



**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS CURITIBA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA E DE
MATERIAIS – PPGEM
DEPARTAMENTO DE PESQUISA E PÓS-GRADUAÇÃO**

ADRIANO PEREIRA BALAU

**USO DE MÉTODOS HEURÍSTICOS E *BRANCH-AND-BOUND* PARA
OTIMIZAÇÃO DO *LAYOUT* FABRIL DA LINHA DE MONTAGEM DE UM
COMPONENTE AUTOMOTIVO NA REGIÃO DE CURITIBA**

DISSERTAÇÃO

**CURITIBA
2013**

ADRIANO PEREIRA BALAU

**USO DE MÉTODOS HEURÍSTICOS E *BRANCH-AND-BOUND* PARA
OTIMIZAÇÃO DO *LAYOUT* FABRIL DA LINHA DE MONTAGEM DE UM
COMPONENTE AUTOMOTIVO NA REGIÃO DE CURITIBA**

Dissertação apresentado ao Programa de Pós-Graduação em Engenharia Mecânica e de Materiais da Universidade Tecnológica Federal do Paraná como requisito parcial à obtenção do título de “Mestre em Engenharia”

Orientador: Prof. Dr. Luiz Carlos de Abreu Rodrigues.

**CURITIBA
2013**

DEDICATÓRIA

À minha esposa Márcia, que sempre demonstrou paciência, carinho, dedicação e confiança, nunca me deixando perder a energia ao longo do caminho de busca pelo conhecimento. Aos meus filhos Bruno e Caio, que mesmo com tão pouca idade, pareceram entender a importância da pesquisa para a melhoria na vida das pessoas, mesmo que isso significasse sacrificar parte do tempo ao seu lado. Aos meus pais José e Maria Eunice, que me ensinaram valores e me incentivaram ao longo de toda minha vida na busca do conhecimento.

AGRADECIMENTOS

Agradeço primeiro a Deus por sempre me dar forças nos percalços enfrentados nos caminhos tortuosos da vida, sem nunca esmorecer.

Agradeço também à minha esposa Márcia e filhos Bruno e Caio, por sempre me apoiarem na elaboração deste trabalho, mesmo que isso significasse menos tempo ao seu lado.

Agradeço também aos meus pais José e Maria Eunice, que sempre me incentivaram na busca incessante pelo conhecimento, sem pedir nada em troca.

Gostaria de agradecer, da mesma forma, ao Prof. Dr. Luiz Carlos Rodrigues, que sempre demonstrou bom humor, respeito, paciência e me inspirou na busca por algo melhor e inovador.

Agradeço também aos professores e funcionários da UTFPR e em específico do PPGEM, que sempre demonstraram determinação e paciência na interminável busca por algo melhor.

BALAU, Adriano P. Uso de métodos heurísticos e *branch-and-bound* para otimização do *layout* fabril da linha de montagem de um componente automotivo na região de Curitiba. 114f. Dissertação (Mestrado em Engenharia) - Programa de Pós-graduação em Engenharia Mecânica e de Materiais, Universidade Tecnológica Federal do Paraná.

RESUMO

As empresas de manufatura, nos dias atuais, estão incessantemente em busca de redução de custos, motivadas pela concorrência e competição, que são características fortes da globalização. No Sistema Toyota de Produção (OHNO, 1988) é ressaltada a questão dos sete desperdícios que podem existir em um processo e que, conseqüentemente, geram custos no produto sem, contudo agregar valor ao mesmo. Um dos desperdícios mais comumente encontrados são os do fluxo do produto semiacabado (*WIP*), matéria-prima ou produto acabado. O estudo de *Layout* visa otimizar a disposição dos recursos dentro de um processo de modo a minimizar, entre outros, o fluxo de materiais. O presente estudo visa apresentar um caso real de uma grande empresa de autopeças na região de Curitiba, PR, que gasta milhões por ano em mudanças de *Layout*. O objeto de estudo é a linha de montagem de um determinado componente que esta empresa fabrica. Através do uso de Métodos Heurísticos propõe-se uma abordagem para a otimização do *Layout* desta linha de montagem. Esta abordagem foi dividida em duas etapas. Na primeira etapa, foi resolvido o problema de formação de células (visando melhorar os tempos computacionais, bem como a qualidade da solução), visando associar as máquinas disponíveis às peças a serem fabricadas. Na segunda etapa, resolve-se o problema de otimização do *layout*, considerando as associações de máquinas às peças feitas na primeira etapa. Nas duas etapas testou-se o uso de uma abordagem meta-heurística (busca tabu) híbrida, bem como o método exato denominado *Branch-and-Bound* (este na primeira etapa), para resolver o problema. Os resultados encontrados no arranjo físico das máquinas mostraram-se bastante promissores.

Palavras-chave: *Layout*, busca Tabu, otimização, métodos heurísticos, *branch-and-bound*.

BALAU, Adriano P. Use of heuristic and *branch-and-bound* methods for the *layout* optimization at an assembly line of automotive parts in a multinational industry located at the region of Curitiba. 114f. Dissertação (Mestrado em Engenharia) - Programa de Pós-graduação em Engenharia Mecânica e de Materiais, Universidade Tecnológica Federal do Paraná.

ABSTRACT

Nowadays, the manufacturing enterprises are constantly looking for costs reduction, driven by rivalry and competition, which are strong globalization characteristics. In the Toyota Production System (OHNO, 1988), are highlighted the seven wastes which can exist in a manufacturing process and that, consequently, generate costs to the product without, however, adding value to it. Some commonly found wastes are the work-in-process (WIP), raw material or finished products flow wastes. The layout study aims to optimize the layout of facilities inside a process to minimize, among others, the materials flow. This study aims to present a real case of a huge auto parts manufacturer enterprise located in Curitiba, PR, which spends millions a year on layout changes. The object of study is the assembly line of a specific component that this company manufactures. Using Heuristic methods, it proposes an approach for the layout optimizing of this assembly line. This approach was divided in two stages: in the first one, the cell formation problem (in order to improve the computational time, as well as the solution quality) was solved in order to associate machines to parts. In the second stage, the layout optimizing problem is solved, considering the combination of machines to parts (made in first stage). In both stages the hybrid meta-heuristics approach (tabu search), as well as the Exact method so called *Branch-and-Bound* (this on first stage), were tested to solve this problem. The results found on layout of facilities were quite promising.

Keywords: facilities layout, tabu search, optimizing, heuristics methods, *branch-and-bound*.

LISTA DE FIGURAS

Figura 1 - Problemas típicos em uma manufatura	15
Figura 2 - Dois tipos diferentes de layout celular: (1) linear e (2) com duas colunas (traduzido de Lei & Wu, 2005).	28
Figura 3 - Classificação de métodos para solução de problemas de formação de células (CF) (traduzido de Papaioannou e Wilson, 2010).	31
Figura 4 - Lista de fontes referentes a estudos de formação de células (traduzido de Papaioannou e Wilson, 2010).	33
Figura 5 - Comparação dos métodos de formação de células (traduzido de Papaioannou e Wilson, 2010).	34
Figura 6 - Estrutura principal de um sistema de manufatura celular (CMS) (traduzido de WANG et al, 1998).	36
Figura 7 - Exemplo de layout de CMS (traduzido de WANG et al., 1998).	37
Figura 8 - Algoritmo SA Híbrido traduzido de Caux et al (2000).	44
Figura 9 - Algoritmo de Preparação do BB.	47
Figura 10 - Procedimento de busca do BB (Algoritmo 5).	49
Figura 11 - Algoritmo de busca tabu implementado.	50,59
Figura 12 - Algoritmo de busca na vizinhança conforme algoritmo de busca tabu implementado – esta figura é chamada de algoritmo 2.	51
Figura 13 - Algoritmo de avaliação da solução, conforme busca na vizinhança para algoritmo de busca tabu implementado – esta figura é chamada de algoritmo 3.	52
Figura 14 - Exemplo de análise de sobreposição.	57
Figura 15 - Algoritmo de busca na vizinhança. (Algoritmo 2)	59
Figura 16. Produção parcial do produto no período considerado. Fonte: empresa estudada.	64
Figura 17. Gráfico de pareto do total dos volumes fornecidos. Fonte: empresa estudada.	65

Figura 18. Operações executadas na linha de montagem do produto. Fonte: empresa estudada.	66
Figura 19. Percentual de ocupação (em %) de uma máquina por operação. Fonte: empresa estudada.	67
Figura 20. Dimensões e quantidades de máquinas no setor (em U.M.). Fonte: empresa estudada.	68
Figura 21. Código de cada operação, quantidade disponível de máquinas, tempos de operação e setup, bem como tamanho de lote para cada operação – Fonte: empresa estudada.	69
Figura 22. Tela do sistema de formação de células no Visual Basic 2008. Fonte: Autor.	71
Figura 23. Solução inicial do caso 1.	73
Figura 24. Solução ótima do caso 1.	73
Figura 25. Solução inicial do caso 2.	73
Figura 26. Solução ótima do caso 2.	73
Figura 27 – Convergência da solução para o caso 1 – configuração de 4 linhas.	78
Figura 28 – Convergência da solução para o caso 2 – configuração de 4 linhas.	78
Figura 29 - Convergência da solução para o caso 1 – configuração de 5 linhas.	78
Figura 30. Alocação encontrada dos postos de trabalho.	81
Figura 31. Coordenada da mediana dos postos na solução encontrada.	81
Figura 32. Resumo do relatório gerado pelo GAMS.	82
Figura 33. Alocação encontrada dos postos de trabalho.	83
Figura 34. Coordenada da mediana dos postos na solução encontrada.	83
Figura 35. Resumo do relatório gerado pelo GAMS.	84
Figura 36. Alocação encontrada dos postos de trabalho.	86
Figura 37. Coordenada da mediana dos postos na solução encontrada.	86
Figura 38. Resumo do relatório gerado pelo GAMS.	87
Figura 39. Alocação encontrada dos postos de trabalho.	88
Figura 40. Coordenada da mediana dos postos na solução encontrada.	89
Figura 41. Resumo do relatório gerado pelo GAMS.	89
Figura 42. Alocação encontrada dos postos de trabalho.	90
Figura 43. Coordenada da mediana dos postos na solução encontrada.	91
Figura 44. Resumo do relatório gerado pelo GAMS.	92

Figura 45. Alocação encontrada dos postos de trabalho.	93
Figura 46. Coordenada da mediana dos postos na solução encontrada.	93
Figura 47. Resumo do relatório gerado pelo GAMS.	94
Figura 48 Alocação encontrada dos postos de trabalho.	96
Figura 49. Coordenada da mediana dos postos na solução encontrada.	96
Figura 50. Alocação encontrada dos postos de trabalho.	97
Figura 51. Coordenada da mediana dos postos na solução encontrada.	98
Figura 52. Alocação encontrada dos postos de trabalho.	99
Figura 53. Coordenada da mediana dos postos na solução encontrada.	99
Figura 54. Alocação encontrada dos postos de trabalho.	100
Figura 55. Coordenadas das medianas dos postos na solução encontrada.	101
Figura 56. Alocação encontrada dos postos de trabalho.	102
Figura 57. Coordenadas das medianas dos postos na solução encontrada.	102
Figura 58. Alocação encontrada dos postos de trabalho.	103
Figura 59. Coordenada da mediana dos postos na solução encontrada.	104
Figura 60. Resumo das soluções obtidas pela abordagem PLIM.	104
Figura 61. Resumo das soluções obtidas pela abordagem BT.	105
Figura 62. Layout resultante, para o caso 1, 4 linhas, abordagem PLIM.	105

LISTA DE TABELAS

Tabela 1- Resultados computacionais para Modelos Difusos (Fuzzy) e Determinísticos (traduzido de PAPAIOANNOU e WILSON, 2008).	39
Tabela 2 - Características do arranjo físico, conforme o produto e processo produtivo (traduzido de BLACK, 1990).	40
Tabela 3 - Parâmetros usados na modelagem.	53
Tabela 4 - Variáveis usadas na modelagem.	53
Tabela 5 - Demanda caso 1.	77
Tabela 6 - Demanda caso 2.	77
Tabela 7 - Evolução da média da melhor solução encontrada.	77
Tabela 8- Demanda caso 3.	79

LISTA DE ABREVIATURAS E SIGLAS

- ACO** Método de Otimização de Colônia de Formigas (*Ant Colony Optimization*)
- BB** *Branch-and-Bound* ou Busca em Árvore
- BT** Busca Tabu (*Tabu Search*)
- AG** Algoritmo Genético (*Genetic Algorithm*)
- PFC** Problema de Formação de Células de Manufatura
- PSO** Método de Otimização de Enxame de Partículas (*Particle Swarm Optimization*)
- SA** *Simulated Annealing*
- SMC** Sistema de Manufatura Celular
- TG** Tecnologia de Grupo (*Group Technology*)
- WIP** *Work-In-Process* ou peças em processamento (não acabadas)

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Caracterização do Problema	15
1.1.1	Introdução ao Problema	16
1.2	Objetivo	18
1.2.1	Objetivos Específicos	18
1.3	Justificativa	18
1.4	Estrutura da Dissertação	19
2	REVISÃO BIBLIOGRÁFICA	20
2.1	Métodos de Otimização	20
2.1.1	Busca Tabu (BT)	21
2.1.2	<i>Simulated Annealing (SA)</i>	22
2.1.3	Algoritmo Genético (AG)	25
2.2	Tecnologia de Grupo ou Formação de células	27
2.3	<i>Layout</i>	38
3	METODOLOGIA	43
4	RESULTADOS	63
4.1	Descrição dos dados do problema	64
4.1.1	Características do problema analisado	69
4.2	Implementação da Solução do Problema de Formação de Células	71
4.3	Resultados da Solução do Problema de Formação de Células	72
4.4	Resolução do Problema de Layout	79
4.4.1	Soluções Obtidas pela Abordagem PLIM	79
4.4.2	Soluções Obtidas pela Abordagem BT Híbrida	95
5	CONCLUSÕES E TRABALHOS FUTUROS	106
6	REVISÃO BIBLIOGRÁFICA	108

1 INTRODUÇÃO

Diversas empresas do setor automotivo têm empregado grandes quantias de dinheiro e recursos na busca incessante do “*layout ideal*”. Um agravante no meio industrial reside no fato que as máquinas podem ter diversos agregados hidráulicos, pneumáticos e elétricos e, conseqüentemente, uma grande infra-estrutura necessária para manter estas partes em perfeito funcionamento. De acordo com Wang e Wu (2001), um *layout* ruim resulta em produtividade baixa, alta quantidade de WIP e fluxo confuso de materiais – vide Figura 1, demonstrando alguns destes problemas. Conforme Slack *et al.* (1999), o arranjo físico ou *layout* é tão relevante que chega a auxiliar no desempenho de uma empresa e, também, na satisfação do cliente. Já de acordo com Aiello *et al.* (2013), o estudo do *layout* das instalações visa o mais eficiente arranjo físico para as instalações que possuem um certa interação, até podendo encontrar mais objetivos. As constantes mudanças de disposição física de recursos e materiais visam sempre, utilizando-se a linguagem proposta pelo “Sistema Toyota de Produção”, a redução dos sete desperdícios (OHNO, 1988) – a Figura 1 demonstra, de forma direta ou indireta, alguns dos problemas típicos na manufatura, que têm relação com estes desperdícios (e que podem até ter uma relação de causa-consequência ou vice-versa, com os mencionados desperdícios). O presente estudo visa lançar mão de ferramentas já conhecidas e com vasta literatura para problemas de otimização, os denominados métodos heurísticos (ex. busca tabu), para o problema de *layout* fabril (arranjo físico das máquinas). A empresa estudada é uma empresa multinacional de autopeças, com presença em diversos países do mundo. Será estudada a linha de montagem de determinado componente feito na planta de manufatura desta empresa.

Na primeira parte do problema, são formadas duas células de manufatura, respeitando restrições de capacidade fabril. Em um primeiro instante, é utilizada uma abordagem denominada BT Híbrida, para alocar as máquinas às células e, a partir daí, com o uso da abordagem de *Branch-and-Bound* (BB), alocar as peças às máquinas (conforme Rodrigues *et al.* 2013).

Após a formação das duas células de manufatura, o Método de BT Híbrida é utilizado (conforme Weller, 2008) para compor o *Layout* da linha de produção considerada no presente estudo. O espaço total disponível para alocar as máquinas

será dividido em duas partes, sendo que cada parte será atribuída a uma célula de manufatura. A função objetivo é a minimização dos custos com transporte entre operações na linha de produção.

Em suma, este trabalho visa resolver o problema de formação de *layout* fabril. É importante mencionar que a grande contribuição deste trabalho, reside justamente no fato de se tratar de um problema real. São utilizados neste problema os métodos meta-heurísticos encontrados na literatura.

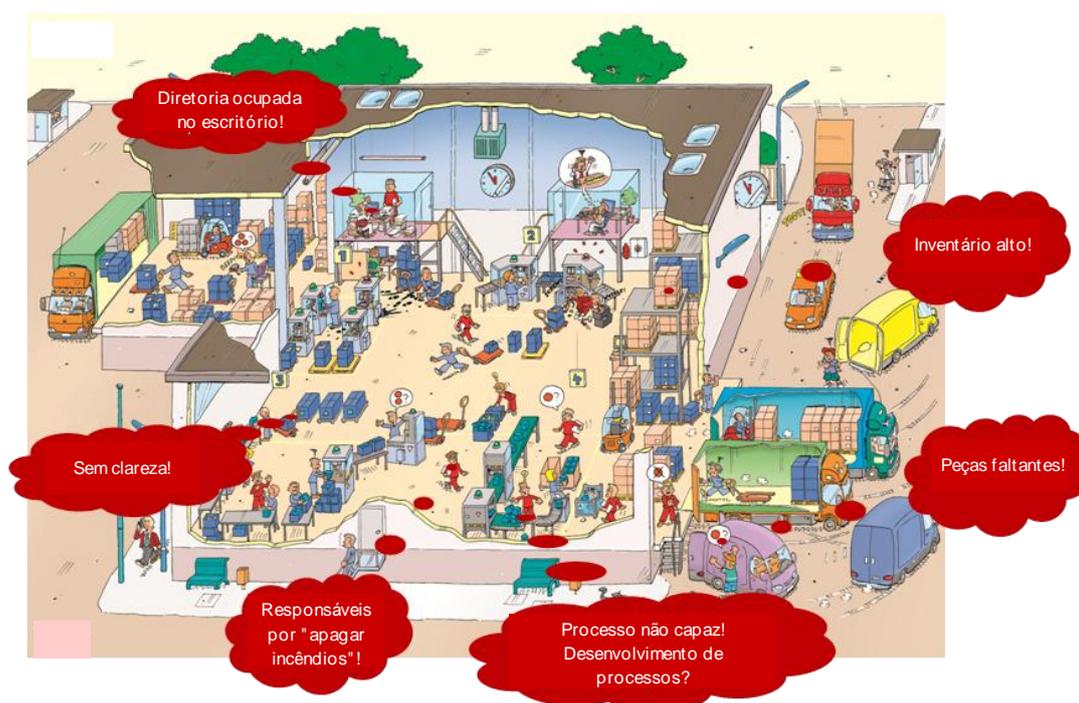


Figura 1 - Problemas típicos em uma manufatura (Fonte: empresa estudada)

1.1 Caracterização do Problema

Conforme Wang e Wu (2001) e Wang *et al.* (1998), nos últimos anos, as empresas têm enfrentado alguns agravantes no seu planejamento em um âmbito geral, como demandas variáveis por parte dos seus clientes variável, um ciclo de vida curto dos seus produtos, competição globalizada e busca por produtos melhores em um espaço curto de tempo, não obstante demandando uma enorme flexibilidade por parte do fornecedor. O sistema de manufatura celular (SMC), através de flexibilidade e eficiência no fluxo de materiais (conseqüentemente, reduz

custos com transporte de materiais), proporciona um menor tempo de *setup*, reduz a quantidade de peças/lotes em processo (*WIP*) e *lead time* de produção. Além disso, a demanda de um produto varia ao longo de seu ciclo de vida, o que dificulta em muito a formação de células de manufatura (WANG e WU, 2001).

