação de produção

A manufatura celular (CM) é uma estratégia amplamente discutida na literatura devido às suas vantagens sobre os sistemas de manufatura em oficinas tradicionais. Em seu trabalho (SEIFODDINI, H., DJASSEMI, M., 1996) destacam que a CM supera a manufatura em oficinas ao reduzir tempos de setup, inventários em processo e tempos de ciclo.

A literatura sobre formação de células de máquinas e peças na manufatura oferece uma variedade de abordagens, desde técnicas heurísticas até métodos exatos, cada uma com suas próprias vantagens e desvantagens. A escolha da metodologia depende das características específicas do problema e dos objetivos do sistema de produção

Tabela 2.1 – Resumo dos trabalhos apresentados na revisão bibliográfica

Autor	Ano	Descrição
KUMAR, K. R., & CHANDRASEKHARAN, M. P.	1990	Introduziu um modelo para cálculo da eficácia de agrupamento
BYCHKOV, I., BATSYN, M., SUKHOV, P., PARDALOS, P.M..	2013	Propôs um modelo MILP (Programação Linear Inteira Mista)
BOUTSINAS, B s	2013	Propôs uma técnica de biclusterização
BROWN, E.C., SUMICHRAST, R.T.	2008	Fez uso de um algoritmo genético especializado
GONÇALVES, J.F., RESENDE, M.G.C	2004	Abordagem combinando algoritmos genéticos com heurísticas de busca local
ELBENANI, B., FERLAND, J. A	2012	Introduzem a resolução exata do CFP utilizando o algoritmo de Dinkelbach
BOCTOR, F.,	1991	Propôs uma nova formulação linear zero-um
ASKIN, R.G., SUBRAMANIAN, S.,	1987	Propõe uma abordagem heurística
ADIL, G.K., RAJAMANI, D., STRONG, D.	1996	Faz uso de um algoritmo de resfriamento simulado (SA)
KUMAR, K.R., KUSIAK, A., VANNELLI, A.	1986	abordam a problemática do CFP em sistemas de manufatura flexíveis

2.1. FORMULAÇÃO MATEMÁTICA

Os sistemas de manufatura celular são projetados para processar peças semelhantes dentro da mesma célula de produção, equilibrar a carga de trabalho das máquinas e minimizar o movimento das peças de uma célula para outra durante o processo de produção. O objetivo mais popular para o problema de formação de células é a eficácia de agrupamento, introduzida por (KUMAR, K. R., & CHANDRASEKHARAN, M. P. 1990).

$$\tau = \frac{n_1^{in}}{n_1 + n_o^{in}}$$

Onde:

n_1 – Número total de operações (uns) na matriz máquinas-partes

n_1^{in} – Número de operações realizadas dentro das células

n_o^{in} – Número de vazios (zeros dentro das células)

Em comparação com outros objetivos, a função de eficácia de agrupamento aborda a melhor estrutura em bloco-diagonal das soluções do problema de formação de células (SARKER, B.R., 2001).

Na literatura, podem ser encontradas várias restrições relacionadas ao tamanho mínimo de uma célula. As três considerações mais populares são as seguintes:

- Permitir células residuais (células contendo apenas máquinas ou apenas peças).
- Permitir singletons (células com uma máquina e várias peças ou vice-versa) e proibir células residuais.
- Permitir apenas células com pelo menos 2 máquinas e 2 peças.

Uma das opções mais populares é permitir "singletons" (células com apenas uma máquina ou apenas uma peça) e proibir células residuais. Nesta seção, consideramos a formulação clássica, onde supomos que "singletons" podem aparecer nas soluções, enquanto células residuais são proibidas. Sendo este, o modelo utilizado em nossos experimentos computacionais.

Um modelo direto de programação fracionária inteira (IFP) para o problema de formação de células com a função objetivo de eficácia de agrupamento permitindo singletons e proibindo células residuais é dado abaixo. Usamos a seguinte notação: m é o número de máquinas, p é o número de partes, a_{ij} é igual 1 se a máquina i processa a parte

j e c é o número máximo possível de células de produção. Como cada célula de produção deve conter pelo menos uma máquina e uma parte definimos $c = \min(m, p)$

Variáveis de decisão (Modelo IFP):

$$x_{ik} = \begin{cases} 1, & \text{se a máquina } i \text{ pertence à célula } k \\ 0, & \text{caso contrário} \end{cases}$$

$$y_{jk} = \begin{cases} 1, & \text{se a parte } j \text{ pertence à célula } k \\ 0, & \text{caso contrário} \end{cases}$$

$$\max = \frac{\sum_{i=1}^m \sum_{j=1}^p \sum_{k=1}^c a_{ij} x_{ik} y_{jk}}{\sum_{i=1}^m \sum_{j=1}^p a_{ij} + \sum_{i=1}^m \sum_{j=1}^p \sum_{k=1}^c (1 - a_{ij}) x_{ik} y_{jk}} \quad (1)$$

$$\sum_{k=1}^c x_{ik} = 1 \quad i = 1, \dots, m \quad (2)$$

$$\sum_{k=1}^c y_{jk} = 1 \quad j = 1, \dots, p \quad (3)$$

$$\sum_{k=1}^m x_{ik} \leq m \cdot \sum_{j=1}^p y_{jk} \quad k = 1, \dots, c \quad (4)$$

$$\sum_{j=1}^p y_{jk} \leq p \cdot \sum_{i=1}^m x_{ik} \quad k = 1, \dots, c \quad (5)$$

A função objetivo (1) é a medida de eficácia do agrupamento onde o numerador é o número de uns dentro dos clusters (n_1^{in}) e os somatórios no denominador são o número total de uns (n_1) e o número de zeros dentro dos clusters (n_0^{in}) respectivamente. As restrições (2) e (3) exigem que cada máquina e cada peça sejam atribuídas exatamente a um cluster de produção. As desigualdades (4) e (5) proibem clusters residuais, isto é, sem máquinas ou partes. A parte esquerda de (4) é o número total de máquinas atribuídas à um cluster específico (esta soma não é maior que m) e a parte direita é o número total de partes atribuídas a esse cluster (multiplicado por m). Isso significa que se tivermos pelo

menos uma máquina atribuída a algum cluster deve haver pelo menos uma parte atribuída a este cluster. Este modelo permite que tenhamos qualquer número de clusters na solução ótima. Por exemplo, se a solução ótima tiver apenas duas células, então as variáveis x_{ik} e y_{jk} será zero para todo k , exceto apenas dois valores de k .

2.2. MODELO MILP

Foi proposto por (BYCHKOV, I.S., BATSYN, M.V., PARDALOS, P.M., 2014), um modelo de programação linear inteira mista para o problema de formação de clusters, que possui grande semelhança com o modelo descrito na seção anterior. Sendo um dos pontos mais importantes a linearização da função objetivo de eficácia de agrupamento. A ideia anterior era linearizar a função objetivo de eficácia de agrupamento fixando o valor do denominador $n_1 + n_o^{in}$ e considerando uma faixa de todos os possíveis números de vazios n_o^{in} . O limite inferior para n_o^{in} é igual a 0 porque, geralmente, pode haver uma solução sem nenhum vazio. O limite superior é calculado usando a seguinte proposição.

Proposição 1 (BYCHKOV, I.S., BATSYN, M.V., PARDALOS, P.M., 2014). O número de vazios na solução ótima satisfaz a seguinte desigualdade:

$$n_o^{in} \leq \left\lfloor \frac{1 - \tau}{\tau} n_1 \right\rfloor$$

Onde τ é o valor de eficácia de agrupamento de qualquer solução viável.

Portanto, se conhecermos uma solução viável, podemos limitar o intervalo de valores possíveis para o número de vazios. Infelizmente, o desempenho desta abordagem depende fortemente da solução viável usamos para obter nossos limites. Desta forma, resolver problemas em situações em que o valor da eficácia do agrupamento é baixo exige muitos recursos computacionais (já que o número de subtarefas é muito grande) e às vezes não conseguimos resolver nem mesmo instâncias de formação de células de tamanho médio.