Nas últimas três décadas (PAPAIOANNOU e WILSON, 2010), pesquisadores dispenderam muito esforço em uma quantidade grande de estudos, visando à formação de células de manufatura (PFC). Os estudos estariam divididos, ainda, segundo esses autores, em três tipos:

- Métodos informais;
- Métodos de análise de codificação de peças;
- Métodos baseados em produção.

Nos métodos baseados em produção, pode-se fazer uma subdivisão em:

- Análise de Agrupamento;
- Aproximações por particionamento gráfico;
- Métodos Exatos;
- Métodos Heurísticos;
- Métodos de Inteligência Artificial.

Neste trabalho, o foco principal está no uso de métodos heurísticos para otimizar o *layout* na linha de montagem de um determinado componente automotivo. É importante considerar a formação de células de manufatura, pelo fato de haver similaridade entre peças e, também, entre máquinas.

1.1.1 Introdução ao Problema

A empresa em questão é do ramo metal-mecânico, com presença em inúmeros países no mundo, faturamento na ordem de 40 bilhões de euros em 2011 e que tem cerca de 300 mil funcionários em todo o mundo. A unidade estudada é responsável pela manufatura de autopeças e está situada na cidade de Curitiba, PR, sendo que é um dos maiores empregadores deste estado, com algo em torno de 3000 funcionários diretos no ano de 2011. O objeto deste estudo é uma das maiores e mais antigas áreas desta unidade. Será estudado o arranjo físico das máquinas na linha de montagem de um dos produtos mais modernos, na empresa, produzidos em Curitiba. A produção deste item gira em torno de 10000 peças por dia. Ao todo, este

produto possui 83 diferentes modelos, e o *mix* mensal de modelos fabricados é de aproximadamente 20.

Como revisado anteriormente, o problema de *layout* possui ampla literatura, mas a aplicação desta literatura a problemas reais não é sempre imediata, já que, normalmente, a literatura assume que há uma alocação das peças às máquinas. Ou seja, a literatura assume que o fluxo das peças nas máquinas já foi definido previamente antes da solução do problema de *layout*. Mas, na empresa considerada, o *layout* da fábrica se caracteriza hoje por um arranjo *jobshop*, onde as máquinas são agrupadas por função (ou tipo de máquina). Cabe informar que o local estudado já passou por alterações e avaliações anteriores de *layout*, mas que não surtiram o efeito desejado. Isso se deveu à falta de um procedimento sistemático que permitisse à empresa definir (e redefinir periodicamente) a alocação de peças (*mix* de produção) às máquinas, considerando que o *mix* de produção varia regularmente. Ou seja, já que este *mix* varia com frequência, as pessoas envolvidas no estudo da alteração do *layout* se viam incapazes de “antever” os futuros *mix* e adotavam uma solução conservadora (um arranjo parecido a um *jobshop*), que não lhes agradava, mas que refletia a sua falta de “ferramentas” para solução do problema.

Como o processo em questão possui uma grande variação com relação à quantidade de peças (83 no total), variações de versões de máquinas dentro da mesma operação (restrições técnicas), serão focados, para efeito de simplificação, os produtos com maior volume de produção (após a aplicação do conceito do Diagrama de Pareto ou Classificação ABC) apenas na Linha de um determinado componente, na linha de montagem. Foi tomada como base, para o volume de produção, a quantidade de itens fornecidos no ano 2011 e, a partir daí, tirada uma média mensal de cada produto, conforme sistemática de classificação ABC.

Como já mencionado, este processo tem problemas de restrições técnicas e, conseqüentemente, fluxos diversos de processamento, principalmente devido a restrições técnicas e processuais (esta última devido a uma sequência obrigatória dentro do processo) e de *mix* de produção (são 83 tipos diferentes na linha de um produto e um volume de produção de aproximadamente 10 mil peças por dia).

Dentro da tríade Qualidade-Custo-Prazo de entrega, é inevitável mencionar que a qualidade fica em primeiro plano, até pela imagem de confiabilidade que a marca tem perante os clientes. Seguido disso, o Prazo de Entrega, normalmente,

fica em segundo lugar, seguido pelo Custo que, dependendo do cenário econômico que a empresa enfrente, pode ser levado como importante ou não.

1.2 Objetivo

Este estudo visa resolver um problema real de *layout* da linha de montagem de um componente automotivo produzido por uma renomada indústria multinacional.

O estudo visa, a partir de dados reais de produção (p. ex., dados de fornecimento, dimensões e quantidades de máquinas), propor uma abordagem original para a solução do problema de *layout* a partir do uso de métodos meta-heurísticos e BB.

1.2.1 Objetivos Específicos

Para a solução do problema de *layout*, foram definidos os seguintes objetivos específicos:

- Apresentar uma abordagem em duas etapas para a solução do problema de *layout*. Inicialmente, soluciona-se o problema de formação de células (PFC), definindo um agrupamento das peças aos recursos (máquinas);
- Após a etapa de formação de célula, propõe-se uma segunda etapa, de solução do *layout* de fábrica;
- Será utilizada uma abordagem heurística para a solução das duas etapas, a BT Híbrida, expondo a utilidade do método para problemas reais de otimização em manufatura.

1.3 Justificativa

Algumas empresas automotivas têm, historicamente, gasto milhões de reais em alterações de *layout* (visando sempre obter ganhos como redução de estoques e quantidade de máquinas, aumentos de produtividade, etc.). Há diversos agravantes neste caso, como a demanda variável do cliente, ciclo de vida do produto cada vez mais curto e aumento na competição global (WANG, LIN e WU, 1998). Estes fatores

fazem com quem a busca de um *layout* que seja, ao menos, compatível com as necessidades da empresa, se torne cada vez mais complexo e distante de ser atingido e conquistado, até por se tratar de um problema que envolve diversas variáveis.

1.4 Estrutura da Dissertação

Na sequência deste estudo, o capítulo 2 tratará da revisão bibliográfica, onde se faz uma revisão de vários métodos heurísticos que possuem diversas aplicações reais para problemas de otimização. Ainda dentro do capítulo 2, são abordados alguns conceitos importantes para o trabalho em questão, como o uso da tecnologia de grupo ou formação de células (que são aplicados mais amplamente nas empresas por sua facilidade e praticidade) e também o conceito e soluções para o problema de estudo de *layout*. No capítulo 3, será explicada a metodologia empregada no estudo, sendo apresentados todos os dados do objeto de estudo e métodos de otimização empregados no presente estudo. No capítulo 4, serão apresentados os resultados baseados nos dados levantados dentro do escopo deste estudo, para a empresa em questão. No capítulo 5, são apresentadas as conclusões das aplicações feitas neste trabalho, a partir da extração dos dados e verificações feitas no estudo presente, de modo a comprovar a efetividade dos métodos empregados.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo são mostrados os trabalhos que utilizam métodos heurísticos bastante significativos encontrados na literatura nos últimos anos, sendo apresentados “puros” ou combinados entre si ou com outros (chamados de “híbridos”), de modo a ter-se o maior ganho possível nos problemas de otimização.

2.1 Métodos de Otimização

Segundo Papaioannou e Wilson (2010), os métodos heurísticos são bastante úteis para problemas referentes à manufatura, pois produzem soluções aceitáveis em um tempo razoável. Por outro lado, para Papaioannou e Wilson (2010), apesar da boa aplicabilidade dos métodos heurísticos em problemas de formação de células, os mesmos têm sido testados apenas em problemas de tamanho relativamente pequeno. Uma alternativa para esse problema (TALBI, 2002), seria o uso da metaheurística híbrida (*Hybrid Metaheuristics*), que possui a habilidade de combinar um método meta-heurístico com outra técnica de otimização de modo a proporcionar uma alta flexibilidade com uma maior eficiência; isso é possível através da combinação das qualidades de cada método.

Alguns estudos (GONÇALVES E RESENDE, 2004 e JAMES *et al.*, 2007) mostram resultados promissores no uso de algoritmo híbrido que combinava busca local com algoritmo genético (AG). Em um destes estudos (JAMES *et al.*, 2007), foi apresentado um algoritmo genético (AG) híbrido em um problema de formação de células, combinando-se a busca local com o agrupamento padrão do algoritmo genético. A qualidade dos resultados encontrados superou o que se encontra na literatura ao comparar com o uso isolado do AG, através de uma melhor qualidade de solução e uma menor variabilidade nas soluções encontradas. Diversos pesquisadores têm aplicado algoritmos meta-heurísticos (PAPAIOANNOU e WILSON, 2010), sendo os cinco mais utilizados o *simulated annealing* (SA), busca tabu (BT), algoritmo genético (AG), otimização de colônia de formigas ou *ant colony optimization* (ACO) e otimização de enxame de partículas ou *particle swarm optimisation* (PSO).

É interessante ainda notar que os métodos *simulated annealing* (SA) e busca tabu (BT) têm em comum, no processo de busca, partir de um estado inicial e

descrever uma trajetória dentro dos limites do problema; sendo que o SA é o mais antigo dos métodos descritos e a BT é uma das buscas mais bem sucedidas para a aplicação em problemas de formação de células (PAPAIOANNOU e WILSON, 2010).

No estudo proposto por Vakharia e Chang (1997), foram desenvolvidos dois Métodos Heurísticos para problemas de Formação de Células, baseados em SA e BT. Neste estudo, a função objetivo era de minimizar os custos totais das máquinas, bem como os custos de manuseio de materiais para movimentos intracelulares. Neste estudo, o SA sobrepôs o BT em termos de qualidade na solução e tempo de processamento computacional.

É interessante mencionar que, no estudo feito por Wu *et al* (2006), o desempenho do método empregado pode ser avaliado, em termos de qualidade da solução encontrada, pelo custo total de movimentos (intracelulares e intercelulares) e pelo número de elementos excepcionais (EE), que são o mesmo que movimentos intercelulares, ou seja, operações feitas na peça (WIP) fora da célula de manufatura, que, por sua vez, foi formada a partir da definição de famílias de produtos. Inclusive, para Xambre e Vilarinho (2003), um dos principais problemas que envolvem a formação de células, e que, conseqüentemente, é o principal foco de vários estudos de formação de células, é a questão do movimento intercelular, que deve ser minimizado.

2.1.1 Busca Tabu (BT)

Conforme Wu *et al* (2009), a vizinhança de uma determinada solução é o conjunto das soluções factíveis que podem ser atingidas através de apenas um movimento ou iteração.

Meta-heurísticas, conforme Pureza e Morabito (2003) são utilizadas para superar a solução local, buscando soluções superiores em termos de qualidade; entre os métodos aplicados, estão o *simulated annealing* (SA) e a busca tabu (BT).

No estudo, feito por Pureza e Morabito (2003), ocorre à incorporação da busca tabu simples à heurística de blocos, dentro de um problema de empilhamento de caixas em paletes. O funcionamento da busca tabu simples (GLOVER e LAGUNA, 1997) ocorre de forma contrária às versões mais avançadas com memória de longo prazo, além de permitir que haja disponibilidade de um padrão em qualquer

iteração e restringir atributos já visitados após certo número de iterações (PUREZA e MORABITO, 2003). Segundo Lei & Wu (2005), a busca tabu trabalha com o princípio de gerar novas soluções vizinhas, através do movimento de troca de posição de duas variáveis envolvidas no estudo (no caso de um problema de *layout* fabril, duas máquinas) por sorteio.

A busca tabu (CHUNG *et al*, 2011) trabalha com aperfeiçoamentos na busca de soluções melhores, além de evitar áreas recentemente visitadas através da procura de novos locais promissores (a chamada vizinhança). Segundo Glover (1990), sugere-se que a melhor Lista Tabu é aquela com tamanho entre cinco e doze.

Lei & Wu (2006) trabalharam com uma busca tabu multiobjetiva (MOTS) baseada em uma otimização conforme pareto, visando minimizar a soma ponderada dos movimentos intracelular e intercelular, bem como a variação de carga na célula. Pode-se perceber neste estudo que a MOTS é bastante útil em otimizações com múltiplos objetivos.

Conforme Lei & Wu (2005), uma busca tabu multiobjetiva (MOTS) baseada no método de pareto para otimização, apresenta agrupamentos de máquinas com objetivos múltiplos (conforme Aiello *et al* - 2013, problemas multiobjetivos, que tratam de vários objetivo ao mesmo tempo sem, contudo, torná-lo um só), gerando vários ganhos, como redução nos custos totais. Os problemas multiobjetivos, ainda segundo Lei & Wu (2005), consideram simultaneamente os fatores em uma linha de produção, como tempo de setup, tempo de processamento e sequencial de operações. No entanto, Lei & Wu (2005) e Lei & Wu (2006) não consideram restrições de capacidade de produção nem roteiros alternativos.

2.1.2 Simulated Annealing (SA)

De acordo com Wang e Wu (2001), assim como a busca tabu (BT), o método *simulated annealing* (SA) visa minimizar o transporte intra e intercelular, dentro dos problemas de formação de células. Este método foi desenvolvido por Metropolis *et al* (1953), para simular o processo de recozimento e têmpera. Segundo Aarts e Korst (1989), este método foi adaptado por Kirpatrick *et al.* (1983), correlacionando o processo físico (simulação do resfriamento lento da matéria) com processos de otimização. O custo de uma solução é equivalente à energia interna de um estado.

Depois de gerada uma solução inicial, faz-se várias tentativas em busca de soluções vizinhas. O algoritmo básico deste método está exibido na figura 1, onde a variável T é a temperatura. Quanto maior for esta variável, maior a probabilidade de aceitação de uma solução ruim. O interessante é que, ao aceitar a solução pior em um dado instante, pode-se evitar ficar preso à uma região de ótimo local, em busca da solução global ótima, através de um algoritmo robusto e efetivo (WANG *et al.*, 1998). A variável Δ é a diferença entre a solução candidata e a solução atual. Em um problema de maximização, por exemplo, se $\Delta < 0$, a solução encontrada é pior que a atual.

De acordo com Wang e Wu (2001), entre outros (EGLESE, 1990; PIRLOT, 1996), o *Simulated Annealing* (SA) é um algoritmo que lança mão da busca local, conseguindo obter soluções com resultados aceitáveis em problemas de otimização relativamente complexos. O problema, segundo Wu *et al* (2009), são essas chamadas soluções locais, piores ou iguais à solução ótima, dependendo da complexidade do problema. A partir disso, o método *simulated annealing* (SA) é conhecido como um método que contempla tanto o procedimento de *hill-climbing* (convergência a um ponto ótimo, que pode ser local ou global) e o procedimento de *backtracking* (“fuga” de uma região de ótimo local).

O método *simulated annealing* (SA) possui a habilidade de escapar de soluções ótimas locais. Além disso, o SA se popularizou na solução de problemas de formação de células e de *layout* nos últimos anos, tendo se tornado um método efetivo de busca para estes problemas.

O algoritmo básico do SA está apresentado conforme figura 1; conforme esta figura e de acordo com Xambre e Vilarinho (2003), uma solução vizinha S' da solução inicial S é gerada. Quando a função objetivo é de minimização, se o $\Delta \leq 0$, a solução candidata (ou vizinha) é aceita, já que é melhor que a solução atual. Se $\Delta > 0$, a probabilidade da solução nova ser aceita é de $\exp(-\Delta/T)$, sendo que a temperatura T diminui a cada iteração, deixando menores as chances de aceitação de uma nova solução pior que a solução atual. A partir daí, é possível piorar a solução com o intuito de explorar outras regiões dentro do espaço de busca (vizinhança), criando, assim, a possibilidade de chegar a uma solução de melhor qualidade (podendo “fugir” de uma região de solução ótima local). Da mesma forma, a partir da figura 1, pode-se verificar que a chance desta aceitação de uma solução pior depende tanto do valor da temperatura, quanto da diferença entre os valores da

função objetivo (Δ) das duas soluções que estão sendo comparadas.

O estudo de Sahin e Ciric (1998) apresenta uma metodologia para resolver problemas de programação de dois níveis, que são problemas que não podem ser resolvidos utilizando técnicas convencionais de otimização. O problema de otimização interna é relaxado e a amplitude deste relaxamento é controlada por um parâmetro análogo à temperatura no *simulated annealing*. Já a otimização externa é solucionada por *simulated annealing*, criando uma aproximação de dupla temperatura.

É também interessante mencionar o estudo feito por Tavakkoli *et al* (2008), que leva em conta um horizonte de planejamento ao longo de vários períodos, levando em conta que o *mix* dos produtos e a demanda da produção são diferentes ao longo dos períodos; ou seja, o que é bom agora não é necessariamente bom depois, em termos de composição de células. Neste estudo, foi visado minimizar o movimento intercelular entre outros. O método utilizado neste estudo foi o SA, que provou ser bastante eficiente, com erros menores de 4% comparado à solução ótima via método exato.

O estudo proposto por Xambre e Vilarinho (2003) menciona casos em que se tem máquinas múltiplas e idênticas funcionalmente, e, daí, é inserido um grau adicional de liberdade, que é a alocação de operações para máquinas específicas. O estudo em questão utilizou uma organização híbrida de processos, tendo células de manufatura consideradas “lado a lado” com arranjos de orientação por processo. Ainda, conforme Xambre e Vilarinho (2003), o uso do Método SA para problemas de larga escala (no caso do problema analisado, com múltiplas máquinas similares) pode ter um bom desempenho, principalmente, em termos de tempo computacional.

O estudo gerado por Wang *et al* (1998), acerca do *layout* de recursos dentro de um Sistema de Manufatura Celular (CMS), modificou o mecanismo de geração da vizinhança próprio do *simulated annealing* (SA) e fez uma comparação com o estudo de Kouvelis e Chiang (1992), produzindo soluções de mesma qualidade, mas com menor tempo de processamento.

No estudo de Wu *et al* (2009), ainda, é indicado que o fator mutação permite que o algoritmo encontre locais não visitados antes, encontrando-se soluções de melhor qualidade e mais eficientes que as usuais, que são normalmente encontradas no *Simulated Annealing* (SA). Isso se torna ainda mais evidente ao se

encontrar problemas maiores envolvidos (WU *et al*, 2009). Neste mesmo estudo de Wu *et al* (2009), utilizou-se o recurso de diversos contadores dentro do algoritmo, para diminuir os tempos computacionais e escapar da solução local ótima.

Levando-se em conta o problema da demanda variável de um produto ao longo de seu ciclo de vida (WANG e WU, 2001), pode-se utilizar o *simulated annealing* (SA) para problemas de atribuição bi quadrática, pois fica inviável utilizar os métodos exatos devido ao fato do tempo computacional aumentar muito à medida que o problema aumenta. Em um problema bi quadrático pode-se ter restrições lineares, mas com a função objetivo apresentada como a soma ponderada do produto de diversas variáveis envolvidas neste.

2.1.3 Algoritmo Genético (AG)

Conforme Papaioannou e Wilson (2010), o algoritmo genético (AG) é baseado na evolução das espécies, que é iniciada de uma população gerada de forma aleatória e alterada ao longo de várias gerações. A cada geração, a adaptação de cada indivíduo é avaliada e, baseado na avaliação desta adaptação, os indivíduos são selecionados e modificados (através de recombinações e mutações), formando novas populações. O algoritmo termina assim que se atinja o número máximo de gerações ou quando se atinge um nível satisfatório de adaptação para a população. Conforme Holland (1975), o Método AG pode ser utilizado para estudar similaridades em um esquema curto e de alto desempenho. Além disso, o mesmo ilustra a necessidade de encontrar a melhor solução através dos efeitos de reprodução, cruzamentos e mutações.

Conforme Wu *et al* (2006), os algoritmos genéticos (AG) combinam elementos de busca estocástica para estudar e explorar o espaço de busca, visando obter boas soluções. Além disso, o método AG também pode ser combinado com outras heurísticas, formando algoritmos híbridos, o que prova que este método tem a qualidade de ser flexível e de, conseqüentemente, se sobressair com relação a outros métodos heurísticos tradicionalmente utilizados.