Foi proposto por (BYCHKOV, I.S., BATSYN, M.V., 2018), juntamente com o uso do método linear inteiro misto modelo, usamos outra forma de linearização – algoritmo Dinkelbach (DINKELBACH, W., 1967). A abordagem paramétrica introduzida por

Dinkelbach é uma das estratégias mais gerais e populares para programação fracionária. Reduz a solução de um problema de programação fracionária para a solução de uma sequência de problemas mais simples. Se considerarmos um modelo de programação fracionária com a seguinte função objetivo:

$$Q(x) = \frac{P(x)}{D(x)}$$

Então o procedimento de Dinkelbach é o seguinte:

Passo 1 Pegue alguma solução viável x^0 , calcule $\lambda_1 = \frac{P(x^0)}{D(x^0)}$ e seja $k = 1$

Passo 2 Resolva o problema original com a função objetivo $Q(x)$ substituindo por $F(\lambda_k) = P(x) - \lambda_k D(x) \rightarrow \max$ e seja x^k a solução ideal.

Passo 3 Se $F(\lambda_k)$ for igual a 0 (ou menor que alguma tolerância predefinida), interrompa o procedimento e retorne x^k como o valor ideal de solução.

Caso contrário, $k = k + 1$, $\lambda_k = \frac{P(x^k)}{D(x^k)}$ volte para o passo 2.

Também utilizaram a abordagem de Dinkelbach para linearização de medidas de eficácia de agrupamento (ELBENANI, B., FERLAND, J.A., 2012), seus tempos computacionais são bastante altos e os resultados são dados apenas para o número fixo específico de células de produção.

CAPÍTULO III

3. ABORDAGEM PROPOSTA BASEADA EM ITERATED LOCAL SEARCH E VND-RD

Neste trabalho, propomos uma abordagem aproximativa que hibridiza a metaheurística ILS com o VND-RD, utilizando este último para ampliar a fase de intensificação da busca local. O algoritmo resultante, denominado ILS-VND-RD, combina a busca local iterada com procedimentos de intensificação, focando em áreas onde a busca local isolada não consegue encontrar soluções vizinhas de melhor qualidade.

3.1. META-HEURÍSTICA ILS

O método *Iterated Local Search* (ILS) ou Busca Local Iterada (SOUZA, M.J.F, 2024) está fundamentado na ideia de que um procedimento de busca local pode ser aprimorado gerando novas soluções iniciais a partir de perturbações sobre uma solução ótima local, gerando assim um mecanismo com grande potencial de fuga de ótimos locais. A implementação padrão do algoritmo ILS contempla os seguintes quatro componentes que devem ser personalizados considerando o problema a ser solucionado, a saber: (a) Procedimento **GeraSolucaoInicial**, que produz uma solução inicial para o problema, denominada s_0 ; (b) Procedimento **BuscaLocal**, que explora as estruturas de vizinhança e retorna uma solução possivelmente melhorada s'' pertencente a uma dessas estruturas; (c) Procedimento **Perturbação**, que modifica “fortemente” a solução atual s , permitindo que se alcance uma solução intermediária s' ; e (d) Procedimento **CriterioAceitacao**, que decide qual solução será usada na próxima etapa de perturbação. A Figura 3.1 mostra o pseudocódigo de um algoritmo ILS básico.

```

procedimento ILS
1  $s_0 \leftarrow \text{GeraSolucaoInicial}();$ 
2  $s \leftarrow \text{BuscaLocal}(s_0);$ 
3 enquanto (os critérios de parada não estiverem satisfeitos) faça
4      $s' \leftarrow \text{Perturbacao}(\text{histórico}, s);$ 
5      $s'' \leftarrow \text{BuscaLocal}(s');$ 
6      $s \leftarrow \text{CritérioAceitacao}(s, s'', \text{histórico});$ 
7 fim-enquanto;
fim ILS;

```

Figura 3.1 – Pseudocódigo ILS (Souza, M.J.F., 2024)

O comportamento que possibilita sucesso na trajetória de busca da metaheurística ILS está na forma como este faz a amostragem de ótimos locais, juntamente com a escolha do método de busca local, das perturbações e do critério de aceitação. Em princípio, qualquer método de busca local pode ser utilizado, mas o desempenho do ILS em termos de qualidade da solução final e velocidade de convergência depende fortemente do método escolhido. Normalmente, usa-se um método de descida, mas também é possível aplicar algoritmos mais sofisticados, como Algoritmos Genéticos, *Simulated Annealing* e Busca Tabu ou outras meta-heurísticas puras ou híbridas.

O mecanismo de perturbação tem por objetivo facilitar a fuga de ótimos locais e reiniciar a busca local em diferentes regiões, todavia deve ser calibrado para ter força suficiente para escapar, mas sem perder as principais características da região onde se encontra o ótimo local atual. O critério de aceitação é usado para atualizar a melhor solução e definir a partir de qual solução continuará o processo de intensificação de busca local. Já a fase de diversificação é viabilizada pela aplicação dos mecanismos de perturbação. A intensificação consiste em explorar mais efetivamente a região atual do espaço de busca, enquanto a diversificação consiste em mover-se para outras regiões do espaço de soluções. Normalmente, o critério de aceitação comumente utilizado é mover-se para o ótimo local s'' somente se ele for melhor que o ótimo local atual s , ou seja, se $f(s'') < f(s)$ no caso de um problema de minimização, ou se $f(s'') > f(s)$ no caso de um problema de maximização.

A Figura 3.2 ilustra o funcionamento do método ILS em um problema de minimização. A partir de um ótimo local s , faz-se uma perturbação que conduz a uma solução intermediária s' . Após a aplicação de um método de busca local a s' , produz-se um novo ótimo local s'' . Considerando como critério de aceitação o fato de $f(s'')$ ser melhor que $f(s)$, a busca prossegue a partir de s'' .

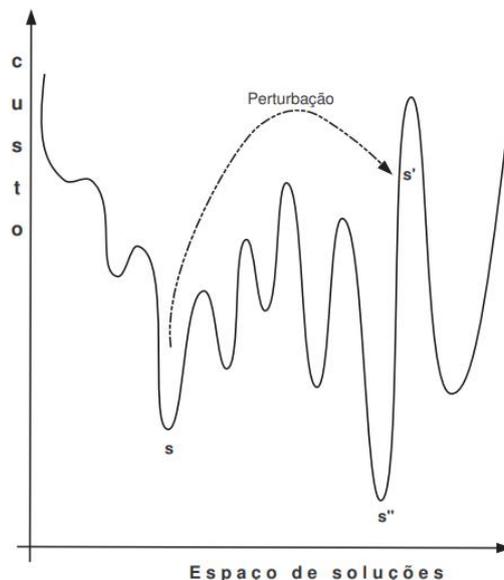


Figura 3.2: Representação esquemática do funcionamento do ILS (Souza, M.J.F., 2024)

3.2. DESCIDA EM VIZINHAÇA VARIÁVEL

O Método de Descida em Vizinhaça Variável (Variable Neighborhood Descent, VND), proposto por (MLADENOVIĆ, N., HANSEN, P., 1997), é um método de refinamento que consiste em explorar o espaço de soluções por meio de trocas sistemáticas de estruturas de vizinhaça, aceitando somente soluções de melhora da solução corrente e retornando à primeira estrutura quando uma solução melhor é encontrada. Segundo os autores, o método VND baseia-se em três princípios básicos:

- Um ótimo local com relação a uma dada estrutura de vizinhaça não corresponde necessariamente a um ótimo local com relação a uma outra estrutura de vizinhaça;
- Um ótimo global corresponde a um ótimo local para todas as estruturas de vizinhaça;
- Para muitos problemas, ótimos locais com relação a uma ou mais estruturas de vizinhaça são relativamente próximas.

Ainda de acordo os autores, o último princípio, de natureza empírica, indica que um ótimo local frequentemente fornece algum tipo de informação sobre o ótimo global. Este é o caso em que os ótimos local e global compartilham muitas variáveis com o mesmo valor, o que sugere uma investigação sistemática da vizinhaça de um ótimo local até a obtenção de uma nova solução de melhor valor. A metaheurística VND é uma poderosa

ferramenta de otimização que aproveita a diversidade de vizinhanças para explorar eficientemente o espaço de soluções. Sua robustez e adaptabilidade a tornam uma escolha popular para diversos problemas práticos em otimização combinatória.

O pseudocódigo deste método, em que se considera o refinamento de uma solução s utilizando uma função de avaliação f a ser minimizada e um conjunto de r diferentes vizinhanças $N = \{N^{(1)}, N^{(2)}, \dots, N^{(r)}\}$, é apresentado pela Figura 3.3. A ordem de exploração das r vizinhanças ($N(1), N(2), \dots, N(r)$) usualmente é definida de acordo com a sua complexidade de exploração.

Procedimento VND($f(\cdot), N(\cdot), r, s$)	
1	Seja r o número de estruturas diferentes de vizinhança;
2	$k \leftarrow 1$; {Tipo de estrutura de vizinhança corrente}
3	enquanto ($k \leq r$) faça
4	Encontre o melhor vizinho $s' \in N^{(k)}(s)$;
5	se ($f(s') < f(s)$)
6	então
7	$s \leftarrow s'$;
8	$k \leftarrow 1$;
9	senão
10	$k \leftarrow k+1$;
11	fim-se;
12	fim-enquanto;
13	Retorne s ;
	fim VND;

Figura 3.3 - Algoritmo VND

O objetivo da ordem de uso das estruturas de vizinhanças é começar usando os movimentos de menor complexidade e deixar para explorar por últimos aquelas de movimentos de maior complexidade. Na linha 4 da Figura 3.3 é acionado um mecanismo de busca pelo melhor vizinho que costuma demandar maior custo computacional.

Uma outra maneira de implementar o VND consiste em aplicar uma ordem aleatória ou randômica para acionar as estruturas de vizinhanças, sem repetição de utilização. Isso é chamado de VND-RD está ilustrado na Figura 3.4 abaixo:

Na figura 3.4, RD_{maxk} é o número máximo de vizinhos analisados na k -ésima vizinhança, considerando que algumas estruturas de vizinhanças podem, ter alta dimensão.

Procedimento VND-RD($f(\cdot)$, $N(\cdot)$, r , RD_{maxk} , s)

```

1  Seja  $r$  o número de estruturas diferentes de vizinhança;
2  Seja  $RD_{maxk}$  o número máximo de iterações na  $k$ -ésima vizinhança;
3   $k \leftarrow 1$ ; {Tipo de estrutura de vizinhança corrente}
4  enquanto ( $k \leq r$ ) faça
5      iterRD  $\leftarrow 1$ ;
6       $s' \leftarrow s$ ;
7      enquanto ( $iterRD \leq RD_{maxk}$ ) faça
8          Selecione, aleatoriamente, um vizinho  $s'' \in N(k)(s')$ ;
9          se ( $f(s'') < f(s')$ ) então
10              $s' \leftarrow s''$ ;
11         fim-se;
12         iterRD  $\leftarrow iterRD + 1$ ;
13     fim-enquanto;
14     se ( $f(s') < f(s)$ )
15         então
16              $s \leftarrow s'$ ;
17              $k \leftarrow 1$ ;
18     senão
19          $k \leftarrow k + 1$ ;
20     fim-se;
21 fim-enquanto;
22 Retorne  $s$ ;
fim VND-RD;
```

Figura 3.4 - Algoritmo VND usando descida randômica

Como visto no pseudocódigo da Figura 3.4, a metaheurística VND-RD pode ser resumida como um processo iterativo em que realiza a descida ou subida, respectivamente quando consideramos problemas de minimização ou maximização, em cada uma das vizinhanças. Para cada vizinhança: i) a metaheurística tenta encontrar uma solução

melhor a partir da solução atual; ii) se uma melhoria é encontrada, a nova solução substitui a anterior e o processo começa novamente a partir dessa nova solução; iii) se não há melhorias em uma vizinhança, a VND-RD muda para a próxima vizinhança. O algoritmo continua a explorar as vizinhanças até que tenham sido aplicada uma série predefinida de vizinhanças ou até que não sejam mais encontradas melhorias em nenhuma das vizinhanças disponíveis. Essa parada evita que o algoritmo fique preso em ciclos ou continue indefinidamente sem progresso.

3.3. METAHEURISTICA HIBRIDA ILS-VND-RD

Neste trabalho está sendo proposta uma abordagem aproximativa baseada na hibridização da metaheurística ILS com VND-RD, sendo esta última utilizada com um mecanismo de ampliação da fase de intensificação da busca local. O algoritmo proposto, doravante denominado ILS-VND-RD, combina então uma busca local iterada com procedimentos de intensificação em áreas onde a busca local sozinha não consegue encontrar soluções vizinhas de melhor qualidade.

No ILS-VND-RD, a construção da solução inicial e a busca local, bem como os procedimentos de perturbação visam maximizar a métrica de eficiência de agrupamento para problema de formação de células de manufatura.

O esboço do ILS-VND-RD é apresentado na Figura 3.5. O laço entre as linhas 3-7 tem inicialmente uma chamada ao mecanismo de perturbação, em seguida a busca local é feita pela metaheurística VND-RD. Ressalte-se que aqui a VND-RD tem um papel secundário, no sentido de que ela apenas substitui o mecanismo mais simples de busca local, sendo guiada pelo ILS. Nesse papel, o VND-RD pode ser interpretado como um mecanismo de busca local mais abrangente, pois naturalmente irá investigar um número maior de soluções de espaço de soluções do problema. Obviamente isto incorre em maior custo computacional.

```

procedimento ILS-VND-RD
1 s0 ←GeraSolucaoInicial();
2 s←BuscaLocal(s0);
3 enquanto (os critérios de parada não estiverem satisfeitos) faça
4     s' ← Perturbacao(histórico,s);
5     s'' ← VND-RD(s');
6     s ←CriterioAceitacao(s,s'',histórico);
7 fim-enquanto;
fim ILS-VND-RD;

```

Figura 3.5 - Algoritmo ILS-VND-RD

Para a implementação dos mecanismos de Busca Local, Perturbação e VND-RD foram construídas diversas estruturas de vizinhança que são descritas nas subseções a seguir.

3.3.1 Estruturas de Vizinhança usadas na Busca Local

O procedimento de busca local para o problema de formação de células de manufatura está baseado em seis estruturas de vizinhanças, a saber:

- Retirada de uma parte de um cluster e inserção em outro cluster – (MOVER_PARTE)
- Retirada de uma máquina de um cluster e inserção em outro cluster - (MOVER_MÁQUINA)
- Troca de partes entre clusters - (SWAP_PARTES)
- Troca de máquinas entre clusters – (SWAP_MÁQUINAS)
- União de dois clusters – (UNIR_CLUSTERS)
- Divisão de um cluster em dois clusters (SPLIT_CLUSTER)

As Figuras de 3.6 a 3.13, ilustram esses movimentos. Em cada Figura é mostrada a solução anterior e a solução obtida pela aplicação do respectivo movimento.

O movimento de MOVER_PARTE inter-clusters é ilustrado abaixo, inicialmente temos o cluster 1 (M1, M2, P1, P2) e o cluster 2 (M3, M4, M5, P3, P4, P5), aplicamos o movimento de mover a parte P4 do cluster 2 para o cluster 1. Essa nova configuração, cluster 1 (M1, M2, P1, P2, P4) e o cluster 2 (M3, M4, M5, P3, P5), pode ser visualizada na disposição das partes na matriz.

MOVER_PARTE (P4 CLUSTER 2 → CLUSTER 1)

	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	1	0	1	0	1
M3	1	0	1	1	1
M4	1	1	1	1	1
M5	0	1	1	1	0



	P1	P2	P4	P3	P5
M1	1	0	1	0	0
M2	1	0	0	1	1
M3	1	0	1	1	1
M4	1	1	1	1	1
M5	0	1	1	1	0

Figura 3.6 - Movimento MOVER_PARTE move uma parte de um cluster para outro

O movimento de MOVER_MÁQUINA inter-clusters é ilustrado abaixo, inicialmente temos o cluster 1 (M1, M2, P1, P2) e o cluster 2 (M3, M4, M5, P3, P4, P5), aplicamos o movimento de mover a máquina M4 do cluster 2 para o cluster 1. Essa nova configuração, cluster 1 (M1, M2, M4, P1, P2) e o cluster 2 (M3, M5, P3, P4, P5), pode ser visualizada na disposição das máquinas na matriz.

MOVER_MÁQUINA (M4 CLUSTER 2 → CLUSTER 1)

	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	1	0	1	0	1
M3	1	0	1	1	1
M4	1	1	1	1	1
M5	0	1	1	1	0



	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	1	0	1	0	1
M4	1	1	1	1	1
M3	1	0	1	1	1
M5	0	1	1	1	0

Figura 3.7 - Movimento MOVER_MÁQUINA move uma máquina de cluster para outro

O movimento de SWAP_PARTES P2 e P5 é ilustrado abaixo, inicialmente temos o cluster 1 (M1, M2, P1, P2) e o cluster 2 (M3, M4, M5, P3, P4, P5), trocamos as COLUNAS correspondentes a P2 e P5 na matriz inicial, tendo como resultado em uma nova configuração na matriz, clusters 1 (M1, M2, P1, P5) e o cluster 2 (M3, M4, M5, P3, P4, P2.)