Segundo Wu *et al* (2009), o algoritmo genético (AG) explora soluções diversificadas através dos princípios de seleção, mutação e cruzamento. No entanto, essas soluções diversificadas sofrem por uma convergência fraca. Esses autores indicam que isso pode ser superado através da combinação ou sinergia do SA com

o AG (vide WONG, 2001 e SOKE e BINGUL, 2006). Esta idéia foi implementada em alguns problemas de peças com mais de uma rota, tendo resultados positivos. Para evitar excessivo tempo de processamento computacional, somente o fator mutação do algoritmo genético (AG) é adotado, principalmente para escapar de soluções locais e evitar uma convergência muito rápida (WU *et al*, 2006). A partir daí, existe a opção de adotar um Algoritmo Híbrido (WU *et al*, 2009) para melhorar em qualidade e eficiência as soluções encontradas.

No estudo proposto por Mak *et al.* (2000) foi apresentada uma aproximação genética adaptativa em busca da solução ótima em problemas de Formação de Células. O princípio deste estudo foi agrupar máquinas e peças através da construção de uma matriz de incidência, em um sequenciamento em linhas e colunas destes elementos, visando aumentar a “energia de união” dos mesmos. Já o estudo proposto por Defersha e Chen (2006) propôs, através do AG, uma função que visava minimizar fatores como o custo de investimento em máquinas, manuseio de materiais e o custo com ferramentas, entre outros. É interessante notar que, devido à complexidade do modelo proposto, apenas problemas de tamanho pequeno poderiam ser resolvidos através dos métodos exatos. Para problemas considerados grandes, comparando-se o tempo computacional despendido com a qualidade da solução encontrada, o algoritmo genético (AG) demonstrou melhor desempenho.

Wu *et al* (2007) propõe um estudo de algoritmo genético hierárquico, onde, ao mesmo tempo, são formadas células de manufatura e, também, se determina o *layout* dos grupos de células. O estudo feito mostrou-se efetivo em termos de qualidade de solução e tempo de convergência.

No estudo proposto por Wu *et al* (2006), o algoritmo desenhado para a formação de células de manufatura inclui uma estrutura hierárquica de cromossomos, com o intuito de codificar um esquema de seleção que considera dois fatores: *i*) funções de ajuste correlatas (mais eficiente) e *ii*) um fator de mutação do grupo (que, conforme estudo, tem um impacto bastante relevante no tempo computacional) para aumentar a probabilidade de mutação, sendo que ambos estes fatores precisam trabalhar juntos para obter bons resultados. Tudo isso visa melhorar a qualidade da solução e acelerar a convergência na busca da mesma. Neste mesmo estudo, buscou-se também comparar vários parâmetros em um

Método AG, como tamanho populacional, taxa de mutação e taxa de cruzamento, para verificar qual o desempenho do Método AG.

2.2 Tecnologia de Grupo ou Formação de células

A tecnologia de grupo ou *group technology (TG)*, segundo Wu *et al* (2009), parte do princípio da similaridade existente entre os tipos de peças em uma linha de produção. Conforme Lei & Wu (2005), uma aplicação bastante relevante da chamada tecnologia de grupo (GT) é a manufatura celular (CM), onde peças que guardam entre si uma relação de similaridade são agrupadas em células de máquinas e várias metodologias têm sido empregadas para, dentro da manufatura celular, minimizar alguns inconvenientes, como o transporte intercelular.

Gupta *et al.* (1996) e Venugopal e Narendran (1992) utilizaram um modelo bicritério utilizando-se dos algoritmos genéticos (AG). Já Khoo *et al.* (2003) propôs uma abordagem utilizando também algoritmos genéticos para atribuir máquinas em células de manufatura, sujeito a fatores como custo de *setup*, por exemplo.

Os estudos de Lei & Wu (2005) e de Venugopal e Narendran (1992) visam a minimização na carga e na movimentação dentro e fora de uma célula; este último visando reduzir custos com transporte. A figura 2 apresenta o *layout* de dois tipos de células, onde se pode verificar a permutação das células em cada caso; permutação essa que está inserida dentro do princípio básico da busca tabu (BT) (LEI & WU, 2005).

Ainda de acordo com Rodrigues *et al* (2013), o conceito de células de manufatura é utilizado a partir do agrupamento de máquinas e peças. As peças são agrupadas em famílias de modo a melhorar a produtividade e eficiência nos sistemas de manufatura. Conforme mencionado por Wu *et al* (2010), entre os benefícios da adoção da tecnologia de grupo (GT) para a formação da manufatura celular (CM), estão a redução no inventário, na menor necessidade de capacidade produtiva, custos em geral, menores *lead times*, rápida resposta para mudanças internas e externas (como alterações no *mix* de produção, nas demandas e falhas de máquinas), entre outros. É importante mencionar, por outro lado, que no uso de células de manufatura, conforme Rodrigues *et al* (2013), a tarefa de balanceamento de carga de produção nem sempre é uma tarefa fácil, podendo-se ter máquinas com excesso ou falta de capacidade produtiva.

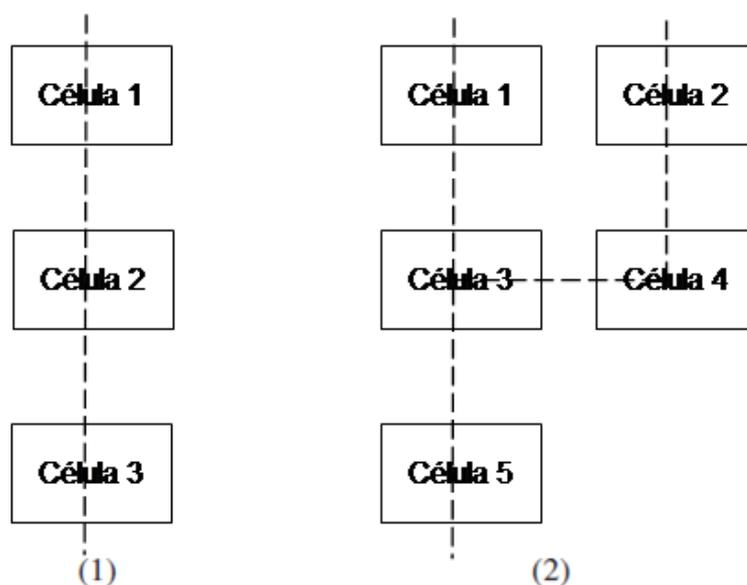


Figura 2 - Dois tipos diferentes de *layout* celular: (1) linear e (2) com duas colunas (traduzido de Lei & Wu, 2005).

Segundo Chung *et al* (2011), as células de manufatura são, em termos práticos, a implementação de Tecnologia de Grupo, onde peças que guardam entre si uma relação de similaridade são unidas na forma de famílias. Ainda de acordo com Chung *et al* (2011), por outro lado, não são facilmente obtidos, e podem demandar aproximações sofisticadas, seja através de modelos de Programação Matemática (ALBADAWI, BASHIR e CHENM, 2005), Algoritmos Heurísticos (WU, CHANG e CHUNG, 2008) e uso do Método do Coeficiente Baseado em Similaridade (ALHOURANI e SEIFODDINI, 2007). Ainda de acordo com Chung *et al* (2011), este último método consegue obter melhores resultados ao se basear em máquinas ou processos, ao invés de se basear em produtos ou peças, por questões de “carga” computacional; o que pode produzir, em estágios iniciais, boas soluções iniciais, enquanto a Busca Tabu gera soluções cada vez melhores através da movimentação na vizinhança em estágios posteriores.

Conforme Selim *et al.* (1998), a tecnologia de grupo tira vantagem das similaridades no processo de manufatura e no desenho das máquinas. Ainda conforme Wu *et al* (2009), a manufatura celular é nada mais nada menos que a aplicação da tecnologia de grupo (GT) em um processo produtivo.

Na tecnologia de grupo, sempre se busca a minimização do transporte entre células (intercelular) e dentro das células (intracelular), conseguindo-se obter uma

redução significativa de custos de transporte (PAPAIOANNOU e WILSON, 2010; LOZANO *et al.*,2001).

Conforme Wu *et al* (2006), a formação de células de manufatura (CM) envolve decisões acerca de três fatores: formação de células, *layout* do grupo e formação de grupos de peças e máquinas. Ainda, no estudo proposto por Wu *et al* (2006), através do uso do algoritmo genético (AG), são formadas as células de manufatura e ao mesmo tempo formado o *layout* do grupo. Ainda conforme Rodrigues *et al* (2013), os problemas de formação de células envolvem, primeiramente, a atribuição das máquinas às células de manufatura para, depois, atribuir as peças e respectivos roteiros às respectivas células, visando minimizar o chamado movimento extracelular das peças, que se caracteriza por processamentos das peças fora das células a que foram alocadas/designadas. Ainda conforme Irani (1999), para se avaliar a qualidade da célula formada, leva-se em conta critérios como a minimização de operações ou movimentos extracelulares, conseqüente minimização de carregamento nas máquinas causado por movimentos extracelulares, bem como a minimização do fluxo de peças entre as células de manufatura formadas.

Segundo Papaioannou e Wilson (2010), a manufatura celular é um sistema híbrido que tira vantagens da produção por *jobbing* (flexibilidade) e em massa (fluxo eficiente e alto volume de produção). Segundo Wemmerlov e Hyer (1989), o objetivo da manufatura celular é reduzir os tempos de setup (utilizando o ferramental da família a ser produzida na célula) e *lead time* na linha de produção (reduzindo o tempo de *setup*, movimentação, tempo de espera), reduzindo assim o inventário e tempo de resposta ao mercado (através de lotes pequenos).

Conforme Papaioannou e Wilson (2010) e Selim *et al.* (1998), as células de manufatura são formadas através da sequência de atividades: 1) famílias de peças são formadas de acordo com os requisitos dentro de seus processos (identificação de famílias de peças); 2) as máquinas são agrupadas em células; 3) as famílias de peças são agrupadas às células.

Papaioannou e Wilson (2010) indicam que, nas últimas três décadas, diversos trabalhos foram realizados em torno de problemas de formação de células, seguindo em três categorias diferentes: 1) Métodos informais – identificação visual de peças e agrupamento em famílias de forma visual, comum em problemas simples, onde se tem pequeno número de peças e máquinas; 2) Métodos de análise de número das peças - sistema de codificação para agrupar e classificar peças de modo a formar

famílias (ideal para redução de componentes); 3) Métodos baseados em produção, sendo estas as classificações possíveis:

a) análise de agrupamento: o objetivo é de agrupar cada objeto, entidade ou atributo em grupos de tal forma que elementos individuais dentro de um grupo possuem um alto grau de associação entre eles, e uma pequena associação entre os grupos. Os procedimentos de agrupamento podem ser de forma matricial, hierárquica ou não-hierárquica;

b) métodos de programação matemática: podem ser lineares ou não-lineares, são aptos a incorporar sequências ordenadas de operações, tempos de *setup* e processamento, e assim por diante. No entanto, quanto mais dados envolvidos, maior é o tempo de processamento. No artigo de Selim *et al.* (1998) há um comparativo entre vários estudos, onde se busca minimizar o custo atribuído para a peça nas operações, máquinas, trabalhadores e ferramental para as células;

c) algoritmos heurísticos e metaheurísticos: os artigos anteriormente citados neste capítulo (usando *simulated annealing*, algoritmos genéticos e busca tabu) são exemplos de heurísticas e metaheurísticas.

Conforme Papaioannou e Wilson (2010), a figura 3 propõe a classificação dos métodos utilizados em programas de formação de células (CF), conforme mencionado acima, e os resultados diferem entre si dependendo dos elementos envolvidos em um problema, como objetivos e restrições envolvidos, aproximação utilizada, tamanho do problema (quantidade de máquinas, quantidade de números de tipo de peças) e qualidade da solução.

As figuras 4 e 5 (Papaioannou e Wilson, 2010) apresentam comparativos entre os estudos que empregam as principais metodologias para problemas de formação de células de manufatura no período de 1997 a 2008. Nestes estudos, os objetivos principais buscados foram relacionados à minimização de movimento intercelular, considerações sobre carga de trabalho em máquinas e questões de custo (por exemplo, custo de *setup*, custo de operação de máquina, etc..). Outras restrições foram também consideradas, como sequência de operação peça/máquina, múltiplas máquinas do mesmo tipo, tempos de processamento ou utilização peça/máquina, planos de processo alternativos e horizonte de tempo multi-período, quando se possui um problema de formação de célula dinâmico. De acordo com a figura 5, podem-se enunciar algumas conclusões de Papaioannou e Wilson (2010):

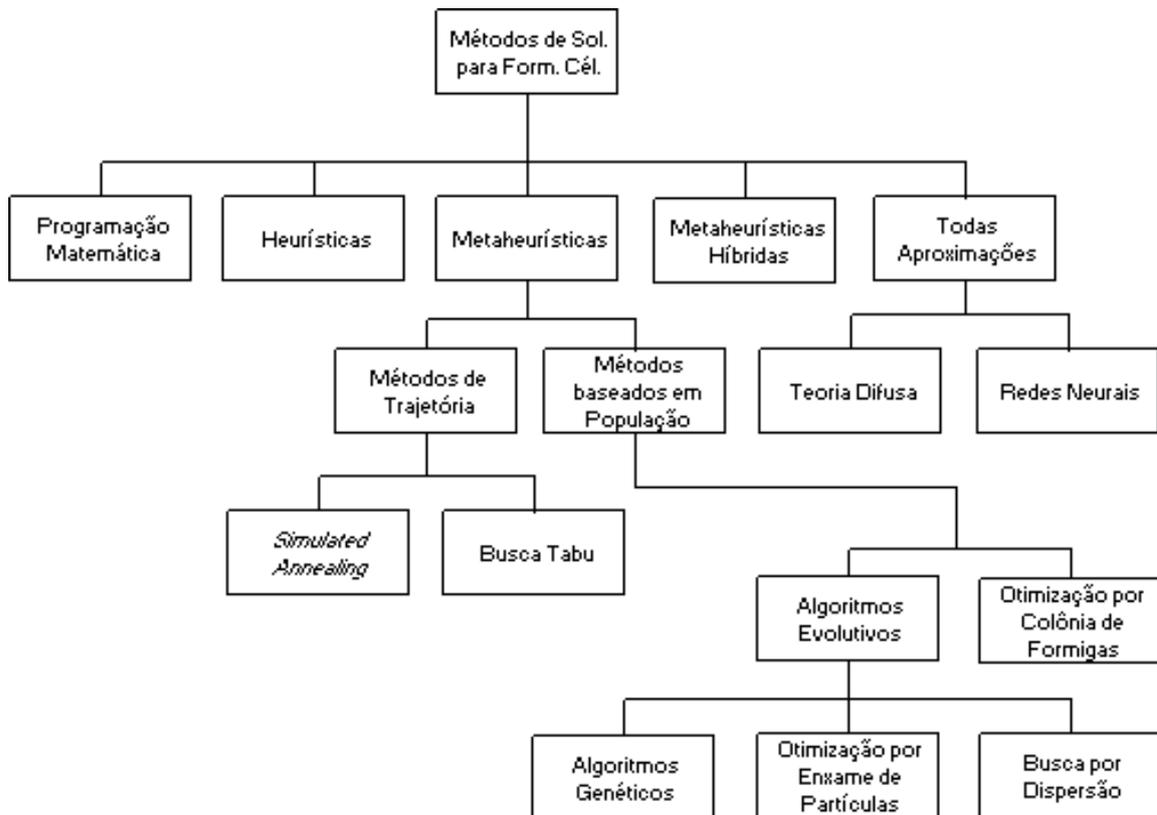


Figura 3 - Classificação de métodos para solução de problemas de formação de células (CF)
(traduzido de Papaioannou e Wilson, 2010).

- A maior parte dos problemas de Formação de Células visa minimizar o movimento intercelular (entre células de manufatura), e outros, em menor quantidade, visam minimizar custos e/ou carga de máquina, como função objetivo;
- Os Métodos Exatos são implementados com dificuldade dependendo do tamanho do problema, devido a limitações no tempo computacional (ou de processamento);
- A complexidade do problema é determinada principalmente a partir da quantidade de células e de máquinas. A quantidade de tipos de peças é um fator secundário que influencia no grau de dificuldade do problema, sendo que este aumenta à medida que também aumenta o número de máquinas.

Ainda segundo Chung *et al* (2011), é importante também dizer que, tradicionalmente, quando a alocação de trabalho e a Formação de Células são executados, supõe-se que as máquinas da linha de produção sejam totalmente confiáveis, o que não condiz com a realidade. O estudo proposto por Chung *et al* (2011), leva em conta, além da consideração de rotas alternativas, importantes colocações sobre confiabilidade de máquinas, que acabam, por fim, envolvendo custos não desejados, como custos de reparos, custos de perda de produção e perda de capacidade, por exemplo. Entre os benefícios do emprego desta forma de *layout* estão as reduções nos tempos de *setup*, custos de manuseio de material, inventário de peças em processo (*WIP* ou *Work-in-Process*), entre outros. Quando cada peça possui rotas alternativas na produção, o Problema de Formação de Célula passa a ser um Problema de Formação de Célula Generalizado (KUSIAK, 1987).

Foi apresentado por Wu *et al* (2010) que os algoritmos genéticos (AG) iniciam a otimização em um problema de formação de células (CF) com um conjunto de possíveis soluções, e não apenas uma, como no caso de BT. O comportamento do AG é marcado pelo equilíbrio entre exploração e avanço no espaço de busca, equilíbrio este muito afetado por parâmetros estratégicos, como por exemplo tamanho de população. O desempenho de um problema que utiliza AG depende muito destes parâmetros. Ainda, segundo Wu *et al* (2010), apesar de o método AG ser um processo adaptativo e dinâmico, o uso de parâmetros constantes (comumente utilizados) é um contraste ao princípio evolucionário deste tipo de método. Então, é natural alterar os parâmetros ao longo da execução do algoritmo, segundo Gen e Chen, (2000). Já no estudo proposto por Rodrigues *et al* (2013), os autores utilizam duas abordagens híbridas da Busca Tabu e do *Simulated Annealing*, em problemas de formação de células, para atribuir as máquinas às células para, posteriormente, utilizar o *branch-and-bound* (ou Busca em árvore) para atribuir as peças (e respectivos roteiros) às respectivas células.