SWAP_PARTES (P2, P5)

	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	1	0	1	0	1
M3	1	0	1	1	1
M4	1	1	1	1	1
M5	0	1	1	1	0



	P1	P5	P3	P4	P2
M1	1	0	0	1	0
M2	1	1	1	0	0
M3	1	1	1	1	0
M4	1	1	1	1	1
M5	0	0	1	1	1

Figura 3.8 - Movimento SWAP_PARTES que efetua a troca de partes inter-clusters

O movimento de SWAP_MÁQUINAS M2 e M4 é ilustrado abaixo, inicialmente temos o cluster 1 (M1, M2, P1, P2) e o cluster 2 (M3, M4, M5, P3, P4, P5), trocamos as LINHAS correspondentes a M2 e M4 na matriz inicial, tendo como resultado em uma nova configuração na matriz, cluster 1 (M1, M4, P1, P2) e o cluster 2 (M3, M2, M5, P3, P4, P5)

SWAP_MÁQUINAS (M2, M4)

	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M2	1	0	1	0	1
M3	1	0	1	1	1
M4	1	1	1	1	1
M5	0	1	1	1	0



	P1	P2	P3	P4	P5
M1	1	0	0	1	0
M4	1	1	1	1	1
M3	1	0	1	1	1
M2	1	0	1	0	1
M5	0	1	1	1	0

Figura 3.9 - Movimento SWAP_MÁQUINAS efetua a troca de máquinas inter-clusters

O movimento de UNIR_CLUSTERS e Unir clusters combina dois ou mais clusters (máquinas e partes) em um único cluster de maior dimensão. Inicialmente temos o cluster 1 (M1, M2, P1, P2), o cluster 2 (M3, M4, P3, P4) e o cluster 3 (M5, P5, P6). Ao realizarmos o movimento para unir os clusters 1 e 2, temos uma nova célula que contém todas as máquinas e partes de ambos os clusters. Observamos que a matriz permanece a mesma, mas a associação de máquinas e partes ao clusters mudou. Sendo nova configuração: cluster 1 (M1, M2, M3, M4, P1, P2, P3, P4) e cluster 2 (M5, P5, P6).

UNIR_CLUSTERS (1, 2)

	P1	P2	P3	P4	P5	P6
M1	1	0	0	1	0	0
M2	1	0	1	0	1	0
M3	1	0	1	1	1	0
M4	1	1	1	1	1	1
M5	0	1	1	1	0	1



	P1	P2	P3	P4	P5	P6
M1	1	0	0	1	0	0
M2	1	0	1	0	1	0
M3	1	0	1	1	1	0
M4	1	1	1	1	1	1
M5	0	1	1	1	0	1

Figura 3.10 - Movimento UNIR_CLUSTERS que junta dois clusters em um único cluster

O movimento de SPLIT_CLUSTER, realiza a divisão de um cluster em dois ou mais clusters de menor dimensão. Inicialmente temos o cluster 1 (M1, M2, P1, P2), o cluster 2 (M3, M4, M5, M6, P3, P4, P5, P6) e o cluster 3 (M7, M8, P7, P8). Ao realizarmos o movimento para dividir o clusters 2, observamos que a matriz permanece a mesma, mas a associação de máquinas e partes ao clusters mudou. Sendo nova configuração: cluster 1 (M1, M2, P1, P2), cluster 2 (M3, M4, P3, P4), cluster 3 (M5, M6, P5, P6), cluster 4 (M7, M8, P7, P8).

SPLIT_CLUSTER (2) → (2,3)

	P1	P2	P3	P4	P5	P6	P7	P8
M1	1	0	0	1	0	0	0	1
M2	1	0	1	0	1	0	0	0
M3	1	0	1	1	1	0	0	0
M4	1	1	1	1	1	1	0	1
M5	0	1	1	1	0	1	0	0
M6	0	0	1	0	1	0	1	1
M7	0	0	0	0	0	0	1	1
M8	1	0	0	1	0	1	1	0



	P1	P2	P3	P4	P5	P6	P7	P8
M1	1	0	0	1	0	0	0	1
M2	1	0	1	0	1	0	0	0
M3	1	0	1	1	1	0	0	0
M4	1	1	1	1	1	1	0	1
M5	0	1	1	1	0	1	0	0
M6	0	0	1	0	1	0	1	1
M7	0	0	0	0	0	0	1	1
M8	1	0	0	1	0	1	1	0

Figura 3.11 - Movimento SPLIT_CLUSTER dividi um cluster em dois novos clusters

3.3.2 Estruturas de Vizinhaça com mecanismos de Perturbação

Para implementar a fase de perturbação foram desenvolvidas as seguintes estruturas de vizinhaça:

- Cadeia de Ejeção de Máquinas — Desloca uma máquina de um cluster para o cluster seguinte, até que no último cluster uma máquina seja deslocada para o primeiro cluster.
- Cadeia de Ejeção de Partes — Desloca uma parte de um cluster para o cluster seguinte, até que no último cluster uma parte seja deslocada para o primeiro cluster.

Esses movimentos são ilustrados nas Figuras 3.12 e 3.13, com a apresentação da solução anterior e da solução posterior obtida pela aplicação do respectivo movimento.

Cadeia de Ejeção de PARTES (P1, P4, P5, P8)

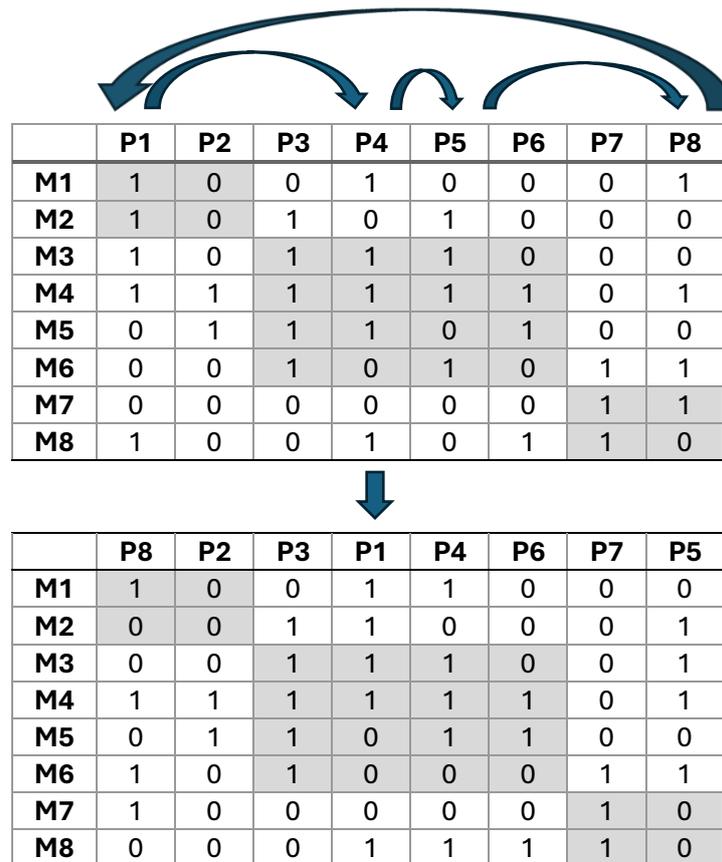


Figura 3.12 - Estruturas de vizinhança Cadeia de Ejeção de PARTES

Cadeia de Ejeção de Máquinas (M2, M4, M6, M7)

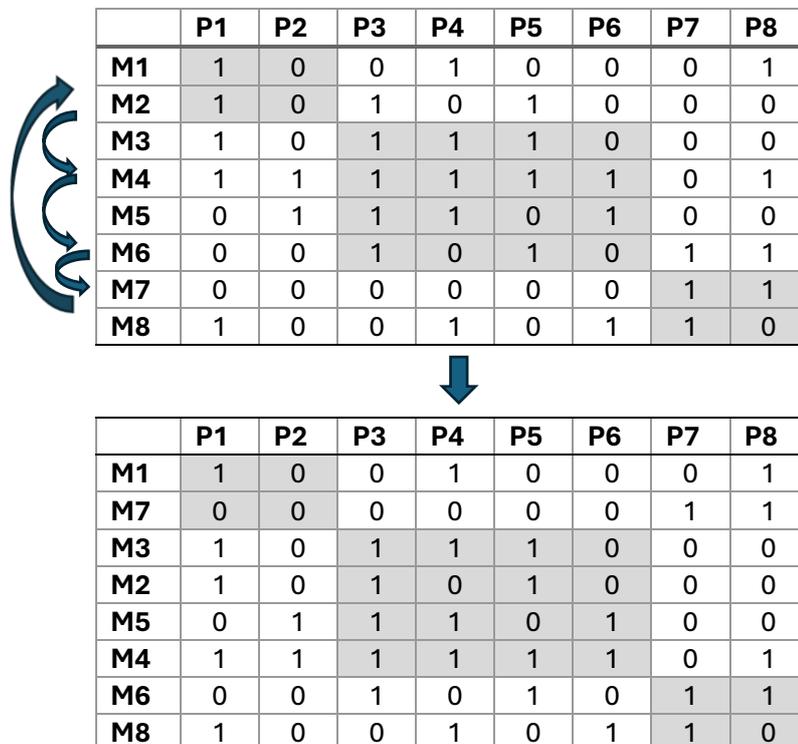


Figura 3.13 - Estruturas de vizinhança Cadeia de Ejeção de MÁQUINAS

CAPÍTULO IV

4. RESULTADOS E DISCUSSÕES

Para os experimentos computacionais realizados nesse trabalho, foram considerados as seguintes restrições:

- proibição de células residuais;
- permissão de singletons (cada cluster possui pelo menos uma máquina e uma parte);

Em relação ao tamanho dos clusters, a opção mais comumente utilizada na literatura tem sido a permissão de *singletons* e a proibição de células residuais.