No.	Fonte Referência	No.	Fonte referência
1.	Zolfaghari and Liang (1997)	27.	Islir (2005)
2.	Vakharia and Chang (1997)	28.	Prabhakaran et al. (2005)
3.	Aljaber et al. (1997)	29.	Venkumar and Haq (2005)
4.	Masnata and Settineri (1997)	30.	Andrés and Lozano (2006)
5.	Selim et al. (1998)	31.	Foulds et al. (2006)
6.	Gravel et al. (1998)	32.	Diaby and Nsakanda (2006)
7.	Lozano et al. (1999)	33.	Lei and Wu (2006)
8.	Sofianopoulou (1999)	34.	Nsakanda et al. (2006)
9.	Liang and Zolfaghari (1999)	35.	Defersha and Chen (2006)
10.	Susanto et al. (1999)	36.	Venkumar and Haq (2006)
11.	Mak et al. (2000)	37.	Torkul et al. (2006)
12.	Ravichandran and Rao (2001)	38.	Wu et al. (2007)
13.	Lozano et al. (2001)	39.	James et al. (2007)
14.	Mukattash et al. (2002)	40.	Tavakkoli-Moghaddam et al. (2007)
15.	Guerrero et al. (2002)	41.	Saidi-Mehrabad and Safaei (2007)
16.	Soleymanpour et al. (2002)	42.	Won and Currie (2007)
17.	Lozano et al. (2002)	43.	Defersha and Chen (2008)
18.	Logendran and Karim (2003)	44.	Durán et al. (2008)
19.	Spiliopoulos and Sofianopoulou (2003)	45.	Safaei et al. (2008)
20.	Park and Suresh (2003)	46.	Yang and Yang (2008)
21.	Won and Lee (2004)	47.	Tavakkoli-Moghaddam et al. (2008)
22.	Kim et al. (2004)	48.	Kao and Li (2008)
23.	Wu et al. (2004)	49.	Spiliopoulos and Sofianopoulou (2008)
24.	Solimanpur et al. (2004a)	50.	Papaioannou and Wilson (2008)
25.	Solimanpur et al. (2004b)	51.	Megala and Rajendran (2008)
26.	Tavakkoli-Moghaddam et al. (2005)	52.	Bajestani et al. (2009)

Figura 4 - Lista de fontes referentes a estudos de Formação de Células (traduzido de Papaioannou e Wilson (2010).0111

	O1	O2	O3	C1	C2	C3	C4	C5	MP	H	T	S	A	G	P	SS	F	N	DS	O	D	C	
1.	✓								✓									✓	(40 × 100)			N	C
2.		✓	✓						✓		✓	✓							(78 × 325)	✓		N	ZOOM/XMP
3.	✓									✓	✓	✓							(100 × 500)			N	Fortran
4.																			(7 × 25)	✓		Y	
5.	✓	✓	✓	✓					✓													N	N/A
6.	✓						✓												(15 × 30)			N	Bor. Delphi
7.	✓										✓								(30 × 90)		✓	Y	
8.	✓				✓				✓			✓							(20 × 30)			N	Fortran
9.	✓				✓	✓			✓										(40 × 100)			Y	C
10.	✓								✓									✓	(30 × 41)			Y	Turbo C++
11.	✓								✓										(40 × 100)	✓		Y	
12.	✓								✓										(9 × 9)			Y	
13.	✓			✓					✓										(50 × 100)		✓	Y	
14.	✓				✓	✓	✓		✓		✓								(13 × 13)			Y	
15.	✓								✓										(20 × 40)			Y	
16.	✓								✓										(40 × 100)			Y	Borland C++
17.	✓								✓										(50 × 150)			Y	
18.	✓	✓							✓										(15 × 30)		✓	N	LINGO & C++
19.	✓								✓			✓							(30 × 30)		✓	N	Fortran
20.	✓			✓					✓										(25 × 40)			Y	
21.	✓								✓									✓	(50 × 150)		✓	N	Pascal & Lindo
22.	✓		✓						✓										(40 × 40)			N	
23.	✓								✓			✓							(18 × 5)		✓	N	Cplex & C
24.	✓	✓							✓										(15 × 30)			N	
25.	✓								✓										(8 × 20)		✓	N	
26.	✓	✓							✓			✓	✓						(20 × 30)			N	Lingo
27.	✓			✓					✓										(40 × 100)			Y	
28.	✓				✓				✓										(40 × 100)			Y	
29.	✓								✓										(40 × 100)			Y	C++
30.	✓								✓										(10 × 10)		✓	N	C
31.	✓	✓			✓	✓			✓			✓							(5 × 7)		✓	N	XPRESS ^{MP} & C
32.	✓	✓			✓	✓			✓										(10000 × 25)		✓	N	Lagr. Relax. & C
33.	✓	✓							✓			✓							(15 × 30)			Y	
34.	✓	✓							✓										(50 × 1000)		✓	Y	C
35.	✓	✓		✓	✓				✓										(30 × 90)		✓	N	Lingo
36.	✓								✓										(40 × 100)			Y	MatLab
37.	✓								✓										(18 × 25)			Y	
38.	✓			✓					✓													N	Visual C++
39.	✓		✓						✓										(40 × 100)		✓	Y	Visual Basic
40.	✓	✓		✓	✓	✓	✓		✓										(10 × 10)		✓	N	Lingo
41.	✓	✓							✓										(9 × 10)		✓	N	Lingo
42.	✓	✓		✓	✓				✓										(16 × 43)		✓	Y	C++
43.	✓	✓							✓										(6 × 12)		✓	N	ILOG CPLEX
44.	✓	✓							✓										(12 × 12)			N	
45.	✓	✓		✓					✓										(6 × 8)			N	Lingo
46.	✓	✓							✓										(46 × 105)			Y	
47.	✓	✓							✓			✓							(17 × 30)		✓	N	Lingo
48.	✓								✓										(50 × 150)		✓	Y	Java
49.	✓			✓	✓				✓										(37 × 53)		✓	Y	Fortran
50.	✓	✓		✓	✓				✓										(9 × 9)		✓	N	XPRESS ^{MP}
51.	✓								✓										(40 × 100)		✓	Y	Visual C++
52.	✓	✓		✓	✓				✓										(9 × 10)		✓	Y	MatLab & ANOVA

Figura 5 - Comparação dos métodos de formação de células (traduzido de Papaioannou e Wilson, 2010).

Devido ao grau de dificuldade de problemas de Formação de Células, metodologias Heurísticas e Meta-heurísticas têm sido adotadas para problemas mais complexos – os métodos mais utilizados nas últimas décadas têm sido o Algoritmo Genético (AG), a Busca Tabu (BT) e o *Simulated Annealing* (SA), geralmente empregando funções objetivo com apenas um critério de otimização. Além disso, apenas metade dos estudos foram feitos fazendo-se comparações com outras metodologias, pelo fato das restrições e funções objetivo geralmente diferirem;

- Em alguns casos, a Lógica *Fuzzy* (ou seja, a consideração de incertezas, onde se utiliza uma matriz binária para se representar se um elemento faz parte ou não de um grupo) foi aplicada em alguns estudos, principalmente onde se tem o propósito de realizar agrupamentos e, principalmente, onde se possui incertezas em certo fatores, como, por exemplo, a variação na demanda dos produtos ao longo de cada período;
- Os métodos de Redes Neurais, PSO e ACO têm sido empregados apenas recentemente para problemas de Formação de Células (CF);
- Entre os estudos considerados nas figuras 4 e 5, apenas poucos aplicaram a combinação de Métodos através de formulações Híbridas.

As figuras 6 e 7 (WANG et al, 1998) ilustram a disposição física em “U” de máquinas dentro de um Sistema de Manufatura Celular (CMS), onde se pode perceber que cada célula de montagem e pré-montagem é composta por estações de trabalho e células de manufatura.

Sofianopolou (1999) propôs um estudo onde foi proposto um modelo não linear para um problema de formação de células, onde havia várias cópias de máquinas com rotas de processo alternativas para as peças. Logendran e Karim (2003) propuseram, por sua vez, um estudo onde é levada em consideração a capacidade do transportador de materiais. Outros elementos como limitações de capacidade de máquinas, quantidade máxima de máquinas em uma célula, podem ser levados em conta. No estudo de Wu *et al.* (2004), as rotas de processo e fatores como volume de produção e tamanho da célula eram considerados (este último considerando a maximização do fluxo total de peças dentro de uma célula); neste estudo, foi utilizado método BT dinâmico e uma estrutura de memória em longo prazo, sendo que a melhor solução inicial gerada (para problemas pequenos e médios) ocorreu através do método de agrupamento e atribuição.

O trabalho de Selim *et al* (1998) mostra estudos compreensivos na revisão dos problemas de formação de células. Kusiak and Cho (1992) apresentou o chamado *coeficiente de similaridade*, que mensura a similaridade de peças baseado em rotas existentes para essas peças dentro de um processo. Ou seja, estes autores apresentaram estudos do *coeficiente de similaridade* baseado em fluxo de peças. Won e Kim (1997) empregaram este coeficiente com um critério de agrupamento com o menor transporte intracelular possível, e baseado em máquinas. De acordo com Wu *et al* (2009), como o número de tipos de peças é normalmente bem maior que o número de máquinas, o coeficiente de similaridade baseado em peças sofre em termos de volume de dados computacionais. Portanto, no estudo de Wu *et al* (2009), foi utilizado o coeficiente de similaridade baseado em máquinas, conforme Won e Kim (1997). Wu *et al* (2010) discutem sobre coeficientes de similaridade para estipular o uso em um processo de formação de células (CF). Através de aproximações gráficas, o processo para formar células de manufatura é iniciado ao se coletar dados de problemas e com a conversão destes em representações gráficas ponderadas. Lei & Wu (2005) publicaram um estudo onde a solução inicial é gerada a partir do coeficiente de similaridade, gerando um algoritmo rápido e efetivo, através da busca tabu (BT).

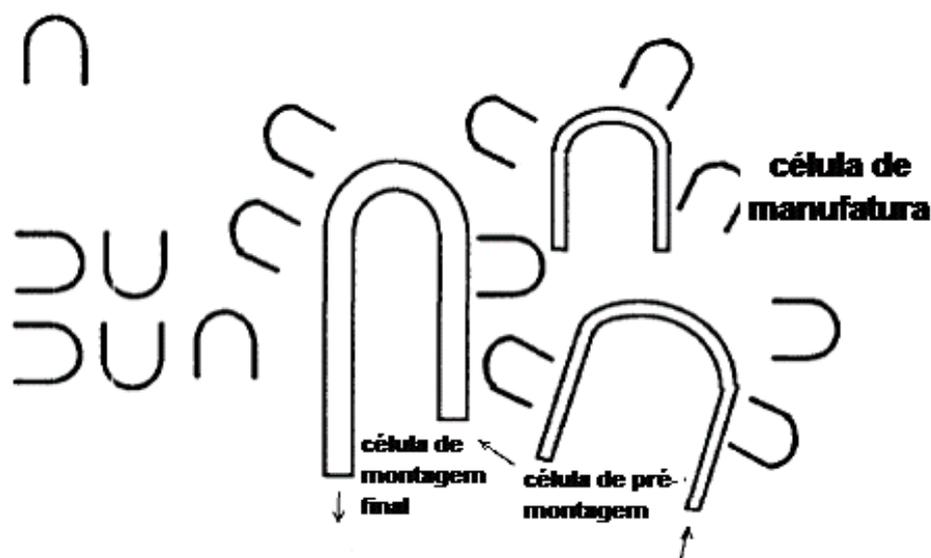


Figura 6 - Estrutura principal de um Sistema de Manufatura Celular (CMS) (traduzido de WANG et al, 1998).

Conforme Papaioannou e Wilson (2010), é importante também mencionar que a tomada de decisão em uma manufatura, muitas vezes, envolve incertezas e ambiguidade de fatores, ficando difícil especificar objetivos e restrições por conta da dificuldade nas estimativas a serem feitas. Desta forma, em problemas de formação de células em que ocorre esta situação, alguns pesquisadores trabalharam com a lógica *fuzzy*, que pode envolver, no agrupamento, métodos exatos, como a programação matemática, métodos heurísticos e redes neurais. Da mesma forma, o estudo proposto por Safaei *et al* (2008) menciona um horizonte de planejamento multiperíodo, onde há variações no *mix* de produto e na demanda de cada produto por período e, como é esperado, verifica-se que o resultado ideal encontrado em um período não é necessariamente bom em outro. Esta flutuação na demanda e também na disponibilidade dos recursos na manufatura (ou seja, máquinas) a cada período, são ambos considerados fatores difusos (ou *fuzzy*) que geram coeficientes na função objetivo e, conseqüentemente, na matriz gerada para a resolução do problema. O objetivo deste estudo é de dar uma configuração celular ótima a cada período.

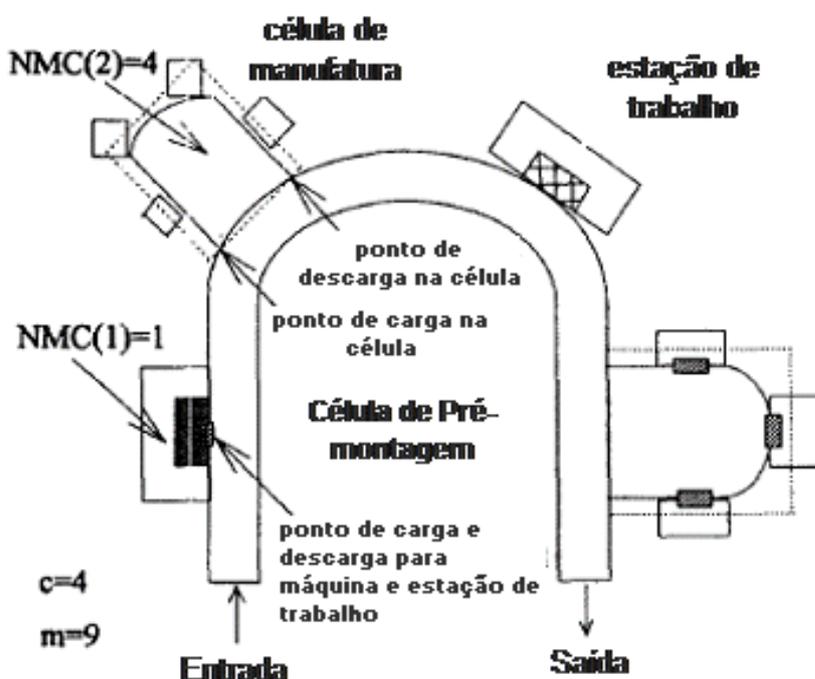


Figura 7 - Exemplo de *layout* de CMS (traduzido de WANG et al., 1998).

Ainda dentro da lógica *fuzzy*, o estudo feito por Papaioannou e Wilson (2008) considera agrupamentos de peças a máquinas e de máquinas em células de manufatura, levando em conta a sequência de operação em cada máquina, carga de utilização peça/máquina, custos de *setup* e máquinas do mesmo tipo. A função objetivo visa minimizar o movimento intercelular e custos de *setup* ao agrupar as máquinas em células. As soluções encontradas foram ilustradas através de exemplos (com discussões sobre os tempos de processamento encontrados e qualidades de soluções) com o intuito de ilustrar o comportamento do modelo encontrado, comparando o desempenho dos modelos difusos (*fuzzy*), devido à presença de incertezas, com os modelos determinísticos à medida que a escala ou tamanho do problema aumenta. A vantagem do uso do método difuso (*fuzzy*) sobre o determinístico é ressaltada em casos de problemas de grande escala (vide quadro 01 que utiliza 2 exemplos, que são o DS3 e DS4, onde facilmente se verificar que o método difuso emprega menor tempo computacional se comparado aos modelos determinísticos) e quando a tolerância das restrições se torna maior. No entanto, o modelo difuso (*fuzzy*) é bastante simples quando aplicado um operador apropriado, em termos de resultados de agrupamento e tempo computacional.

É interessante notar, também, que, conforme Xambre e Vilarinho (2003), na maior parte dos modelos de formação de células (CF), a máquina onde cada operação irá ocorrer já está previamente definida. Entretanto, em grande parte dos problemas reais, isso não ocorre, havendo aí, então, a possibilidade de que a alocação da operação possa ser mais um grau de liberdade em um problema de formação de células (CF).

2.3 Layout

Kouvelis *et al*(1992) defende que o tema *layout* fabril é um dos mais importantes a serem resolvidos nos primeiros estágios do planejamento fabril. Conforme Thompkins & White (1984), de 20 a 50% dos custos operacionais de uma manufatura podem ser decorrente de custos com *layout* ou manuseio de materiais. Ainda conforme Thompkins *et al* (1996), um planejamento correto de *layout* e também de fluxo de materiais pode reduzir de 10 a 30% os custos operacionais em uma manufatura.

Tabela 1- Resultados computacionais para Modelos Difusos (Fuzzy) e Determinísticos (traduzido de PAPAIOANNOU e WILSON, 2008).

<i>Conjunto de dados 3: Tamanho do Problema (M x P) = (7 x 10); NCELLS = 5; EMAX=8; P0=21; PR1=2; PR2=5; Exemplos de máquina=27</i>							
Caso	Operador	Função de Pertinência	Células Criadas	Células distintas utilizadas para cada peça	Revisitas posteriores de peças	Tempo de CPU	Custo total
1	"Min"	Linear não-crescente	-	-	-	>57 horas	-
2	"Min"	Triangular	3	12	1	6966	246,91
3	"e"	Linear não-crescente	-	-	-	>57 horas	-
4	"e"	Triangular	3	12	1	10431	246,91
5	"Soma mínima delimitada"	Linear não-crescente	3	13	1	17	251,49
6	"Soma mínima delimitada"	Triangular	3	13	1	51	267,75
7	Modelo Determinístico	EMAX=8	4	12	2	16499	264,14
8	Modelo Determinístico	EMAX=8	3	12	0	8341	262,14
<i>Conjunto de dados 3: Tamanho do Problema (M x P) = (9 X 9); NCELLS = 5; EMAX=8; P0=35; PR1=4; PR2=5; Exemplos de máquina=28</i>							
Caso	Operador	Função de Pertinência	Células Criadas	Células distintas utilizadas para cada peça	Revisitas posteriores de peças	Tempo de CPU	Custo total
1	"Min"	Linear não-crescente	-	-	-	>50 horas	-
2	"Min"	Triangular	3	11	0	25448	243,27
3	"e"	Linear não-crescente	4	11	1	114171	249,69
4	"e"	Triangular	3	11	0	2752	243,27
5	"Soma mínima delimitada"	Linear não-crescente	3	12	5	9	263,69
6	"Soma mínima delimitada"	Triangular	3	12	5	49	263,69
7	Modelo Determinístico	EMAX=8	4	11	1	48974	257,99
8	Modelo Determinístico	EMAX=12	3	10	1	688	245,11

Conforme Drira *et al* (2007), o *layout* de instalações chega a impactar no desempenho do sistema de produção. Inclusive, neste estudo, é feita uma revisão na literatura, sobre abordagens aplicadas ao problema de *layout*, considerando-se vários aspectos comumente encontrados na manufatura, como por exemplo, formato de máquina (regular/irregular).

Conforme a Tabela 02, em um processo de produção em massa, o *layout* pode estar entre o arranjo físico celular e o arranjo por produto. No primeiro arranjo, normalmente, o arranjo físico é em "U", e se organiza em células, que são independentes entre si. Já o arranjo por produto visa mais a formação de linhas, baseado em um volume maior de produção, algo determinante para a formação das linhas de produto (p. ex., diferenciando produtos de alto volume e de baixo volume de produção).

Conforme Xambre e Vilarinho (2003), no arranjo físico orientado por processo, máquinas idênticas são agrupadas, obtendo-se assim muita flexibilidade, mas com alto estoque em processo (ou, no jargão fabril, WIP) e grande movimentação de materiais. Por outro lado, no arranjo orientado por produto, as máquinas são posicionadas em linhas de produção dedicadas à manufatura de produtos específicos, havendo então baixos custos e WIP, além de se operar com altos volumes de produção e bom controle de produção, mas com baixa flexibilidade. Ainda conforme Burbidge (1992), a manufatura celular visa cobrir algumas “lacunas” deixadas pelos arranjos físicos orientados por processo e por produto de arranjo físico de maquinário, tendo as vantagens dos sistemas orientados por produto e processo com uma alta variedade e médio *mix* de produção.

Uma questão importante a ser tratada é que o estudo do melhor *layout* fabril parece passar por “modismos”, não levando sempre em conta o que é melhor e mais benéfico para o problema em questão, tentando-se apenas copiar o que outros fizeram. Na empresa estudada neste trabalho, o *layout* migrou, no ano de 2003, do Arranjo Orientado por Processos para o Arranjo por Produtos, baseado, em parte, no aumento do volume histórico de produção.

Tabela 2 - Características do arranjo físico, conforme o produto e processo produtivo (traduzido de BLACK, 1990).

PRODUTO		PROCESSO DE MANUFATURA	ARRANJO FÍSICO
Volume	Variedade		
↓	baixo	por Projeto	Posicional
	·	<i>Jobbing</i>	
	médio	Batelada (Lotes)	por Processo
	·	em Massa	Celular
	alto	baixa	Contínuo

Nos dias atuais, as empresas, constantemente, dispendem tempo em exaustivas reuniões para alterações de *layout*, que podem gerar paradas de linha e altos custos para as organizações. Em muitos casos, estas mudanças são feitas sem um embasamento técnico (por ex., gerando arranjo por processo ou por produto), sem a necessária comprovação de que o objetivo (por ex., menor caminho a ser percorrido pela peça, menor estoque de itens WIP *etc.*) será alcançado. Na

verdade, muitas alterações tentam “copiar” outras empresas, sem a necessária compreensão das implicações das alterações propostas. Dentro do *lean thinking* (manufatura enxuta) e do sistema toyota de produção (alguns autores consideram estes temas idênticos), um dos pilares de sustentação é a busca incessante pelo desperdício zero. Segundo Ohno (1988), os sete desperdícios são:

- Excesso de produção: Produzir mais e antes que o necessário;
- Inventário: Peças semi-acabadas entre operações;
- Transporte: Movimento de peças.
- Processos Desnecessários: Passos não necessários no processo.
- Má Qualidade: Peças que necessitam retrabalho ou são sucata.
- Manuseio: Movimentos desnecessários do trabalhador.
- Espera: Trabalhadores esperando por máquinas ou peças.

O estudo de otimização do arranjo físico ou *layout* através do uso de Métodos Heurísticos, embora não seja um tema, normalmente, conhecido em empresas no Brasil, é bastante importante, pois tem o potencial de reduzir cada um dos sete desperdícios (OHNO, 1988), principalmente, os desperdícios no transporte e no manuseio.

No entanto, há estudos bem focados neste tema, como o proposto por Chiang & Lee (2004), que propõe a formação de células juntamente com um *layout* formado por uma sequência de locação de máquinas linear bidirecional, focando a minimização do movimento intercelular (respeitando as restrições impostas). O estudo proposto por Chwif *et al* (1998) propôs aproximações aplicáveis para instalações fabris, considerando áreas, formas e orientações, bem como para problemas de *layout* de máquinas, considerando-se pontos de entrada e saída nas máquinas.

Já Wang *et al* (2001) propôs uma solução do problema de *layout* através da manufatura celular, visando a minimização dos custos gerados pelos movimentos intercelulares e intracelulares de peças. Por outro lado, Wang *et al* (2005) realizou um estudo de *layout* considerando também aspectos como fatores de formato e utilização de áreas fabris, já que usualmente em estudos de *layout*, seriam considerados apenas fatores como custos com manuseio de materiais.

Por outro lado, Ou-Yang e Utamima (2013), Kothari e Ghosh (2013) e Amaral (2006), propuseram estudos para a solução de *layout* em linha reta (coluna simples), que podem produzir soluções muito boas em um tempo computacional razoável. No

estudo de Ou-Yang e Utamima (2013), a função objetivo é a minimização da soma ou da soma ponderada das distâncias entre os chamados pares de instalações (chamado “*facilities pairs*”); isso é feito no estudo de Ou-Yang e Utamima (2013) através do uso de algoritmos meta-heurísticos híbridos, e no estudo de Kothari e Ghosh (2013), através de BT. A grande vantagem do estudo de Ou-Yang e Utamima (2013) é que são considerados vários aspectos, como restrições de fluxo, tamanho, espaço livre, custos de instalações e aspectos de segurança. Já no estudo de Kothari e Ghosh (2013), são utilizadas técnicas para aceleração a busca na vizinhança.

Houve ainda estudos como proposto por Aiello *et al* (2013) e Garcia-Hernandez *et al* (2013) e Konak (2012), que consideram áreas irregulares de máquinas ou de *layout*, o que sugere uma aproximação do que se pode encontrar talvez na maioria das empresas, tornando o problema “real”. No estudo de Aiello *et al* (2013), foi aplicado o AG Multiobjetivo, que se demonstrou como um método bastante promissor para este tipo de problema; já no estudo de Garcia-Hernandez *et al* (2013), houve a aplicação de um AG interativo (permite interações do algoritmo com o DM ou Tomador de Decisão, que compara a nova solução gerada com a melhor solução gerada). É interessante notar, ainda conforme Aiello *et al* (2013), que o método multiobjetivo realiza uma ordenação (aplicando o Princípio de Pareto) com as melhores soluções. Ao invés de gerar apenas uma solução, sendo assim mais realístico. Já o estudo proposto por Chiang & Chinag (1998) realizou um estudo utilizando-se de vários métodos de otimização conhecidos (como por exemplo, BT e SA), bem como a combinações destes métodos, de forma a aproveitar as vantagens de cada um.

Tendo em vista aspectos de mudança rápida de demandas, *mix* de produtos nas linhas de produção, bem como troca de linhas de produtos ao longo do tempo, Yu-Hsin Chen (2013), considerou um problema de instalações de *layout* dinâmico. Este é um problema de otimização combinatória (utilizando redes meta-heurísticas), onde pode haver a troca e armazenamento de possíveis soluções, buscando assim encontrar soluções globais e locais. O estudo proposto por Georgiadis *et al* (1999) propôs um modelo matemático para alocar máquinas em espaços bi- ou tridimensionais. Já no estudo proposto por Amaral (2009), propôs um estudo de *layout* em linha reta (coluna simples), visando a minimização da soma acumulada das distâncias entre os postos de trabalho.

3 METODOLOGIA

O presente estudo, em função do apresentado no parágrafo anterior, será iniciado com a formação de duas células de manufatura, utilizando BT Híbrida para alocação de máquinas às células e, posteriormente, *BB* (alocação de roteiros às peças), com poda através da busca em profundidade, para melhorar os tempos computacionais. Na segunda parte do problema, que é a formação do *Layout*, a BT Híbrida mostrou-se relativamente eficiente, pois conseguiu soluções de boa qualidade com um tempo computacional relativamente baixo (vide Weller, 2008); por outro lado, é verdade que os métodos utilizados na primeira parte do problema visaram diminuir a “carga” computacional atribuída à segunda parte, pois procuraram resolver parte do problema do *Layout*, através da formação de duas células de manufatura, e conseqüente divisão do espaço fabril disponível.

Ambas as simulações foram feitas utilizando o *Visual Basic 2008 Express Edition*, e foram feitas em um computador com processador Pentium Core i3-2330M (Processador de 2,2 Ghz, 3MB de memória cachê, 4GB de memória RAM).

3.1 Formação de Células

Em Weller (2008), foram propostas duas abordagens híbridas para a formação de células usando SA e BT. Cada uma das duas abordagens é apresentada na sequência, indicando-se em que elas se diferenciam da literatura.

3.1.1 Abordagem de *Simulated Annealing* (SA) Híbrido

O *Simulated Annealing* (SA) usado na figura 8 (traduzido de Caux *et al*, 2000) se assemelha ao descrito em Wang e Wu, 2001, com a diferença que na figura 8, o algoritmo apresentado possui adicionalmente a variável s^{best} , que é a melhor solução encontrada até a iteração em determinado momento; além disso, pode-se perceber que há uma diferença com a BT Híbrida, que faz o teste de todas as soluções vizinhas, enquanto no SA é gerada apenas uma solução vizinha a cada iteração.

```

1: Definir os parâmetros temperatura inicial ( $C$ ) e probabilidade do movimento de troca ( $pp$ );
2: Gerar solução inicial  $s$ ;
3: while Não atingiu o critério de parada do
4:   Gerar solução  $s'$  vizinha da solução vigente  $s$ ;
5:    $\Delta = Z(s') - Z(s)$ ;
6:   if  $\Delta \leq 0$  then
7:     Nova solução vigente:  $s = s'$ ;
8:     if  $Z(s) < Z(s^{best})$  then
9:        $s^{best} = s$ 
10:    end if
11:  else
12:    Gerar um número aleatório  $r$ , onde  $0 < r < 1$ ;
13:    if  $r < \exp(-\Delta/T)$  then
14:      Nova solução vigente:  $s = s'$ ;
15:    end if
16:  end if
17:  Atualizar a temperatura  $T$ ;
18: end while
19: Escrever a melhor solução encontrada  $s^{best}$ .

```

Figura 8 - Algoritmo SA Híbrido traduzido de Caux et al (2000).

O SA usado é dito híbrido porque o SA apenas realiza a alocação de máquinas às células. Posteriormente, o *Branch-and-Bound* (BB) será utilizado na segunda parte do problema de formação de células. O SA, descrito na figura 1, é bastante simples. Considera-se que há dois movimentos possíveis, onde o *movimento de inserção* corresponde à mudança da célula onde uma máquina está alocada e o *movimento de troca* corresponde à troca de posição entre duas máquinas (a primeira máquina vai para a célula onde está a segunda máquina e vice-versa). Assim, no passo 1 do algoritmo são definidos os parâmetros iniciais: definição da temperatura inicial (que influencia a chance de aceitação de uma solução pior que a vigente) e a chance (probabilidade de ocorrência) do movimento de troca.

Na linha 2 do algoritmo, gera-se a solução inicial s , que corresponde ao sorteio de uma célula para a alocação de cada máquina, respeitando-se o limite de máquinas por célula de manufatura. Se o limite máximo de máquinas é atingido em uma célula, ela não poderá ser escolhida a partir dali. A geração da solução inicial s é finalizada com a execução do BB, descrito na seção 3.1.2, que definirá o custo da

solução inicial gerada ao identificar o melhor roteiro de fabricação de cada peça (respeitando a capacidade de produção).

Na terceira linha do algoritmo, ele prossegue até que o critério de parada seja atingido. O critério de parada pode ser tanto o número total de iterações como o número de iterações sem alterar a melhor solução encontrada s^{best} .

Na sequência do algoritmo, gera-se a uma nova solução s' . Se a solução gerada for melhor que a solução vigente ($\Delta \leq 0$, para um problema de minimização), a solução gerada se torna a nova solução vigente, como indicado na linha 7. Neste caso, testa-se na linha 8 se a solução gerada é a melhor solução encontrada.

Caso a solução gerada seja pior que a solução vigente ($\Delta > 0$), gera-se um número aleatório r , na linha 12 do algoritmo. Nas linhas 13 a 15, se $r < \exp(-\Delta/T)$, a solução gerada s' se tornará a nova solução vigente s . Na sequência, a “temperatura” do SA é atualizada, aplicando-se o mesmo critério proposto por Caux *et al* (2000):

$$T = \ln(n+1) \tag{1}$$

3.1.2 **Branch-and-Bound (BB)**

Este método é, provavelmente, a principal contribuição dessa dissertação, tendo inclusive sido publicado em Rodrigues *et al.* (2013). Ele é utilizado quando ocorre uma alocação de máquinas às células e assim uma solução vizinha s' é criada. Esta é uma etapa da BT híbrida, utilizada no presente estudo, visando alocar roteiros a peças, respeitando as restrições impostas (restrições de capacidade de cada máquina e intuito de minimizar sempre operações extracelulares). No entanto, caso o BB não seja adequadamente inserido, ele pode gerar um tempo computacional alto.

A fim de executar atribuições de roteiros às peças, são propostos dois algoritmos que trabalham conjuntamente. O algoritmo da figura 9 é responsável pela preparação das informações necessárias ao BB, enquanto o algoritmo da figura 10 realiza o BB propriamente dito.

- **Algoritmo de preparação do BB:**

Uma vez que as máquinas foram alocadas nas células, na primeira etapa do SA ou da BT, na linha 3 do algoritmo da figura 9 identifica-se o número de operações extracelulares para cada roteiro r da peça p , já que a sua alocação é testada em cada célula c . Assim, identifica-se a célula onde ocorre o menor número de operações extracelulares para o roteiro r e registra-se o número de operações extracelulares em $ECmin_r$.

Peças com apenas um roteiro possível não são incluídas na lista $EXPAND_n$, já que a atribuição ótima do roteiro à peça já é conhecido. O objetivo da lista $EXPAND_n$ é favorecer a busca em profundidade no BB.

Na linha 6, para qualquer peça p com apenas um roteiro r disponível, atualizar a capacidade disponível de qualquer máquina m associada ao roteiro r (onde $m \in MR_r$). Isto é realizado, na máquina m , subtraindo-se da capacidade disponível Cap_m o tempo de processamento total do roteiro, levando em consideração a demanda da peça p .

Depois que todos os roteiros associados à peça p foram analisados, na linha 11, para cada peça p , identifique a diferença entre os dois menores $ECmin_r$, indicando-a como $diff_p$.

Na linha 8, as peças são ordenadas numa lista chamada $EXPAND_n$, onde n é o nó raiz do BB, em ordem decrescente de $diff_p$.

Na linha 12, some o menor $ECmin_r$ (para cada peça p). Esta soma é usada como a solução relaxada ou limite inferior do nó raiz do BB.

Na linha 14, aplicar o procedimento de busca do BB (algoritmo da figura 12). Este procedimento foi chamado de Algoritmo 5.

```

1: for  $p = 1$  to  $P$  do
2:   for  $r \in RP_p$  do
3:     Identificar a célula onde ocorre o menor número de operações
       extra-celulares  $ECmin_r$  para cada roteiro  $r$  da peça  $p$ ;
4:     if  $p$  com apenas um roteiro disponível then
5:       for  $m \in MR_r$  do
6:         Atualizar a capacidade disponível  $Cap_m$ ;
7:       end for
8:     end if
9:   end for
10:   $diff_p =$  diferença entre os dois menores  $ECmin_r$ ;
11:  Limite inferior do BB = Limite inferior do BB + menor  $ECmin_r$ ;
12:  Ordenar peças, em ordem decrescente de  $diff_p$ , numa lista chamada
        $EXPAND_n$ , onde  $n$  é o nó raiz do BB;
13: end for
14: Aplicar o Algoritmo 5 (executar o BB)

```

Figura 9 - Algoritmo de Preparação do BB.

- **Algoritmo de busca do BB:**

No nó raiz, a lista $EXPAND_0$ contém todas as peças exceto aquelas com apenas um roteiro disponível. A lista $EXPAND_n$ é atualizada a medida que novos nós são gerados na árvore de busca, através da alocação de um roteiro de fabricação a uma peça. Cada nó n tem sua própria lista $EXPAND_n$. Na medida em que a busca evolui, através da geração de novos nós, o procedimento de BB mantém uma lista dos nós abertos, chamada $OPEN$. Nós abertos correspondem a soluções parciais, quando a lista $EXPAND_n$ ainda não está vazia. Inicialmente, quando a lista $OPEN$ é composta apenas do nó raiz, o limite inferior do procedimento de BB é calculado através do algoritmo de preparação do BB, dado na figura 9, e o limite superior da busca (valor da melhor solução completa factível encontrada) é definido como $+\infty$. Em problemas de minimização, os nós abertos na lista $OPEN$ são colocados em ordem crescente do valor da função objetivo (ou função de avaliação), tal que o primeiro nó nesta lista corresponde à solução parcial avaliada como a mais promissora. O leitor encontrará uma explicação detalhada da lista $OPEN$ e do BB em Pearl (1984). O procedimento de busca do BB (Pearl, 1984) é resumido no algoritmo da figura 10.

Na linha 1 do algoritmo de busca do BB, define-se que o limite superior da função objetivo, chamado $limsup$, é igual a $+\infty$. Neste algoritmo, são eliminados da lista $OPEN$ todos os nós cujo valor da solução é maior ou igual a $limsup$.

Na linha 2 do algoritmo, verifica-se se a lista *OPEN* está vazia. A busca do BB é encerrada quando a lista *OPEN* estiver vazia; caso contrário, ramificar o (fazer o “*branching*” do) primeiro nó da lista *OPEN*.

Na linha 7, remover o primeiro nó da lista *OPEN* e na lista $EXPAND_{No_{filho}}$ elimine a primeira peça da lista $EXPAND_{No_{pai}}$, mantendo as demais peças.

A cada novo nó gerado, na linha 9, atualizar a capacidade disponível das máquinas, identificando-a como $Cap_{m, No_{filho}}$.

Na linha 12, o valor estimado da função objetivo do No_{filho} é dado pela soma do valor da solução estimado para No_{pai} mais Δ , onde Δ é a diferença entre o $EC_{min,r}$ para o roteiro r e a menor $EC_{min,r}$ para a peça p , como indicado na linha 11. Note que sempre $\Delta \geq 0$.

Na linha 13 é testado se a lista $EXPAND_n$ não está vazia. Se foi criado um nó aberto (solução parcial) factível, na linha 14, ele é inserido e ordenado na lista *OPEN*. Se a lista $EXPAND_n$ está vazia, novas soluções completas (nós fechados) factíveis foram identificadas.

No teste da linha 16, se a nova solução completa é factível e o seu valor da função objetivo ($Z(No_{filho})$) é menor do que o limite superior da função objetivo, chamado *limsup*, então, na linha 17, *limsup* é atualizado para o valor da função objetivo desta solução ($Z(No_{filho})$).

Na linha 18, é realizado o procedimento de *fathoming*, ou seja, eliminar qualquer nó aberto (da lista *OPEN*) cujo valor estimado da função objetivo seja maior ou igual ao limite superior.

A maior contribuição do BB apresentado está em propor a utilização da lista $EXPAND_n$. Pretende-se fomentar, tanto quanto possível, uma busca em profundidade. Portanto, parece haver uma poda significativa do espaço de busca do BB, como indicado nos tempos de processamento da abordagem, apresentados nos resultados.

```

1:  $limsup = +\infty$ ;
2: while Lista OPEN não é um conjunto vazio do
3:    $No_{pai}$  = primeiro nó da lista OPEN;
4:    $p1$  = primeira peça na lista  $EXPAND_{No_{pai}}$ ;
5:   for  $r \in RP_{p1}$  do
6:     Gerar um novo nó  $No_{filho}$ ;
7:     Remover o primeiro nó da lista OPEN e a primeira peça na lista
        $EXPAND_{No_{filho}}$ ;
8:     for  $m \in MR_r$  do
9:       Atualizar a capacidade disponível  $Cap_{m, No_{filho}}$ ;
10:      if  $No_{filho}$  factível then
11:         $\Delta$  = diferença entre o  $ECmin_r$  para o roteiro  $r$  e a menor
           $ECmin_r$  para a peça  $p1$ ;
12:         $Z(No_{filho}) = Z(No_{pai}) + \Delta$ ;
13:        if lista  $EXPAND_{No_{filho}}$  não está vazia then
14:          Inserir e ordenar  $No_{filho}$  na lista OPEN em ordem cres-
            cente do valor estimado da função objetivo;
15:        end if
16:        if lista  $EXPAND_{No_{filho}}$  vazia AND  $limsup > Z(No_{filho})$ 
          then
17:           $limsup = Z(No_{filho})$ ;
18:          Realizar o procedimento de fathoming;
19:        end if
20:      end if
21:    end for
22:  end for
23: end while
24: Retornar o valor da solução encontrada

```

Figura 10 - Procedimento de busca do BB (Algoritmo 5).

3.1.3 Abordagem de Busca Tabu (BT) Híbrida

A figura 11 ilustra o pseudo-algoritmo da Busca Tabu que é utilizada neste estudo, em uma das etapas da formação de células. A variável s^{best} identifica a melhor solução encontrada.

Conforme linha 2, a solução inicial é gerada de forma aleatória; a solução inicial s é a s^{best} , caso $Z(s')$ menor que $Z(s^{best})$.

O critério de diversificação mencionado na linha 7 equivale à quantidade de iterações sem encontrar um novo s^{best} . A solução vigente é s , enquanto a melhor solução vizinha é s' .

A busca na vizinhança, conforme linha 8, está melhor explicada na figura 12.

A cada iteração, conforme linhas 9, 10 e 11, é verificado se a melhor solução gerada s' supera s^{best} .

Após isso, na linha 12, a nova solução vigente s é a melhor solução s' ; na Linha 15, é exibida a melhor solução encontrada, que é s^{best} .

```

1:  $s^{best} = +\infty$ 
2: while não satisfaz o critério de parada do
3:   Gerar solução inicial ( $s$ )
4:   if  $Z(s) < Z(s^{best})$  then
5:      $s^{best} = s$ 
6:   end if
7:   while não satisfaz o critério de diversificação do
8:     Executar o Algoritmo 2 (gerar soluções vizinhas de  $s$ ) e selecionar
       o melhor vizinho ( $s'$ )
9:     if  $Z(s') < Z(s^{best})$  then
10:       $s^{best} = s'$ 
11:    end if
12:    Atualizar a solução vigente:  $s = s'$ 
13:  end while
14: end while
15: Escrever  $s^{best}$ 

```

Figura 11 - Algoritmo de Busca Tabu Implementado.

É importante mencionar que no presente estudo, a fim de propiciar uma maior eficiência na BT em uma etapa componente da formação de células deste estudo, eliminaram-se soluções vizinhas não promissoras através da poda no espaço de busca. Neste estudo foram realizadas três restrições: a quantidade mínima de máquinas por célula, a quantidade máxima de máquinas por célula e o movimento de troca entre máquinas que estão em células diferentes, que ocorre só quando as máquinas são relacionadas, ou seja, quando há pelo menos um roteiro de fabricação onde estas máquinas estão presentes – este procedimento de restrição economiza em tempo de processamento.