4.1. AMBIENTE DE TESTES E INSTÂNCIAS

Para os experimentos computacionais realizados nesse trabalho, foram utilizados dois conjuntos de dados, *testsetA* e *testsetB*, disponíveis no site researchgate.net, detalhados a seguir:

Conjunto de testes A (*testsetA*) – Conjunto de dados clássico retirado de (GONÇALVES, J.F., RESENDE, M.G.C., 2004). Ele contém 35 instâncias de teste com tamanhos de 5x7 até 40x100 (máquinas x partes). Este conjunto de dados é muito popular entre os pesquisadores do problema de células de manufatura e existem muitos resultados computacionais obtidos por diferentes métodos (heurísticas e metaheurísticas em geral).

Conjunto de testes B (*testsetB*) - Conjunto de dados retirado de (BYCHKOV, I.S., BATSYN, M.V., 2018). Possui 32 instâncias de teste com tamanhos de 6 x 6 até 50 x 150 (máquinas x partes) sendo as mesmas submetidas a formulação clássica do problema de formação de células de manufatura.

As tabelas abaixo apresentam as instâncias de cada conjunto de dados

Tabela 4.1 - Descrição do Conjunto A de Instâncias teste da literatura

Instância	Fonte	máquinas	partes
A1	King and Nakornchai (1982) - Fig. 1a	5	7
A2	Waghodekar and Sahu (1984) - Problem 2	5	7
A3	Seifoddini (1989b)	5	18
A4	Kusiak and Cho (1992)	6	8
A5	Kusiak and Chow (1987)	7	11
A6	Boctor (1991) - Example 1	7	11
A7	Seifoddini and Wolfe (1986)	8	12
A8	Chandrasekaran and Rajagopalan (1986a)	8	20
A9	Chandrasekaran and Rajagopalan (1986b)	8	20
A10	Mosier and Taube (1985a)	10	10
A11	Chan and Milner (1982)	15	10
A12	Askin and Subramanian (1987)	14	24
A13	Stanfel (1985)	14	24
A14	McCormick et al. (1972)	16	24
A15	Srinivasan et al. (1990)	16	30
A16	King (1980)	16	43
A17	Carrie (1973)	18	24
A18	Mosier and Taube (1985b)	20	20
A19	Kumar et al. (1986)	23	20
A20	Carrie (1973)	20	35
A21	Boe and Cheng (1991)	20	35
A22	Chandrasekharan and Rajagopalan (1989) - Dataset 1	24	40
A23	Chandrasekharan and Rajagopalan (1989) - Dataset 2	24	40
A24	Chandrasekharan and Rajagopalan (1989) - Dataset 3	24	40
A25	Chandrasekharan and Rajagopalan (1989) - Dataset 5	24	40
A26	Chandrasekharan and Rajagopalan (1989) - Dataset 6	24	40
A27	Chandrasekharan and Rajagopalan (1989) - Dataset 7	24	40
A28	McCormick et al. (1972)	27	27
A29	Carrie (1973)	28	46
A30	Kumar and Vannelli (1987)	30	41
A31	Stanfel (1985) - Fig. 5	30	50
A32	Stanfel (1985) - Fig. 6	30	50
A33	King and Nakornchai (1982)	30	90
A34	McCormick et al. (1972)	37	53
A35	Chandrasekharan and Rajagopalan (1987)	40	100

Gostaríamos de destacar que muitos pesquisadores da área utilizam o conjunto de dados de 35 instâncias GT fornecido por (GONÇALVES, J.F., RESENDE, M.G.C., 2004). Esses exemplos foram retirados de diferentes artigos de pesquisa sobre formação de células de manufatura (referências às fontes originais são mostradas na Tabela 4.1). Algumas instâncias problemáticas neste conjunto de dados contêm erros e diferem daqueles apresentados nos artigos originais. Muitos pesquisadores, incluindo (ELBENANI, B., FERLAND, J.A., 2012) e (RIAN G.S. PINHEIRO, et al 2016) realizaram seus experimentos computacionais utilizando os dados de (GONÇALVES,

J.F., RESENDE, M.G.C., 2004). Foram revisadas todas as fontes originais, comparando e formando a versão corrigida desse conjunto de dados. Também foram coletadas muitas outras instâncias problemáticas menos populares que formaram esse novo conjunto de dados.

Tabela 4.2 - Descrição do Conjunto B de Instâncias teste da literatura

Instância	Fonte	máquinas	partes
B1	Adil et al. (1996)	6	6
B2	Pa Rkin and Li (1997)	6	7
B3	Brown and Sumichrast (2001)	6	11
B4	Chan and Milner (1982)	7	5
B5	Kusiak and Chow (1987)	7	8
B6	Zolfaghari and Liang (2002)	7	8
B7	Won and Kim (1997)	7	10
B8	Sarker and Khan (2001)	8	8
B9	Nair (1999)	8	10
B10	Islam and Sarker (2000)	8	10
B11	Kumar et al. (1986)	9	15
B12	Ham et al. (1985)	10	8
B13	Viswanathan (1996)	10	12
B14	Shargal et al. (1995)	10	38
B15	Won and Kim (1997)	11	10
B16	Seifoddini (1988)	11	22
B17	Moon and Chi (1992)	12	19
B18	Li (2003)	14	14
B19	Chan and Milner (1982) - Fig. 3a	15	10
B20	Yang and Yang (2008) - Fig. 6b	15	15
B21	Yang and Yang (2008) - Fig. 6c	15	15
B22	Yang and Yang (2008) - Fig. 6d	15	15
B23	Harhalakis et al. (1994)	17	20
B24	Seifoddini and Djassemi (1991)	18	24
B25	Sandbothe (1998)	20	10
B26	Nagi et al. (1990)	20	51
B27	Won and Kim (1997)	26	28
B28	Yang and Yang (2008) - Fig. 7	28	35
B29	Seifoddini and Djassemi (1996)	35	15
B30	Seifoddini and Djassemi (1996)	41	50
B31	Yang and Yang (2008) - Fig. 12	46	105

Os códigos utilizados nesta dissertação foram elaborados na linguagem de programação Python sendo executados utilizando o aplicativo Visual Studio Code 1.91. Como plataforma para nossos cálculos, Notebook Dell G15 com processador 13^a geração Intel® Core™ i5-13450HX (10-core, cache de 20MB, até 4.6GHz) com sistema operacional Windows 11 Home e placa de vídeo NVIDIA® GeForce® RTX™ 3050, 6GB GDDR6.

4.2. RESULTADOS FINAIS

Cada instância teste foi executada 10 vezes, sempre usando sementes randômicas distintas, e ao final foram registrados o melhor valor de função objetivo alcançado, a média e mediana dessas soluções. Também foram calculadas medidas de variabilidade, tais como: desvio padrão e coeficiente de variação (C.V.).

Tabela 4.3 – Resultado abordagem ILS-VND-RD conjunto A

Instância	Exato	ILS-VND-RD	Média	Desvio	Mediana	C.V.
A1	0.8235	0.8235	0.7456	0.0873	0.7555	11.71%
A2	0.6957	0.6957	0.6714	0.0429	0.6957	6.39%
A3	0.7959	0.7959	0.7937	0.0071	0.7959	0.89%
A4	0.7692	0.7692	0.7692	0.0000	0.7692	0.00%
A5	0.7083	0.7083	0.6764	0.0485	0.6942	7.17%
A6	0.6087	0.6087	0.5680	0.0271	0.5626	4.78%
A7	0.6944	0.6944	0.6431	0.0438	0.6458	6.81%
A8	0.8525	0.7969	0.7365	0.0436	0.7443	5.91%
A9	0.5872	0.5841	0.5508	0.0316	0.5492	5.74%
A10	0.7500	0.7500	0.7405	0.0300	0.7500	4.05%
A11	0.7206	0.7015	0.6505	0.0493	0.6594	7.58%
A12	0.7183	0.7183	0.6799	0.0408	0.6905	6.00%
A13	0.9200	0.8542	0.8028	0.0517	0.8184	6.44%
A14	0.5326	0.5155	0.4822	0.0287	0.4871	5.95%
A15	0.6899	0.6899	0.6432	0.0378	0.6514	5.88%
A16	0.5753	0.5563	0.5025	0.0365	0.5032	7.27%
A17	0.5773	0.5773	0.5456	0.0396	0.5541	7.26%
A18	0.4345	0.4255	0.4041	0.0144	0.4035	3.57%
A19	0.5798	0.5628	0.5390	0.0195	0.5422	3.62%
A20	0.7791	0.7405	0.7060	0.0353	0.7196	5.00%
A21	0.5081	0.4922	0.4709	0.0169	0.4722	3.59%
A22	1.0000	0.9466	0.7946	0.0964	0.7840	12.14%
A23	0.8511	0.8085	0.6990	0.0719	0.6803	10.29%
A24	0.7351	0.7162	0.6333	0.0640	0.6319	10.11%
A25	0.5329	0.4902	0.4691	0.0177	0.4718	3.77%
A26	0.4895	0.4694	0.4347	0.0210	0.4337	4.84%
A28	0.5482	0.5469	0.5425	0.0020	0.5418	0.37%
A30	0.6331	0.6096	0.5822	0.0236	0.5912	4.05%
A31	0.5977	0.5682	0.5267	0.0274	0.5212	5.20%
A33	0.4800	0.4067	0.3816	0.0188	0.3816	4.91%
A35	0.8403	0.5956	0.5648	0.0255	0.5670	4.52%

Na tabela 4.3, temos os resultados sumarizados para as 10 execuções para cada uma das instâncias do conjunto A. Na primeira coluna, temos o nome da instância, em seguida, o

valor exato da solução ótima, o melhor valor obtido pela nossa abordagem ILS-VND-RD, o valor médio, o desvio-padrão, a mediana e, por último, o coeficiente de variação. Podemos observar que nosso algoritmo encontra rapidamente o valor ótimo para as instâncias de menor dimensão, de A1 até A7. Para as demais instâncias, o valor ótimo só foi obtido somente em mais 4 instâncias, porém ficou muito próximo do valor ótimo das demais soluções.

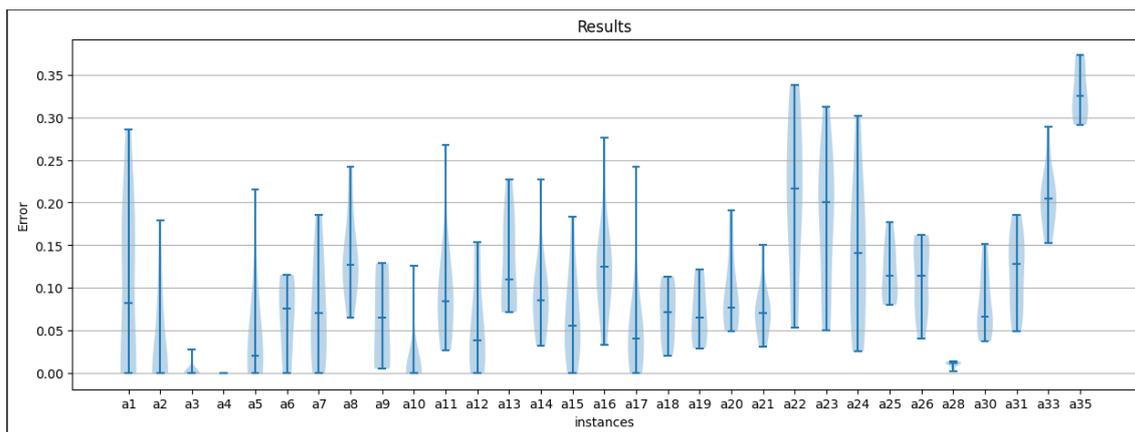


Figura 4.1 – Variação da Solução conjunto A

Na Figura 4.1 observamos que para as instâncias do conjunto A obtivemos a melhor solução em 11 das 35 instâncias. Quando consideramos um gap máximo de 5%, observamos que isto é alcançado em 25 das 35 instâncias. Quando analisamos a variabilidade da qualidade das soluções obtidas para cada instância em particular, considerando o conjunto das 10 execuções, fica claro que existe considerável variação no comportamento da abordagem proposta.

Tabela 4.4 – Resultado abordagem ILS-VND-RD conjunto B

Instância	Exato	ILS-VND-RD	Média	Desvio	Mediana	C.V.
B1	0.8095	0.8095	0.8095	0.0000	0.8095	0.00%
B2	0.7222	0.7222	0.7105	0.0290	0.7222	4.08%
B3	0.6071	0.6071	0.5756	0.0263	0.5769	4.56%
B4	0.8889	0.8889	0.8526	0.1147	0.8889	13.45%
B5	0.7500	0.7500	0.7381	0.0118	0.7333	1.59%
B6	0.7391	0.7391	0.6617	0.0630	0.6742	9.53%
B7	0.8148	0.7692	0.7241	0.0464	0.7172	6.41%
B8	0.7222	0.7222	0.6832	0.0451	0.6992	6.60%
B9	0.7576	0.7353	0.7133	0.0348	0.7353	4.88%
B10	0.9000	0.9000	0.8325	0.0334	0.8333	4.01%
B11	0.7273	0.7273	0.7114	0.0117	0.7103	1.64%
B12	0.8276	0.8148	0.7979	0.0181	0.8003	2.27%
B13	0.5962	0.5882	0.5750	0.0103	0.5757	1.78%
B14	0.6405	0.6129	0.5566	0.0439	0.5638	7.89%
B15	0.8333	0.8333	0.7894	0.0407	0.7931	5.16%
B16	0.7391	0.7262	0.6822	0.0450	0.7030	6.60%
B17	0.6552	0.6441	0.6079	0.0292	0.6132	4.80%
B18	0.6129	0.6111	0.5822	0.0245	0.5839	4.21%
B19	0.8000	0.7818	0.7118	0.0401	0.7103	5.63%
B20	0.8710	0.8065	0.7879	0.0285	0.8065	3.62%
B21	0.8333	0.8077	0.7582	0.0412	0.7524	5.44%
B22	0.7258	0.7000	0.6765	0.0265	0.6885	3.92%
B23	0.8111	0.7753	0.7525	0.0322	0.7714	4.28%
B24	0.5673	0.5566	0.5153	0.0300	0.5253	5.82%
B25	0.7600	0.7391	0.6846	0.0411	0.6889	6.00%
B26	0.6068	0.5948	0.5510	0.0287	0.5551	5.21%
B27	0.7248	0.7095	0.7007	0.0104	0.7027	1.49%
B28	0.6729	0.6588	0.6370	0.0189	0.6385	2.96%
B29	0.5730	0.5500	0.5273	0.0181	0.5294	3.43%
B30	0.7308	0.6962	0.6573	0.0289	0.6557	4.40%
B31	0.6799	0.6329	0.5723	0.0388	0.5694	6.77%

Na tabela acima, temos os resultados sumarizados para as 10 execuções para cada uma das instâncias do conjunto B. Foi repetida a mesma estrutura de colunas da Tabela de resultados para as instâncias do conjunto A. Podemos observar que nosso algoritmo encontra rapidamente o valor ótimo para as instâncias de menor dimensão, de B1 até B6, B8, B10, B11, B15. Para as demais instâncias, o valor obtido ficou muito próximo do valor ótimo das demais soluções, com gaps inferiores a 10%.