A figura 12 explica como funciona a busca na vizinhança da BT. A linha 04 avalia se o movimento de inserção da máquina m na célula c satisfaz as três restrições; caso sim, se executa outro algoritmo, conforme figura 13, que avalia a solução encontrada. Na linha 1, ocorre a execução do algoritmo 4, que é o BB (explorado e apresentado na próxima seção).

```

1:  $Z(s') = + \infty$ ;
2: for  $m = 1$  to  $M$  do
3:   for  $c = 1$  to  $C$  do
4:     if movimento de inserção na célula  $c$  satisfaz restrições then
5:       Executar o Algoritmo 3 (Avaliação da Solução);
6:     end if
7:   end for
8:   for  $m' = 1$  to  $M$  do
9:     if movimento de troca de células (entre as máquinas  $m$  e  $m'$ )
       satisfaz restrições then
10:      Executar o Algoritmo 3 (Avaliação da Solução);
11:    end if
12:  end for
13: end for

```

Figura 12 - Algoritmo de busca na vizinhança conforme algoritmo de Busca Tabu Implementado – esta figura é chamada de Algoritmo 2

Após isso, na linha 2, é verificado se o movimento da máquina m (ou m' para movimento de troca), que faz parte da solução vizinha, se é tabu. Caso o movimento da máquina seja tabu, aplica-se o critério de aspiração a partir da linha 3 e, caso o movimento seja melhor que s^{best} (para um problema de minimização), assume-se que s' é s^{best} (vide também linhas 4 a 7). Neste caso, conforme linha 8, a máquina m ou m' se torna um movimento tabu. Caso contrário, na linha 10, se o movimento da máquina m ou m' não for tabu, é verificado se a solução vizinha gerada é melhor que a solução vizinha atual; caso sim, conforme linha 11 assume-se que a solução vizinha s' é a melhor solução vizinha achada até o momento (vide também linhas 12 e 13). Finalmente, na linha 14, assume-se que m ou m' se torne um movimento tabu.

Uma questão importante entre os métodos com apenas um objetivo e os métodos com dois ou mais objetivos para serem otimizados é que este último se torna um conjunto de otimização de pareto, segundo Lei & Wu (2005).

Conforme Lei & Wu (2005), comparando-se a busca tabu multiobjetivo (MOTS) com o simulated annealing paralelo (PSA), é verificado que o primeiro apresenta melhores resultados no agrupamento de peças e no desempenho em problemas de formação de células multiobjetivos. Ou seja, o algoritmo utilizado na Busca Tabu Multiobjetivo (MOTS) é promissor em problemas multiobjetivos de formação de células.

```

1: Executar o Algoritmo 4;
2: if  $m$  (ou  $m'$ , se for um movimento de troca)  $\in$  lista tabu then
3:   if  $Z(s^{best}) > Z(\text{solução vizinha})$  then
4:      $s^{best} = \text{solução vizinha}$ ;
5:      $Z(s^{best}) = Z(\text{solução vizinha})$ ;
6:      $s' = \text{solução vizinha}$ ;
7:      $Z(s') = Z(\text{solução vizinha})$ ;
8:     Inclui  $m$  (e  $m'$ , se for um movimento de troca) na lista tabu;
9:   end if
10: else
11:   if  $Z(s') > Z(\text{solução vizinha})$  then
12:      $s' = \text{solução vizinha}$ ;
13:      $Z(s') = Z(\text{solução vizinha})$ ;
14:     Inclui  $m$  (e  $m'$ , se for um movimento de troca) na lista tabu;
15:   end if
16: end if

```

Figura 13 - Algoritmo de Avaliação da solução, conforme busca na vizinhança para algoritmo de Busca Tabu Implementado – esta figura é chamada de Algoritmo 3.

3.2 Layout de Fábrica

Uma vez solucionado o problema de formação de célula, que é responsável por definir a melhor alocação de peças às máquinas, o próximo passo é resolver o problema de *layout* aproveitando a alocação de peças às máquinas, que foi obtida na solução do problema de formação de células. Nesta seção são apresentadas as duas abordagens originais propostas para a solução do problema de *layout*. A primeira abordagem usa programação matemática para solucionar o problema. A segunda abordagem combina busca tabu com programação matemática.

Drira *et al.* (2007) apresentam uma ampla revisão da literatura sobre as diferentes abordagens aplicadas para a solução deste problema.

3.2.1 Proposta de uma Abordagem de Programação Matemática

Nesta seção, é proposta uma abordagem original para a solução do problema de *layout*. Esta abordagem é baseada em Programação Linear Inteira Mista (PLIM). Para facilitar a compreensão desta seção, a tabela 3 apresenta a relação dos parâmetros usados e a tabela 4 apresenta as variáveis usadas.

A equação 2 representa a função objetivo, que expressa a ponderação de dois objetivos. O principal objetivo (com peso 10) é minimizar o volume transportado, que

será o produto do volume de peças $dem_{m,mm}$ pela distância percorrida ($dist_X_{m,mm} + dist_Y_{m,mm}$), representando o deslocamento ao longo dos corredores da área. Um objetivo secundário é agrupar todos os postos na menor área possível, minimizando a soma das dimensões ocupadas ao longo dos eixos x e y , respectivamente, $maior_X$ e $maior_Y$. O motivo pelo qual, na mencionada equação 2, o volume de peças transportadas possuirá um peso 10, enquanto as dimensões das máquinas em x e y possuirão peso 1, reside no fato de que a redução no volume de peças a serem movimentadas ao longo da linha de produção poderá gerar uma economia imediata (que é no momento a política da empresa), ao contrário de uma possível redução na área ocupada pelas máquinas na manufatura.

Tabela 3 - Parâmetros usados na modelagem.

$altura_m$	altura (amplitude ao longo do eixo y) ocupada pelo posto de trabalho m
$dem_{m,mm}$	volume de peças deslocado de um posto m para o posto mm
$largura_m$	largura (amplitude ao longo do eixo x) ocupada pelo posto de trabalho m

Tabela 4 - Variáveis usadas na modelagem.

$alt_{l,c}$	altura (amplitude ao longo do eixo y) ocupada pelo posto de trabalho m
$dist_X_{m,mm}$	distância a ser percorrida entre os postos m e mm ao longo do eixo x
$dist_Y_{m,mm}$	distância a ser percorrida entre os postos m e mm ao longo do eixo y
$distanc_X_{l,c,ll,cc}$	distância a ser percorrida entre as posições da coluna c da linha l e da coluna cc da linha ll ao longo do eixo x
$distanc_Y_{l,c,ll,cc}$	distância a ser percorrida entre as posições da coluna c da linha l e da coluna cc da linha ll ao longo do eixo y
$larg_{l,c}$	largura (amplitude ao longo do eixo x) ocupada pelo posto de trabalho m
$maior_X$	dimensão da área ocupada ao longo do eixo x
$maior_Y$	dimensão da área ocupada ao longo do eixo y
$pc_{m,l,c}$	variável binária que será um (1) se o posto m está localizado na posição c da linha l . Caso contrário, o valor será nulo
$sobre_X_{l,c,ll,cc}$	variável binária que será um (1) se, no eixo x , há sobreposição entre as posições da coluna c da linha l e da coluna cc da linha ll . Caso contrário, o valor será nulo
$total$	valor da função objetivo

Para o problema de *layout*, as restrições deste modelo matemático são indicadas nas expressões 3 até 22. A fim de otimizar a disposição do *layout* dos postos de trabalho, o espaço é representado por um *grid* expresso por linhas l (ao longo do eixo x) e por colunas c (ao longo do eixo y). A variável binária $pc_{m,l,c}$ identifica a posição na coluna c e a linha l em que o posto m será alocado. Assim, a restrição 3 é gerada para cada posto m , impondo que cada posto só pode ser alocado a uma única posição numa linha l e posição c . A restrição 4 é gerada para todas as posições do *grid* (linhas l e colunas c), expressando que no máximo haverá uma máquina (posto de trabalho) alocada por posição. Haverá mais posições disponíveis no *grid* que postos de trabalho, permitindo um ajuste mais adequado dos postos, devido ao fluxo de peças e a dimensão diferenciada de cada posto. Este aspecto é expresso nas restrições 5 e 6, indicando a largura e a altura ocupadas por cada posto m alocado.

$$(\min)total = \sum_m \sum_{mm > m} 10 * dem_{m,mm} * (dist_X_{m,mm} + dist_Y_{m,mm}) + maior_X + maior_Y \quad (2)$$

Sujeito a:

$$\sum_l \sum_c pc_{m,l,c} = 1 \quad \forall m \quad (3)$$

$$\sum_m pc_{m,l,c} \leq 1 \quad \forall l, c \quad (4)$$

$$larg_{l,c} = \sum_m largura_m * pc_{m,l,c} \quad \forall l, c \quad (5)$$

$$alt_{l,c} = \sum_m altura_m * pc_{m,l,c} \quad \forall l, c \quad (6)$$

As restrições 7 e 8 expressam as posições dos postos ao longo do eixo x . Assim, a restrição 7 impõe o sequenciamento das posições das “colunas” ao longo de uma linha l . A variável $pos_X_{l,c}$ identifica a posição do centro de gravidade (ou mediana) do posto alocado na coluna c da linha l . A restrição 7 impõe que a posição $c+1$ ($pos_X_{l,c+1}$) esteja mais afastada da origem que a posição c ($pos_X_{l,c}$). Assim, a posição $pos_X_{l,c+1}$ deve ser maior ou igual ao valor da posição $pos_X_{l,c}$ somado a metade das larguras dos postos que ocupam as posições (colunas) c e $c+1$ na linha l . Ou seja, no caso de haver postos alocados a estas duas posições, a restrição impede que estes postos se sobreponham, já que estão na mesma linha. A restrição 8 só é gerada para a primeira coluna ($c = 1$) de cada linha, impondo que haja um

afastamento mínimo da borda da área ocupada (origem do eixo x) equivalente à metade da largura do posto que ocupar esta posição.

As restrições 9 e 10 definem o valor da distância ao longo do eixo x entre as posições na coluna c da linha l e na coluna cc da linha ll , considerando-se que $c < cc$. A restrição 11 expressa que a distância entre duas posições será a mesma, independentemente da origem e do destino, sendo necessária porque as restrições 9 e 10 só foram geradas para $c < cc$.

As restrições 7 até 11 são expressas em função de posições nas linhas (l) e colunas (c). A restrição 12 é gerada para expressar cada deslocamento de peças do posto m para mm , ou vice-versa, sendo gerada apenas se $dem_{m,mm} > 0$. Note que a distância $dist_{m,mm}$ só será maior ou igual a zero se os dois postos m e mm estão alocados às posições na coluna c da linha l ou na coluna cc da linha ll , respectivamente, ou vice-versa (note que m ou mm podem ser alocados em cada uma das duas posições escrevendo-se quatro variáveis binárias por restrição). Assim, a restrição será gerada também para toda combinação de posições (linhas e colunas).

$$pos_X_{l,c+1} \geq pos_X_{l,c} + \frac{(larg_{l,c+1} + larg_{l,c})}{2} \quad \forall l, c < cont_l \quad (7)$$

$$pos_X_{l,c} \geq \frac{larg_{l,c}}{2} \quad \forall l, c = 1 \quad (8)$$

$$distanc_X_{l,c,ll,cc} \geq pos_X_{ll,cc} - pos_X_{l,c} \quad \forall l, ll, cc, c < cc \quad (9)$$

$$distanc_X_{l,c,ll,cc} \geq pos_X_{l,c} - pos_X_{ll,cc} \quad \forall l, ll, cc, c < cc \quad (10)$$

$$distanc_X_{l,c,ll,cc} = distanc_X_{ll,cc,l,c} \quad \forall l, ll, c, cc \quad (11)$$

$$dist_X_{m,mm} \geq distanc_X_{l,c,ll,cc} - 9999 * (2 - pc_{m,l,c} - pc_{mm,l,c} - pc_{m,ll,cc} - pc_{mm,ll,cc}) \quad (12)$$

$$\forall m, mm / dem_{m,mm} \geq 0, l, c, ll, cc / l \neq ll \vee c \neq cc$$

As restrições 13 até 17 referem-se ao eixo y , sendo que as mesmas explicações feitas, respectivamente, para as restrições 7 a 12 são válidas a estas restrições. A restrição 18 foi proposta para complementar o modelo matemático, impondo que, se há um fluxo de peças entre os postos m e mm (verificado por $dem_{m,mm} > 0$), então as linhas l e ll que alocam os dois postos devem estar uma do lado da outra ou serem a mesma ($l = ll$).

A restrição 18 só é gerada se $l > ll + 1$ ou $ll > l + 1$, indicando que há uma ou mais linhas entre l e ll . Neste caso, se os postos m e mm forem alocados nestas linhas, a distância $dist_{m,mm} = 9999$, penalizando as alocações dos postos m e mm . Na prática, a restrição 18 contribui para a convergência da busca para alocações promissoras, mas também indica que as alocações de m e mm não seriam consideradas “boas” ou ideais numa indústria.

$$pos_Y_{l+1,cc} \geq pos_Y_{l,c} \quad \forall l < L, c, cc \quad (13)$$

$$pos_Y_{l,c} \geq \frac{alt_{l,c}}{2} \quad \forall l = 1, c \quad (14)$$

$$distanc_Y_{l,c,ll,cc} = pos_Y_{ll,cc} - pos_Y_{l,c} \quad \forall l, c, ll \geq l, cc \quad (15)$$

$$distanc_Y_{l,c,ll,cc} = distanc_Y_{ll,cc,l,c} \quad \forall l, c, ll, cc \quad (16)$$

$$dist_Y_{m,mm} \geq distanc_Y_{l,c,ll,cc} - 9999 * (2 - pc_{m,l,c} - pc_{mm,l,c} - pc_{m,ll,cc} - pc_{mm,ll,cc}) \quad (17)$$

$$\forall m, mm/dem_{m,mm} \geq 0, l, c, ll, cc / l \neq ll \vee c \neq cc$$

$$dist_Y_{m,mm} \geq 9999 * (pc_{m,l,c} + pc_{mm,l,c} + pc_{m,ll,cc} + pc_{mm,ll,cc} - 1) \quad (18)$$

$$\forall m, mm/dem_{m,mm} \geq 0, l, c, ll, cc / l > ll + 1 \vee ll > l + 1$$

O modelo matemático gerado não permite sobreposições entre dois postos. Isso é garantido pelas restrições 19 e 20 para alocações em linhas distintas, onde $ll > l$, já que a restrição 7 impede sobreposições de postos numa mesma linha l . Na restrição 19, considerando-se que a soma da metade da largura de dois postos, respectivamente alocados na coluna c da linha l e na coluna cc da linha ll , seja maior que a distância entre os dois postos ($distanc_{l,c,ll,cc}$), então o lado esquerdo da restrição 19 terá valor positivo, indicando um risco de sobreposição e, portanto, forçando a variável binária $sobre_X_{l,c,ll,cc}$ a assumir o valor 1. Um exemplo desta situação é apresentado na figura 14, onde são apresentados três postos de trabalho, **A**, **B** e **C**. Os postos **A** e **B** estão na mesma linha l_1 e, por isso, nesta figura, $distanc_{l_1, c_1, l_1, c_2} = (larg_{l_1, c_1} + larg_{l_1, c_2}) / 2$. Mas, se os postos **A** e **B** estivessem afastados, então, seria $distanc_{l_1, c_1, l_1, c_2} > (larg_{l_1, c_1} + larg_{l_1, c_2}) / 2$. Já o terceiro posto, **C**, localizado na linha l_2 do *grid*, nota-se que ele “sobrepõe” os postos **A** e **B** no eixo x. Na parte inferior da figura 14 são apresentadas setas representando metade da largura dos postos **B** e **C**. Observe que estas setas (representando metade da largura dos

postos **B** e **C**) se sobrepõem, indicando que entre estes dois postos, $distanc_{l_2, c_1, l_1, c_2} < (larg_{l_2, c_1} + larg_{l_1, c_2})/2$. Consequentemente, teremos na restrição 19 que $sobre_X_{l_2, c_1, l_1, c_2} = 1$. Neste caso, a restrição 20 forçará que o espaçamento mínimo no eixo y entre as duas posições seja maior ou igual a metade da soma das alturas destes dois postos, como é mostrado visualmente na figura 14. As restrições 21 e 22 definem o limite superior dos eixos x e y, verificando o limite dos postos alocados na última coluna e linha, respectivamente.

$$\frac{(larg_{l,c} + larg_{ll,cc})}{2} - distanc_X_{l,c,ll,cc} \leq 9999 * sobre_X_{l,c,ll,cc} \quad (19)$$

$$\forall l, c, ll > l, cc$$

$$pos_Y_{ll,cc} \geq pos_Y_{l,c} + \frac{(alt_{l,c} + alt_{ll,cc})}{2} - 9999 * (1 - sobre_X_{l,c,ll,cc}) \quad (20)$$

$$\forall l, c, ll > l, cc$$

$$maior_X \geq pos_X_{l,c} + \frac{larg_{l,c}}{2} \quad \forall l, c = C \quad (21)$$

$$maior_Y \geq pos_Y_{l,c} + \frac{alt_{l,c}}{2} \quad \forall l = L, c \quad (22)$$

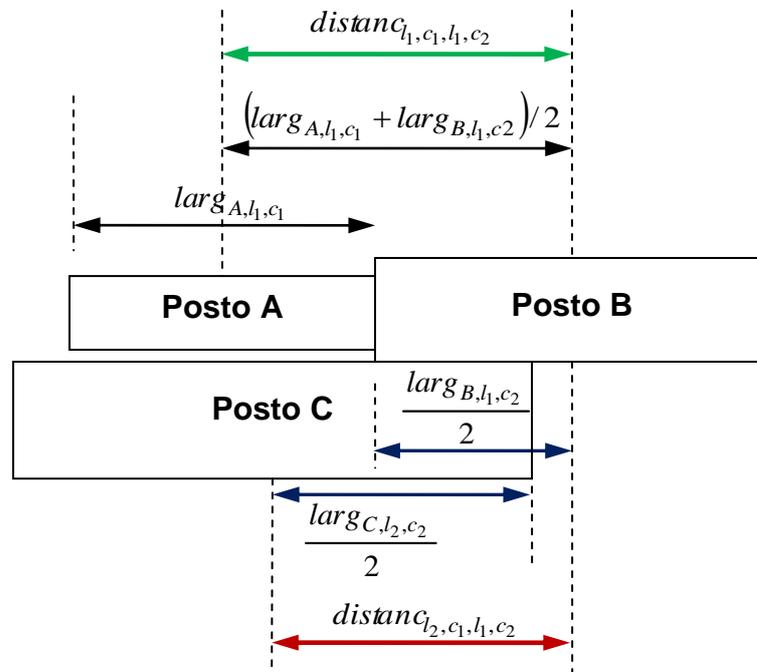


Figura 14 - Exemplo de análise de sobreposição.

3.2.2 Abordagem de Busca Tabu Híbrida

Uma segunda abordagem foi proposta para a solução do problema de *layout*. Trata-se de uma abordagem de Busca Tabu híbrida, que utiliza a BT para alocar os postos às linhas e uma abordagem PLIM para: *i*) definir o ordenamento das linhas, influenciado pelas alocações da BT; e *ii*) definir o posicionamento de cada posição (que receberá um posto de trabalho) do *grid*.

Na abordagem PLIM da seção anterior, a variável $pc_{m,l,c}$ define a alocação do posto de trabalho m à uma coluna c de uma linha l , havendo no máximo um posto alocado a cada posição (definida por uma coluna c de uma linha l). Nesta seção, a BT controlará a alocação da variável $pc_{m,l,c}$, tornando-a um parâmetro para a execução da abordagem PLIM, na BT híbrida. Neste caso, o que interessa para a BT são os dois primeiros índices (posto m e linha l) da variável $pc_{m,l,c}$. O índice c apenas controla o número de postos (máquinas) alocados numa linha, funcionando como uma lista de alocações de postos numa linha.