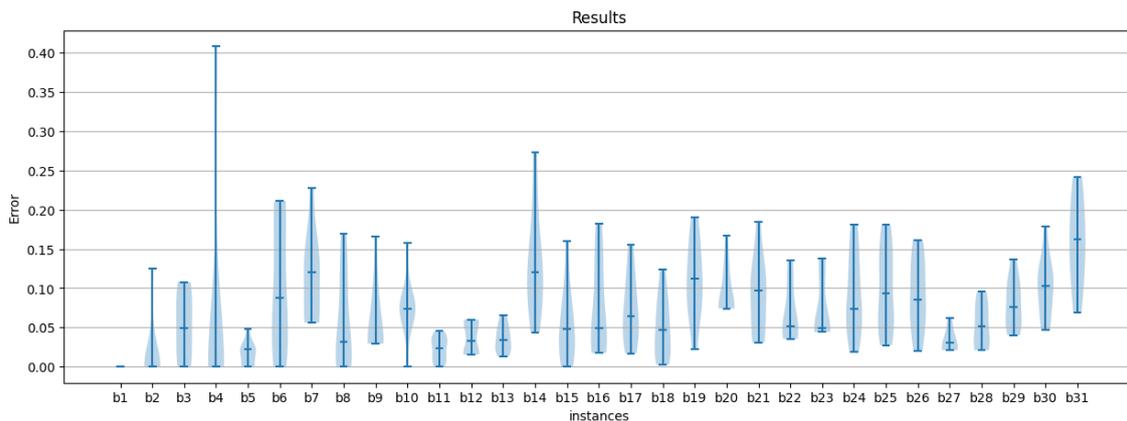


Figura 4.2 – Variação da Solução conjunto B

Na Figura 4.2 observamos que para as instâncias do conjunto B obtivemos a melhor solução em 11 das 31 instâncias. Quando consideramos um gap máximo de 5%, observamos que isto é alcançado em 28 das 31 instâncias. Quando analisamos a variabilidade da qualidade das soluções obtidas para cada instância em particular, considerando o conjunto das 10 execuções, verificamos que o comportamento da abordagem proposta observado para o conjunto A de instâncias também se repete para o conjunto B.

Quando analisamos o aspecto de robustez do algoritmo, notamos que há considerável variabilidade nos resultados obtidos durante as 10 execuções. O comportamento estocástico do algoritmo ILS-VND-RD explica parte dessa variabilidade, entretanto essa questão precisa ser mais bem endereçada. Isto é bastante perceptível ao analisarmos a amplitudes dos resultados apresentados para cada instância nas Figuras 4.1 e 4.2

4.3. COMPARAÇÃO COM O ESTADO DA ARTE

Os melhores resultados para esta versão do Problema de Formação Celular são atribuídos a (BYCHKOV, I.S., BATSYN, M.V., PARDALOS, P.M., 2014), que resolveram com sucesso muitas instâncias até a otimização usando um modelo exato conhecido como modelo de dois índices. Sua abordagem foi capaz de abordar quase todas as instâncias do conjunto de testes A e do conjunto de testes B.

Em contraste, nossa metodologia, ILS-VND-RD, concentrou-se em alcançar rapidamente soluções ideais para as instâncias menores nos conjuntos de testes A e B.

Para instâncias de médio e grande porte, nossa abordagem identificou com sucesso aproximações de alta qualidade soluções, com um desvio inferior a 5% do valor ótimo do objetivo função.

Ao analisar as instâncias do Conjunto B, a abordagem ILS-VND-RD de forma eficaz e encontrou rapidamente soluções ideais para as instâncias B1 a B6. Este desempenho foi consistente com os resultados observados para os casos do Conjunto A, onde soluções foram obtidas, normalmente com lacunas inferiores a 5%.

Tabela 4.5 – Comparativo Literatura X ILS-VND-RD conjunto A

Dataset	Elbenani and Ferland (2012)	Bychkov et al. (2014)	Two-index model	ILS
A1	0.8235	0.8235	0.8235	0.8235
A2	0.6957	0.6957	0.6957	0.6957
A3	0.7959	0.7959	0.7959	0.7959
A4	0.7692	0.7692	0.7692	0.7692
A5	0.6087	0.6087	0.6087	0.6087
A6	0.7083	0.7083	0.7083	0.7083
A7	0.6944	0.6944	0.6944	0.6944
A8	0.8525	0.8525	0.8525	0.7969
A9	0.5872	0.5872	0.5872	0.5841
A10	0.7500	0.7500	0.7500	0.7500
A11	0.9200	0.9200	0.9200	0.8542
A12	0.7183	0.7183	0.7183	0.7183
A13	0.7206	0.7206	0.7206	0.7015
A14	0.5326	*	0.5326	0.5155
A15	0.6953	*	0.6899	0.6899
A16	0.5753	*	0.5753	0.5563
A17	0.5773	*	0.5773	0.5773
A18	*	*	0.4345	0.4255
A19	0.5081	*	0.5081	0.4922
A20	0.7791	*	0.7791	0.7405
A21	0.5798	*	0.5798	0.5628
A22	1.0000	1.0000	1.0000	0.9466
A23	0.8511	*	0.8511	0.8085
A24	0.7351	*	0.7351	0.7162
A25	*	*	0.5329	0.4902
A26	*	*	0.4895	0.4694
A28	0.5482	*	0.5482	0.5469
A30	0.6331	*	0.6331	0.6096
A31	*	*	*	0.5682
A33	0.6064	*	0.6064	0.4067
A35	0.8403	*	0.8403	0.5956

Tabela 4.6 – Comparativo Literatura X ILS-VND-RD conjunto B

Dataset	Heuristic bound	Two-index (no residual cells)	ILS
B1	0.8095	0.8095	0.8095
B2	0.7222	0.7222	0.7222
B3	0.6071	0.6071	0.6071
B4	0.8889	0.8889	0.8889
B5	0.7500	0.7500	0.7500
B6	0.7391	0.7391	0.7391
B7	0.8148	0.8148	0.7692
B8	0.7222	0.7222	0.7222
B9	0.7576	0.7576	0.7353
B10	0.9000	0.9000	0.9000
B11	0.7273	0.7273	0.7273
B12	0.8276	0.8276	0.8148
B13	0.5962	0.5962	0.5882
B14	0.6405	0.6405	0.6129
B15	0.8333	0.8333	0.8333
B16	0.7391	0.7391	0.7262
B17	0.6552	0.6552	0.6441
B18	0.6027	0.6129	0.6111
B19	0.8000	0.8000	0.7818
B20	0.8710	0.8710	0.8065
B21	0.8333	0.8333	0.8077
B22	0.7258	0.7258	0.7000
B23	0.8111	0.8111	0.7753
B24	0.5673	0.5673	0.5566
B25	0.7600	0.7600	0.7391
B26	0.6068	0.6068	0.5948
B27	0.7248	0.7248	0.7095
B28	0.6729	0.6729	0.6588
B29	0.5730	0.5730	0.5500
B30	0.7308	0.7308	0.6962
B31	0.6799	0.6799	0.6329

CAPÍTULO V

5. CONCLUSÕES

Este trabalho apresentou uma abordagem aproximada capaz de gerar excelentes soluções para o problema de formação de células de manufatura, utilizando a metaheurística *Iterated Local Search* (ILS) combinada com a metaheurística *Variable Neighborhood Descent* (VND) para explorar diversas estruturas de vizinhança e melhorar a diversificação de soluções através perturbações estratégicas. Nosso algoritmo ILS-VND-RD apresentou excelente desempenho computacional, alcançando as soluções ótimas de 22 instâncias presentes em um benchmark com 62 problemas testes utilizados na literatura da área.

Ao comparar nossa metodologia, ILS-VND-RD, com abordagens existentes, como o modelo de dois índices de (BYCHKOV, I.S., BATSYN, M.V., PARDALOS, P.M., 2014), destacamos a eficiência e a eficácia de nossa estratégia tanto para instâncias menores quanto para grandes instâncias. Especificamente, ILS-VND-RD alcança soluções ótimas para as primeiras seis instâncias no Conjunto B e alcançou soluções aproximadas de alta qualidade para instâncias de dimensões médias e grandes, mantendo um gap inferior a 5% do valor ótimo da respectiva função objetivo.

No geral, a abordagem proposta contribui significativamente para a área de pesquisa em algoritmos para o problema de formação de células de manufatura, fornecendo uma estrutura metodológica que equilibra alta qualidade de soluções obtidas e baixo custo computacional.

Trabalhos futuros poderiam explorar mais refinamentos na abordagem ILS-VND-RD em relação à sua parametrização de execução, bem como incorporar novas estruturas de vizinhança. Outra importante ação futura é a combinação com o uso do modelo matemático de dois índices para explorar uma vizinhança de forma exata e assim criar uma abordagem de mathheurística.