Na busca na vizinhança da BT, cada movimento corresponde ao deslocamento de um posto de trabalho de uma linha l , onde ele estava, para outra linha ll , onde $ll \neq l$. Neste movimento, o posto deslocado será sempre posicionado na posição cc da linha ll de, onde cc identifica a última posição da lista de postos alocados à linha ll . Na busca na vizinhança da BT proposta, serão testados todos os movimentos possíveis. A figura 11, apresentada anteriormente na seção 3.1.3, descreve o algoritmo de BT usado. Ele é reapresentado a seguir para maior comodidade do leitor. O Algoritmo 2, indicado na linha 8 da figura 11, realiza a busca na vizinhança, já descrita na figura 12. Um detalhe final da execução do modelo PLIM da BT híbrida é que, como a solução do PLIM (dependendo de como foi implementado) pode ter um tempo de execução elevado, foi implementada uma “poda” do PLIM, interrompendo a busca (solução do PLIM) sempre que a solução relaxada (durante a busca) for pior que o valor da melhor solução vizinha encontrada, identificando que esta busca não é mais promissora. Alguns detalhes adicionais da implementação são apresentados no capítulo dos resultados.

```

1:  $s^{best} = +\infty$ 
2: while não satisfez o critério de parada do
3:   Gerar solução inicial ( $s$ )
4:   if  $Z(s) < Z(s^{best})$  then
5:      $s^{best} = s$ 
6:   end if
7:   while não satisfez o critério de diversificação do
8:     Executar o Algoritmo 2 (gerar soluções vizinhas de  $s$ ) e selecionar
       o melhor vizinho ( $s'$ )
9:     if  $Z(s') < Z(s^{best})$  then
10:       $s^{best} = s'$ 
11:    end if
12:    Atualizar a solução vigente:  $s = s'$ 
13:  end while
14: end while
15: Escrever  $s^{best}$ 

```

Figura 11 - Algoritmo de Busca Tabu Implementado.

```

1: for  $m = 1$  to  $M$  do
2:   identifique a linha  $l$  em que o posto  $m$  está alocado na solução vigente
3:   for  $ll = 1$  to  $L$  do
4:     if  $ll \neq l$  then
5:       gerar a solução candidata  $s'$ , inserindo  $m$  na última posição  $c$  da lista da linha  $ll$ 
6:       executar o PLIM da BT híbrida
7:       identifica se  $s'$  é a melhor solução entre as soluções vizinhas
8:     end if
9:   end for
10: end for

```

Figura 15 - Algoritmo de busca na vizinhança. (Algoritmo 2)

- **Modelo matemático PLIM da BT híbrida**

Como dito anteriormente, a variável $pc_{m,l,c}$ que define a alocação do posto de trabalho m à uma coluna c de uma linha l foi definida na geração da solução candidata s' da BT. O papel do modelo PLIM na BT híbrida é de identificar a qualidade desta solução candidata, definindo a posição dos postos de trabalho e o ordenamento (ordem de posicionamento no *layout*) das linhas. Assim, $pc_{m,l,c}$ será um parâmetro no PLIM. Haverá apenas três variáveis binárias no PLIM:

- $prec_X_{l,c,cc}$ identifica, na linha l , se a posição c precede ou não a posição cc no eixo x ;
- $prec_Y_{l,||}$ identifica se a linha l precede ou não a linha $||$ no eixo y ;
- $sobre_X_{l,c,||,cc}$ variável binária que será um (1) se, no eixo x , há sobreposição entre as posições da coluna c da linha l e da coluna cc da linha $||$. Caso contrário, o valor será nulo.

O PLIM da BT híbrida contém as expressões 2, 5 e 6, apresentadas anteriormente, eliminando-se ou alterando-se todas as demais. A seguir são apresentadas as novas restrições inseridas a este modelo. A restrição 7 foi substituída neste modelo pelas restrições 23 e 24. Como os índices c e cc não tem um ordenamento definido na linha l neste modelo, se $prec_X_{l,c,cc} = 1$, então a posição c precede cc e a restrição 23 impõe que $pos_X_{l,cc} \geq pos_X_{l,c} + (larg_{l,cc} + larg_{l,c})/2$. Neste caso, a restrição 24 se torna sem efeito. Mas, se $prec_X_{l,c,cc} = 0$, então a posição c não precede cc (ou cc precede c) e a restrição 23 se tornará sem efeito, enquanto a restrição 24 impõe que $pos_X_{l,c} \geq pos_X_{l,cc} + (larg_{l,cc} + larg_{l,c})/2$. Este artifício utilizado nas restrições 23 e 24 é denominado de *Big-M*.

As restrições 25 e 30 substituem, respectivamente, as restrições 8 e 14. Como as colunas c não estão ordenadas como na abordagem da seção 3.2.1 e, na BT híbrida, c é apenas um índice da lista de alocações numa linha do *grid*, é preciso aplicar as restrições 25 e 30 a todos as posições, desde que haja um posto associado a esta posição (ou seja, $c \leq cont_l$), onde $cont_l$ indica a quantidade de postos alocados na linha l .

$$pos_X_{l,cc} \geq pos_X_{l,c} + \frac{(larg_{l,cc} + larg_{l,c})}{2} - 9999 * (1 - prec_X_{l,c,cc}) \quad (23)$$

$\forall l, c \leq cont_l, cc \leq cont_l \therefore cc \neq c$

$$pos_X_{l,c} \geq pos_X_{l,cc} + \frac{(larg_{l,cc} + larg_{l,c})}{2} - 9999 * prec_X_{l,c,cc} \quad (24)$$

$\forall l, c \leq cont_l, cc \leq cont_l \therefore cc \neq c$

$$pos_X_{l,c} \geq \frac{larg_{l,c}}{2} \quad \forall l, c \leq cont_l \quad (25)$$

As restrições 9, 10, 11, 12 e 19 foram agrupadas nas restrições 26 e 27, já que $pc_{m,l,c}$ são parâmetros na BT híbrida. Assim, $distanc_{l,c,ll,cc}$ não foi usada, pois já se sabe de antemão qual posição (identificada pelos índices c da linha l) foi associada ao posto m . Nas restrições 26 e 27, apesar dos somatórios para os índices l e c , o parâmetro $pc_{m,l,c} = 1$ impõe que apenas uma variável seja escrita para estes somatórios, que identifica a posição (referente a l e c) do posto m . O mesmo ocorre para os somatórios para os índices ll e cc , devido ao parâmetro $pc_{mm,ll,cc} = 1$, que identifica a posição (referente a ll e cc) do posto mm .

As restrições 13 e 20 foram substituídas neste modelo pelas restrições 28 e 29. Como as linhas l e ll não tem um ordenamento definido neste modelo, se $prec_Y_{l,ll} = 1$, então a linha l precede a linha ll e a restrição 28 se parece com a restrição 20, que foi eliminada. Neste caso, se $prec_Y_{l,ll} = 1$, a restrição 29 se tornará sem efeito. Mas, se $prec_X_{l,c,cc} = 0$, então a linha l não precede ll (ou ll precede l) e a restrição 28 se torna sem efeito, enquanto a restrição 29 impõe que $pos_Y_{l,c} \geq pos_Y_{ll,cc} + ((larg_{l,c} + larg_{ll,cc})/2) - 9999 * (1 - sobre_X_{l,c,ll,cc})$, parecendo-se com o inverso da restrição 20. As restrições 31 e 32 substituem a restrição 19, considerando-se que $pc_{m,l,c} = 1$ e $pc_{mm,ll,cc} = 1$ são parâmetros do problema. A restrição 32 é redundante, mas permite verificar que todas as combinações foram consideradas neste modelo PLIM associado à BT.

$$dist_X_{m,mm} \geq \sum_l \sum_c^{pc_{m,l,c}=1} Pos_X_{l,c} - \sum_{ll} \sum_{cc}^{pc_{mm,ll,cc}=1} Pos_X_{ll,cc} - \frac{(larg_{l,c} + larg_{ll,cc})}{2} * (1 - sobre_X_{l,c,ll,cc}) \quad \forall m, mm \quad (26)$$

$$dist_X_{m,mm} \geq \sum_{ll} \sum_{cc}^{pc_{mm,ll,cc}=1} Pos_X_{ll,cc} - \sum_l \sum_c^{pc_{m,l,c}=1} Pos_X_{l,c} - \frac{(larg_{l,c} + larg_{ll,cc})}{2} * (1 - sobre_X_{l,c,ll,cc}) \quad \forall m, mm \quad (27)$$

$$pos_Y_{ll,cc} \geq pos_Y_{l,c} + \frac{(alt_{l,c} + alt_{ll,cc})}{2} - 9999 * (1 - sobre_X_{l,c,ll,cc}) - 9999 * (1 - prec_Y_{l,ll}) \quad \forall l, c \leq cont_l, cc \leq cont_{ll} \therefore ll > l \quad (28)$$

$$pos_Y_{l,c} \geq pos_Y_{ll,cc} + \frac{(alt_{l,c} + alt_{ll,cc})}{2} - 9999 * (1 - sobre_X_{l,c,ll,cc}) - 9999 * prec_Y_{l,ll} \quad \forall l, c \leq cont_l, cc \leq cont_l \therefore ll > l \quad (29)$$

$$pos_Y_{l,c} \geq \frac{alt_{l,c}}{2} \quad \forall l, c \leq cont_l \quad (30)$$

$$larg_{l,c} + larg_{ll,cc} - 2 * dist_X_{m,mm} \leq 9999 * sobre_X_{l,c,ll,cc} \quad \forall m, mm, l, c, ll, cc \therefore pc_{m,l,c} = 1 \text{ e } pc_{mm,ll,cc} = 1 \quad (31)$$

$$sobre_X_{ll,cc,l,c} = sobre_X_{l,c,ll,cc} \quad \forall l, ll, c, cc / l \neq ll \vee c \neq cc \quad (32)$$

Assim como as restrições 26 e 27 foram geradas para definir a distância no eixo x entre dois postos, m e mm , as restrições 33 e 34 fazem o mesmo para o eixo y. As restrições 33 e 34 substituem as restrições 15, 16 e 17 da primeira abordagem PLIM. As restrições 35 e 36 substituem a restrição 18, penalizando a BT por alocar os postos m e mm nas linhas l e ll se $l > ll + 1$ ou $ll > l + 1$. Por fim, as restrições 37 e 38 substituem as restrições 21 e 22, definindo o limite superior dos eixos x e y. Neste caso, todas as posições são verificadas para identificar a largura total e a altura total, identificando a área total ocupada pelas máquinas.

$$dist_Y_{m,mm} \geq \sum_l \sum_{pc_{m,l,c}=1} pos_Y_{l,c} - \sum_{ll} \sum_{pc_{mm,ll,cc}=1} pos_Y_{ll,cc} \quad \forall m, mm \quad (33)$$

$$dist_Y_{m,mm} \geq \sum_{ll} \sum_{pc_{mm,ll,cc}=1} pos_Y_{ll,cc} - \sum_l \sum_{pc_{m,l,c}=1} pos_Y_{l,c} \quad \forall m, mm \quad (34)$$

$$dist_Y_{m,mm} \geq 9999 \quad \forall m, mm, l, ll > l + 1, c, cc / pc_{m,l,c} = 1 \therefore pc_{mm,ll,cc} = 1 \quad (35)$$

$$dist_Y_{m,mm} \geq 9999 \quad \forall m, mm, ll, l > ll + 1, c, cc / pc_{m,l,c} = 1 \therefore pc_{mm,ll,cc} = 1 \quad (36)$$

$$maior_X \geq pos_X_{l,c} + \frac{larg_{l,c}}{2} \quad \forall l, c \leq cont_l \quad (37)$$

$$maior_Y \geq pos_Y_{l,c} + \frac{alt_{l,c}}{2} \quad \forall l, c \leq cont_l \quad (38)$$

4 RESULTADOS

Este estudo foi montado em torno de dois pilares: a revisão da literatura e a pesquisa-campo (dados de produção, mapeamentos, *etc.*). Em um primeiro instante, foram formadas células de manufatura, visando minimizar os movimentos extracelulares dos produtos. As máquinas são agrupadas em células para, daí, alocarem-se peças e roteiros às máquinas.

Neste estudo, foi apresentada a abordagem de Busca Tabu (BT) (que gerou a publicação Rodrigues *et al.* (2013)) para resolver a primeira parte do problema de formação de células e, posteriormente, utilizado o *Branch-and-Bound (BB)* para resolver a segunda. A função objetivo foi a minimização dos movimentos extracelulares. É importante ainda mencionar que Caux *et al.* (2000) propôs o *Simulated Annealing*, para resolver o problema de alocação de máquinas nas células para, posteriormente, utilizar o *Branch-and-Bound (BB)* para associar as peças às respectivas células, tendo servido de inspiração para a abordagem de BT híbrida que foi proposta.

Em um segundo instante, após a formação de células com respectivas máquinas, roteiros e peças alocados, foi feito um estudo do *layout*, levando em conta o espaço ocupado (vide figura 20) e a quantidade de máquinas por operação, sendo obedecida a sequência de operações para cada número de tipo de produto (de 1 ao 50), conforme figuras 18 e 19.

É importante ressaltar que se trata de um estudo que utiliza dados reais de um setor de produção da empresa em questão. Alguns dados foram modificados para descaracterizar os dados da empresa devido à política de confidencialidade desta.

O objetivo será de verificar qual método apresenta melhores resultados em termos de tempo de processamento computacional e qualidade de solução (solução ótima local com relação a solução ótima global).

4.1 Descrição dos dados do problema

A figura 16 exibe a demanda dos produtos para o período considerado, M1 até M9. Para este estudo os dados de demanda usados são apresentados na coluna “média mensal” e ordenados de forma decrescente.

Código Produto	Qtde Fornecida									TOTAL	MÉDIA MENSAL	PERCENTUAL DO TOTAL	SOMA PERCENTUAL ACUMULADA
	M1	M2	M3	M4	M5	M6	M7	M8	M9				
1	29100	30280	29024	34680	40532	29746	36162	57887	37344	324755	36084	14,25%	14,25%
2	25200	31110	31866	29628	31649	26025	33467	36943	34291	280179	31131	12,29%	26,54%
3	22000	17700	28300	42600	28060	9792	17760	23424	1056	190692	21188	8,37%	34,91%
4	17274	15179	25732	20710	17345	15442	20415	17296	11237	160630	17848	7,05%	41,96%
5	7708	4209	13808	4752	14556	6960	8645	16109	7296	84043	9338	3,69%	45,64%
6	13396	1300	9000	5100	15240	9504	4128	16032	9792	83492	9277	3,66%	49,31%
7	2000	9800	5000	15996	9748	2112	9312	14016	10656	78640	8738	3,45%	52,76%
8	7348	8135	7265	4691	7737	3840	5765	4960	7180	56921	6325	2,50%	55,25%
9	4985	2652	6400	4236	5496	2559	10161	10792	7417	54698	6078	2,40%	57,65%
10	7553	5086	3632	8280	5226	4683	4545	9265	4795	53065	5896	2,33%	59,98%
11	1464	200	1308	1800	8688	6540	12228	8052	11208	51488	5721	2,26%	62,24%
12	3980	6696	5410	3545	4210	4821	4248	0	8004	40914	4546	1,80%	64,04%
13	18300	15850	5113	1431	0	0	0	0	0	40694	4522	1,79%	65,82%
14	19812	14035	4361	0	0	0	0	0	0	38208	4245	1,68%	67,50%
15	7200	3290	12205	5750	4024	1920	1344	0	0	35733	3970	1,57%	69,07%
16	1400	3280	5736	898	1500	2052	5892	2844	9576	33178	3686	1,46%	70,52%
17	5104	12100	10300	168	2184	0	660	108	540	31164	3463	1,37%	71,89%
18	208	1200	6400	1355	3868	6444	3552	4032	3648	30707	3412	1,35%	73,24%
19	24	2016	750	420	5815	2768	6434	4212	4638	27077	3009	1,19%	74,42%
20	11800	9459	3425	0	0	0	0	0	93	24777	2753	1,09%	75,51%
21	2408	0	920	1020	5006	3192	4512	2700	3156	22914	2546	1,01%	76,52%
22	4796	2000	5496	100	3572	1152	960	1440	3360	22876	2542	1,00%	77,52%
23	300	8136	468	268	0	450	4612	3432	4746	22412	2490	0,98%	78,50%
24	3912	1400	4696	1660	2880	2304	0	1152	1920	19924	2214	0,87%	79,38%
25	3419	1200	171	2929	4610	1308	2634	672	2784	19727	2192	0,87%	80,24%
26	787	3048	1655	6314	2098	1854	0	598	0	16354	1817	0,72%	80,96%
27	2000	1096	0	200	480	1296	3516	3936	3504	16028	1781	0,70%	81,66%
28	0	972	468	0	2532	300	3156	96	5580	13104	1456	0,57%	82,24%
29	400	0	0	1812	3936	192	1308	2112	1308	11068	1230	0,49%	82,72%
30	700	2500	4100	700	924	96	0	0	1248	10268	1141	0,45%	83,17%
31	0	96	600	1500	800	1824	0	1152	4224	10196	1133	0,45%	83,62%
32	2000	400	0	1436	1056	768	1080	1152	2208	10100	1122	0,44%	84,06%
33	0	300	500	1800	864	1248	576	1440	2400	9128	1014	0,40%	84,47%
34	96	0	0	156	0	1956	1248	1332	3700	8488	943	0,37%	84,84%
35	526	1349	1270	983	669	156	943	1670	810	8376	931	0,37%	85,21%
36	396	432	1020	0	3216	84	0	0	2208	7356	817	0,32%	85,53%
37	0	965	0	0	252	847	2616	0	2332	7012	779	0,31%	85,84%
38	96	0	456	4300	256	0	1536	0	0	6644	738	0,29%	86,13%
39	0	168	0	0	1020	1656	1080	2076	240	6240	693	0,27%	86,40%
40	600	0	600	2000	864	1728	0	320	0	6112	679	0,27%	86,67%
41	0	422	770	449	900	1915	1443	0	99	5998	666	0,26%	86,93%
42	100	1600	0	0	480	1080	744	1056	768	5828	648	0,26%	87,19%
43	216	96	624	0	0	1416	1740	324	1116	5532	615	0,24%	87,43%
44	192	0	500	1400	1440	480	0	480	384	4876	542	0,21%	87,64%
45	0	1500	300	1200	768	0	0	288	768	4824	536	0,21%	87,86%
46	6100	6108	10900	11400	7424	11880	8988	13536	10296	86632	9626	3,80%	91,66%
47	4600	3100	4000	13800	9580	4224	7392	3168	5184	55048	6116	2,42%	94,07%
48	6000	2400	7536	0	6804	4116	10380	6660	5916	49812	5535	2,19%	96,26%
49	4300	3500	3400	5500	1920	1344	1728	9696	5160	36548	4061	1,60%	97,86%
50	1800	84	4000	0	2016	192	5568	4524	2892	21076	2342	0,92%	98,79%

Figura 16. Produção parcial do produto no período considerado. Fonte: empresa estudada.

Ainda conforme figura 16, os “tipos” 1 a 45 do produto analisado possuem o mesmo roteiro de produção. É importante ainda mencionar que os produtos com códigos 46 a 50 foram posicionados no fim da tabela, embora não sejam os produtos

com menor volume de produção, ou seja, não obedecem a ordenação decrescente já mencionada aqui. Estes 50 tipos do produto designado com maior volume de produção, de um total de 83 tipos (ou seja, 60,24% dos tipos), respondem por 98,79% (de uma demanda total de 2.279.225 unidades do período M1 ao M9) dos volumes totais de produção para o período mencionado, seguindo o Princípio de Pareto (Figura 17).

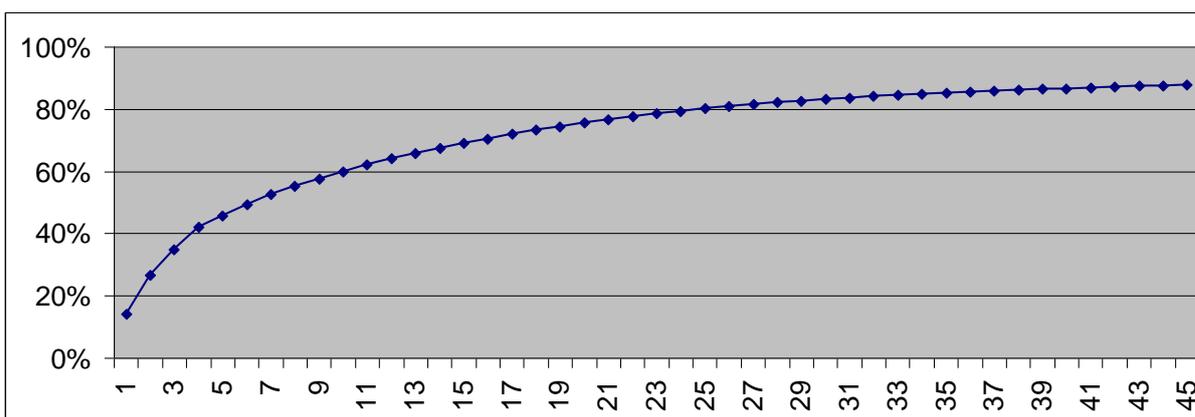


Figura 17. Gráfico de Pareto do total dos volumes fornecidos. Fonte: empresa estudada.