REFERÊNCIAS

- ADIL, G.K., RAJAMANI, D., STRONG, D., 1996. *Cell formation considering alternate route- ings*. Int. J. Prod. Res. 34 (5), 1361–1380.
- ANDRÉS C, LOZANO S., 2006 *A particle swarm optimization algorithm for part-machine grouping*. Robot Comput Integr Manuf 22(5):468–474.
- ASKIN, R.G., SUBRAMANIAN, S., 1987. *A cost-based heuristic for group technologycon- figuration*. Int. J. Prod. Res. 25 (1), 101–113.
- BOCTOR, F., 1991. *A jinear formulation of the machine-part cell formation problem*. Int. J. Prod. Res. 29 (2), 343–356.
- BALLAKUR A, STEUDEL HJ., 1987. *A within-cell utilization based heuristic for designing cellular manufacturing systems*. Int J Prod Res 25(5):639–665
- BOUTSINAS, B., 2013. *Machine-part cell formation using biclustering*. Eur. J. Oper. Res. 230, 563–572.
- BROWN, E.C., SUMICHRAS, R.T., 2001. *CF-GGA: a grouping genetic algorithm for the cell formation problem*. Int. J. Prod. Res. 39 (16), 3651–3669.
- BRUSCO, M.J., 2015. *An exact algorithm for maximizing grouping efficacy in part-machine clustering*. IIE Trans. 47 (6), 653–671.
- BURKE L., KAMAL S., .1995. *Neural networks and the part family/ machine group formation problem in cellular manufacturing: a framework using fuzzy ART*. J Manuf Syst 14(3):148–159.
- BYCHKOV, I., BATSYN, M., SUKHOV, P., PARDALOS, P.M., 2013. *Heuristic algorithm for the cell formation problem*. In: *Models, Algorithms, and*

Technologies for Network Analysis. In: Springer Proceedings in Mathematics & Statistics, 59, pp. 43–69.

BYCHKOV, I.S., BATSYN, M.V., PARDALOS, P.M., 2014. *Exact model for the cell formation problem*. *Optim. Lett.* 8 (8), 2203–2210.

BYCHKOV, I.S., BATSYN, M.V., 2018 *An efficient exact model for the cell formation problem with a variable number of production cells* .*Computers & Operations Research*, Volume 91, Pages 112-120, ISSN 0305-0548, <https://doi.org/10.1016/j.cor.2017.11.009>.

ELBENANI, B., FERLAND, J.A., 2012. *Cell Formation Problem Solved Exactly with the Dinkelbach Algorithm*. CIRRELT-2012-07. Montreal. Quebec., pp. 1–14.

GONÇALVES, J.F., RESENDE, M.G.C., 2004. *An evolutionary algorithm for manufacturing cell formation*. *Comput. Ind. Eng.* 47 (2–3), 247–273.

BURBIDGE. J.L., 1971 *Production flow analysis*. *Production Engineer*, 50(4-5):139-152.

KRAMER R., SUBRAMANIAN A., VIDAL T., CABRAL L. 2015. *A matheuristic approach for the pollution-routing problem* *European Journal of Operational Research*, 243 pp. 523-539.

KUMAR, K. R., & CHANDRASEKHARAN, M. P. 1990. *Grouping efficacy: A quantitative criterion for goodness of block diagonal forms of binary matrices in group technology*. *International Journal of Production Research*, 28(2), 233–243.

KUMAR, K.R., KUSIAK, A., VANNELLI, A., 1986. *Grouping of parts and components in flexible manufacturing systems*. *Eur. J. Oper. Res.* 24, 387–397.

LEI D, WU Z 2005 *Tabu search approach based on a similarity coefficient for cell formation in generalized group technology*. *Int J Prod Res* 43(19):4035–4047.

- LI X, BAKI M, ANEJA Y. P., 2010 *An ant colony optimization metaheuristic for machine-part cell formation problems*. *Comput Oper Res* 37(12):2071–2081.
- LOGENDRAN, R., & PUVANUNT, V. 1997. *Duplication of machines and subcontracting of parts in the presence of alternative cell locations*. *Computers & industrial engineering*, 33(1-2), 235-238.
- LOZANO S, CANCA D, GUERRERO F, GARCIA J., 2001 *Machine grouping using sequence-based similarity coefficients and neural networks*. *Robot Comput Integr Manuf* 17(5):399–404.
- MAHDAVI, I., MAHDI, M., SOLIMANPUR, M., AND HEIDARZADE, A. 2009. *Expert Systems with Applications Genetic algorithm approach for solving a cell formation problem in cellular manufacturing*. *Expert Systems With Applications*, 36(3):6598–6604.
- MAHMOODIAN, V., JABBARZADEH, A., REZAZADEH, H., & BARZINPOUR, F., 2017. *A novel intelligent particle swarm optimization algorithm for solving cell formation problem*. *Neural Computing and Applications*, 31, 801 - 815.
- MIRZAPOUR AL-E-HASHEM SMJ, ARYANEZHAD MB, JABBARZADEH A 2011 *A new approach to solve a mixed-model assembly line with a bypass subline sequencing problem*. *Int J Adv Manuf Technol* 52(9–12):1053–1066
- MUKHERJEE, I. AND RAY, P. K. 2006. *A review of optimization techniques in metal cutting processes*. *Computers and Industrial Engineering*, 50(1-2):15–34.
- GAREY, M. R., JOHNSON D. S., 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., USA.
- N. MLADENOVIĆ AND P. HANSEN. 1997 *Variable Neighborhood Search*. *Computers and Operations Research*, 24:1097–1100.

- NOKTEHDAN A, KARIMI B, HUSSEINZADEH KASHAN A., 2010. *A differential evolution algorithm for the manufacturing cell formation problem using group based operators*. Expert Syst Appl 37(7): 4822–4829.
- PAPAIOANNOU G., WILSON J. M., 2010 *The evolution of cell formation problem methodologies based on recent studies (1997–2008): review and directions for future research*. Eur J Oper Res 206(3):509–521.
- RIAN G.S. PINHEIRO, IVAN C. MARTINS, FÁBIO PROTTI, LUIZ S. OCHI, LUIDI G. SIMONETTI, ANAND SUBRAMANIAN, 2016, *On solving manufacturing cell formation via Bicluster Editing*, European Journal of Operational Research,
- SARKER, B.R., 2001. *Measures of grouping efficiency in cellular manufacturing systems*. Eur. J. Oper. Res. 130, 588–611.
- SARKER, B.R., KHAN, M., 2001. *A comparison of existing grouping efficiency measures and a new weighted grouping efficiency measure*. IIE Trans. 33, 11–27.
- SEIFODDINI, H., DJASSEMI, M., 1996. *The threshold value of a quality index for formation of cellular manufacturing systems*. Int. J. Prod. Res. 34 (12), 3401–3416.
- SHARGAL, M., SHEKHAR, S., IRANI, S.A., 1995. *Evaluation of search algorithms and clustering efficiency measures for machine-part matrix clustering*. IIE Trans. 27 (1), 43–59.
- SOLEYMANPOUR M, VRAT P, SHANKAR R 2002 *A transiently chaotic neural network approach to the design of cellular manufacturing*. Int J Prod Res 40(10):2225–2244
- SOUZA, M. J. F., 2024 *Inteligência Computacional para Otimização: meta-heurísticas*. Departamento de Computação, Universidade Federal de Ouro Preto, Ouro Preto, Minas Gerais, Disponível em

[http://www.decom.ufop.br/prof/marcone/Disciplinas/InteligenciaComputacional/Inteligencia Computacional.pdf](http://www.decom.ufop.br/prof/marcone/Disciplinas/InteligenciaComputacional/Inteligencia%20Computacional.pdf).

SUCUPIRA, IGOR RIBEIRO. 2004, *Métodos Heurísticos Genéricos, Metaheurísticas e Hiperheurísticas*-. Dissertação – Departamento de Ciência da Computação, Universidade de São Paulo, São Paulo.

SULE, S. AND SURESH, N. C. 2009. *A goal-programming approach for design of hybrid cellular manufacturing systems in dual resource constrained environments*. Computers & Industrial Engineering, 56(2):560–575.

WEMMERLÖV U, HYER NL 1989, *Cellular manufacturing in the US industry: a survey of users*. Int J Prod Res 27(9):1511–1530

DINKELBACH, W., 1967 *On Nonlinear Fractional Programming*. Management Science 13(7):492-498

XUE, G. AND OFFODILE, O. F. 2020. *Integrated optimization of dynamic cell formation and hierarchical production planning problems*. Computers and Industrial Engineering, 139(October 2019):106155.