A figura 17 segue a distribuição de operações por produto (para os 50 produtos que respondem por 98,79% da demanda total para o período M1 a M9, considerado neste estudo). Pela alta significância do volume fornecido destes 50 números de tipo com relação ao volume total de vendas no período mencionado, estes 50 itens serão o objeto de estudo deste trabalho.

Pode-se verificar pela figura 18 que a sequência de operações dentro do processo é relativamente fixa e padronizada, podendo-se da mesma forma perceber que há apenas dois tipos diferentes de sequencial de produção. Nesta figura, apresenta-se na 1ª linha o tipo do produto. Na sequência, cada coluna apresenta, alternadamente, a operação executada e o tempo padrão (em UT) multiplicado pelo tamanho do lote, por operação, por produto.

A figura 19 apresenta a quantidade média (entre os meses M1 e M9) percentual, ocupada de uma máquina por produto ao longo de um mês. Esta figura apresenta, para cada produto, o número da operação na linha superior e, linha de baixo, a demanda média mensal multiplicada pelo tempo padrão (em UT), considerando-se *setup*, divididos pela disponibilidade mensal (em UT), em %. Cada coluna representa a sequência do roteiro de fabricação.

PRODUTO	OPERAÇÕES / TEMPO PADRÃO X TAMANHO LOTE DIVIDIDOS POR 150															
	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO	MAQ	TEMPO
1	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
2	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
3	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
4	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
5	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
6	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
7	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
8	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
9	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
10	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
11	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
12	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
13	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
14	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
15	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
16	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
17	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
18	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
19	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
20	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
21	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
22	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
23	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
24	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
25	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
26	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
27	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
28	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
29	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
30	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
31	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
32	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
33	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
34	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
35	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
36	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
37	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
38	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
39	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
40	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
41	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
42	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
43	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
44	O1	200	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
45	O1	60	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50		
46	O9	15	O8	75	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50
47	O9	15	O8	75	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50
48	O9	15	O8	75	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50
49	O9	15	O8	75	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50
50	O9	15	O8	75	O2	32,5	O3	30,5	O4	31	O5	30,5	O6	22,5	O7	50

Figura 18. Operações executadas na linha de montagem do produto. Fonte: empresa estudada

A figura 20 indica a quantidade de máquinas disponíveis para cada operação, bem como as dimensões por máquina. Atualmente, o espaço disponível é de 13,7 x 23,6 UM (Área total de 323,32UM²). Ainda conforme esta figura, a área total ocupada pelas máquinas é de 312,85 UM². Ao contrário do estudo de Chwif *et al* (1998), considera-se que o fator de aspecto das máquinas não pode variar. Ou seja, a relação entre altura e largura das máquinas (postos de trabalho), denominada fator de aspecto não pode variar neste trabalho.

PRODUTO	DEMANDA MÉDIA MENSAL X TEMPO PADRÃO / DISPONIBILIDADE MENSAL						
	O1	O2	O3	O4	O5	O6	O7
1	97,1%	15,8%	14,8%	15,1%	14,8%	10,9%	24,3%
2	83,8%	13,6%	12,8%	13,0%	12,8%	9,4%	20,9%
3	57,0%	9,3%	8,7%	8,8%	8,7%	6,4%	14,3%
4	48,0%	7,8%	7,3%	7,4%	7,3%	5,4%	12,0%
5	25,9%	4,2%	4,0%	4,0%	4,0%	2,9%	6,5%
6	25,1%	4,1%	3,8%	3,9%	3,8%	2,8%	6,3%
7	25,0%	4,1%	3,8%	3,9%	3,8%	2,8%	6,2%
8	23,5%	3,8%	3,6%	3,6%	3,6%	2,6%	5,9%
9	17,0%	2,8%	2,6%	2,6%	2,6%	1,9%	4,3%
10	16,5%	2,7%	2,5%	2,6%	2,5%	1,9%	4,1%
11	15,9%	2,6%	2,4%	2,5%	2,4%	1,8%	4,0%
12	15,4%	2,5%	2,3%	2,4%	2,3%	1,7%	3,8%
13	12,2%	2,0%	1,9%	1,9%	1,9%	1,4%	3,1%
14	12,2%	2,0%	1,9%	1,9%	1,9%	1,4%	3,0%
15	11,4%	1,9%	1,7%	1,8%	1,7%	1,3%	2,9%
16	10,9%	1,8%	1,7%	1,7%	1,7%	1,2%	2,7%
17	9,9%	1,6%	1,5%	1,5%	1,5%	1,1%	2,5%
18	9,3%	1,5%	1,4%	1,4%	1,4%	1,0%	2,3%
19	9,2%	1,5%	1,4%	1,4%	1,4%	1,0%	2,3%
20	7,4%	1,2%	1,1%	1,1%	1,1%	0,8%	1,9%
21	6,9%	1,1%	1,0%	1,1%	1,0%	0,8%	1,7%
22	6,8%	1,1%	1,0%	1,1%	1,0%	0,8%	1,7%
23	6,3%	1,0%	1,0%	1,0%	1,0%	0,7%	1,6%
24	6,0%	1,0%	0,9%	0,0%	0,0%	0,7%	1,5%
25	5,9%	1,0%	0,9%	0,0%	0,0%	0,7%	1,5%
26	4,9%	0,8%	0,7%	0,0%	0,0%	0,6%	1,2%
27	4,8%	0,8%	0,7%	0,0%	0,0%	0,5%	1,2%
28	3,9%	0,6%	0,6%	0,0%	0,0%	0,4%	1,0%
29	3,3%	0,5%	0,5%	0,0%	0,0%	0,4%	0,8%
30	3,1%	0,5%	0,5%	0,0%	0,0%	0,3%	0,8%
31	3,0%	0,5%	0,5%	0,0%	0,0%	0,3%	0,8%
32	3,0%	0,5%	0,5%	0,0%	0,0%	0,3%	0,8%
33	2,7%	0,4%	0,4%	0,0%	0,0%	0,3%	0,7%
34	2,5%	0,4%	0,4%	0,0%	0,0%	0,3%	0,6%
35	2,5%	0,4%	0,4%	0,0%	0,0%	0,3%	0,6%
36	2,2%	0,4%	0,3%	0,0%	0,0%	0,2%	0,5%
37	2,1%	0,3%	0,3%	0,0%	0,0%	0,2%	0,5%
38	2,0%	0,3%	0,3%	0,0%	0,0%	0,2%	0,5%
39	1,9%	0,3%	0,3%	0,0%	0,0%	0,2%	0,5%
40	1,8%	0,3%	0,3%	0,0%	0,0%	0,2%	0,5%
41	1,8%	0,3%	0,3%	0,0%	0,0%	0,2%	0,4%
42	1,7%	0,3%	0,3%	0,0%	0,0%	0,2%	0,4%
43	1,7%	0,3%	0,3%	0,0%	0,0%	0,2%	0,4%
44	1,5%	0,2%	0,2%	0,0%	0,0%	0,2%	0,4%
45	1,4%	0,2%	0,2%	0,0%	0,0%	0,2%	0,4%
46	4,1%	6,1%	2,7%	2,5%	2,5%	1,8%	4,1%
47	3,7%	5,6%	2,4%	2,3%	2,3%	1,7%	3,7%
48	2,7%	4,0%	1,7%	1,6%	1,7%	1,2%	2,7%
49	2,0%	3,0%	1,3%	1,2%	1,3%	0,9%	2,0%
50	1,7%	2,5%	1,1%	1,0%	1,0%	0,8%	1,7%

Figura 19. Percentual de ocupação de uma máquina por operação. Fonte: empresa estudada

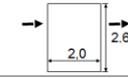
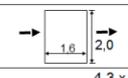
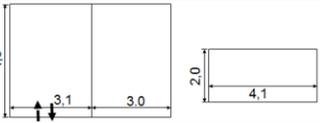
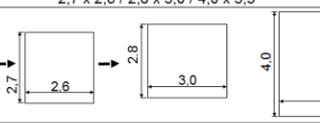
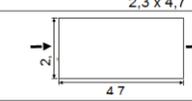
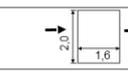
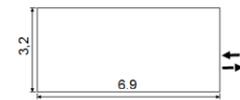
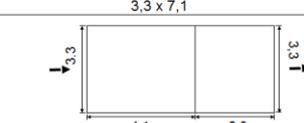
OPERAÇÃO	QUANTIDADE DE MÁQUINAS	DIMENSÕES POR MÁQUINA COM TODOS AGREGADOS (EM UM), COM INDICAÇÃO DE ENTRADA E SAÍDA DE PEÇAS	ÁREA OCUPADA POR MÁQUINA (EM UM ²)	ÁREA TOTAL OCUPADA PELAS MÁQUINAS (EM UM ²)	FA	ÁREA TOTAL	MÁQUINAS	ÁREA TOTAL POR OPERAÇÃO
O1	8	2,6 x 2,0 	5,20	41,60	1,3	5,2	1 A 8	41,6
O9	1	2,0 x 1,6 	3,20	3,20	1,25	3,2	9	3,2
O2	2	4,3 x 6,1 / 2,0 x 4,1 	43,86	87,72	0,53	34,83	10 A 11	69,66
O3	2	2,7 x 2,6 / 2,8 x 3,0 / 4,0 x 3,5 	36,40	72,80	0,44	40	12 A 13	80
O4	2	3,5 x 4,4 	15,40	30,80	0,8	15,4	14 A 15	30,8
O5	2	2,3 x 4,7 	10,81	21,62	0,49	10,81	16 A 17	21,62
O6	3	2 x 1,6 	3,20	9,60	1,25	3,2	18 A 20	9,6
O7	1	3,2 x 6,9 	22,08	22,08	0,46	22,08	21	22,08
O8	1	3,3 x 7,1 	23,43	23,43	0,46	23,43	22	23,43
Total	22		163,58	312,85				

Figura 20. Dimensões e quantidades de máquinas no setor (em U.M.). Fonte: empresa estudada.

A figura 21 ilustra a quantidade de máquinas com respectivos tempos padrão e tamanhos de lotes tratados na produção. Pode-se perceber um grau de preenchimento mensal relativamente bom para as operações O1, O9, O2, O3, O4 e O5, acima de 50%, até porque neste estudo não são consideradas perdas de produção com peças reaproveitadas, retrabalhadas e sucateadas. O grau de ocupação destas máquinas é relevante, tendo-se em conta que são máquinas com custo de depreciação relativamente alto; ao contrário dos postos de trabalho que correspondem à operação O6, que são bancadas de custo de aquisição muito menor se comparados a aquelas operações (mais que 100 vezes menor). É também importante mencionar, para as operações O7 e O8, que o grau de ocupação para estas não é relevante, já que estas são utilizadas apenas em uma amostragem da

produção total. Isto não quer dizer que as máquinas referentes a estas operações não serão utilizadas nos estudos referentes a este trabalho.

OPERAÇÃO	QUANTIDADE DE MÁQUINAS (EM UNIDADES)	TEMPO PADRÃO (UT)	TEMPO SETUP (UT)	CAPACIDADE DIÁRIA POR OPERAÇÃO (UNIDADES) - DESCONSIDERANDO SETUP	CAPACIDADE DIÁRIA POR OPERAÇÃO (UNIDADES) - CONSIDERANDO SETUP	CAPACIDADE MENSAL POR OPERAÇÃO - DESCONSIDERANDO-SE SETUP	CAPACIDADE MENSAL POR OPERAÇÃO - CONSIDERANDO-SE SETUP	GRAU DE PREENCHIMENTO MENSAL POR OPERAÇÃO - DESCONSIDERANDO-SE SETUP	GRAU DE PREENCHIMENTO MENSAL POR OPERAÇÃO - CONSIDERANDO-SE SETUP	TAMANHO LOTE MÍNIMO (UNIDADES)	TEMPO PADRÃO X TAMANHO LOTE (UT X UNIDADES)
O1	8	200,0	0	14688	14688	484704	484704	52%	52%	150	30000
O9	1	50	0	7344	7344						7500
O2	2	32,5	4500	22597	17612	497132	387471	51%	65%		4875
O3	2	30,5	7500	24079	15226	529731	334977	48%	76%		4575
O4	2	31	900	23690	22645	521187	498194	49%	51%		4650
O5	2	30,5	300	24079	23725	529731	521941	48%	49%		4575
O6	3	22,5	0	48960	48960	1077120	1077120	24%	24%		3375
O7	1	12	0	30600	30600	673200	673200	38%	38%		1800
O8	1	75	1500	4896	4536	107712	99792	235%	254%	11250	

Figura 21. Código de cada operação, quantidade disponível de máquinas, tempos de operação e setup, bem como tamanho de lote para cada operação – Fonte: empresa estudada

Ainda conforme a Figura 21, considerando-se os *setups*, o grau de ocupação das máquinas teve valores médios elevados de 45% para 53%, ou seja, houve um aumento no grau de ocupação considerado para as máquinas de 8%, após considerar o *setup* nas máquinas; isso para uma média de 18 *setups* por dia, que é um valor médio, entre os meses M1 a M9, de *setups* por máquina por dia, conforme fornecido pela empresa. Para 22 dias úteis por mês e um tempo diário disponível de 367200 UT (sem considerar o *setup*), obtém-se um tempo diário disponível final de 337824 UT, considerando-se o *setup* nas máquinas (ou de 7432128 UT por mês, considerando-se o *setup*).

4.1.1 Características do problema analisado

- Para efeito de simplificação, não serão consideradas as restrições técnicas (ou seja, todas as máquinas fazem todos os produtos).

- Os agregados hidráulicos, pneumáticos e elétricos, bem como as áreas de circulação dos operadores em cada operação, serão considerados neste estudo como partes integrantes das máquinas.

- Em algumas operações, as máquinas não possuem uma forma retangular perfeita; no entanto, para efeitos de simplificação no estudo, adotou-se que as máquinas possuem este formato.

- O sequencial de operações, conforme figuras 20 e 21, deve necessariamente ser obedecido, não podendo, portanto, sofrer alterações.

- Será considerado que a linha em questão seja balanceada.

- Não será considerada a variação de tempo de operação e de setup por número de tipo de produto, para efeitos de simplificação.

- Será possível movimentar e rotacionar livremente a máquina dentro do estudo de *layout*, mas obedecendo-se as idéias de adesão dos agregados e de local de entrada e saída de peças, conforme indicado na figura 20.

- Neste estudo, o *layout* do chão-de-fábrica poderá ter uma infinidade de formatos, não sendo, necessariamente, um arranjo tradicional em linha reta.

- A operação O1 não será colocada em sala climatizada, ao contrário do que acontece no caso real, até porque não há comprovações técnicas seguras que seja realmente necessário fazê-lo. Nesta mesma operação, será considerado um grau de aproveitamento médio de 30% (ou seja, da cada 100 tentativas de montagem, 35 peças são montadas), o que é bastante razoável, tendo em conta estudos que foram feitos no passado acerca destas operações – este grau de aproveitamento já está embutido no Tempo Padrão, conforme figura 24.

- Dentro de cada operação, as máquinas serão consideradas iguais em dimensão.

- Para cada linha de produção, as máquinas relacionadas às operações O3, O4 e O5 trabalham em uma linha automática, portanto, as mesmas deverão estar em linha e na sequência.

- As operações O1 e O9, por serem idênticas (diferenciando-se apenas pela tecnologia empregada) terão sua capacidade somada para efeitos de considerações de capacidade, conforme figura 21.

- São considerados 22 dias úteis em um mês, em média, neste estudo, para os cálculos de Capacidade Mensal, conforme figura 21; da mesma forma, é considerada uma média de 18 setups de máquinas por dia, por máquina, em média.

4.2 Implementação da Solução do Problema de Formação de Células

A figura 22 apresenta o sistema de formação de células desenvolvido para solucionar o problema de formação de células. Este estudo considera a formação de duas células para um total de 22 máquinas que serão alocadas.

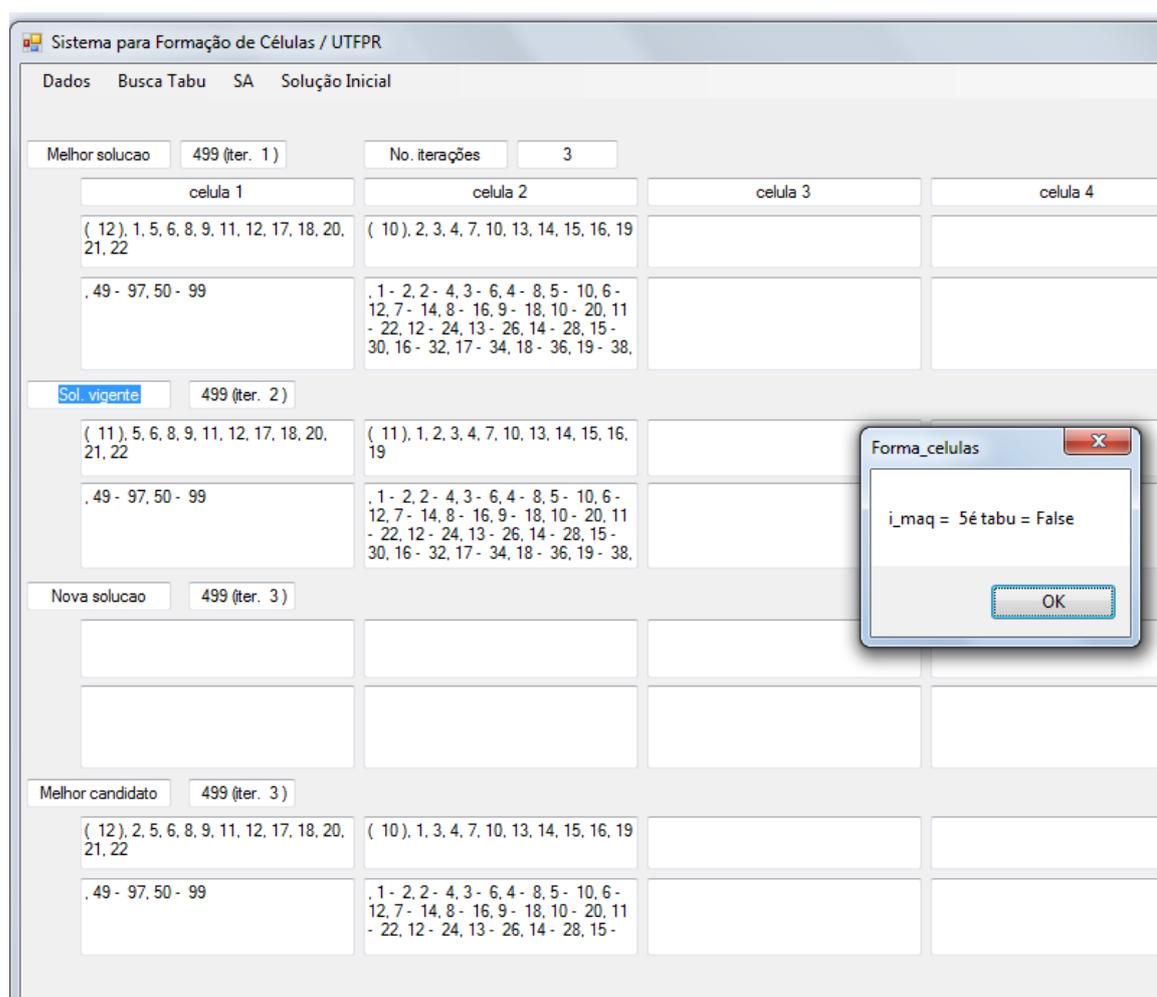


Figura 22. Tela do sistema de formação de células no Visual Basic 2008. Fonte: Autor

A figura 22 exibe a melhor solução encontrada, a solução vigente, a nova solução e o melhor candidato. À frente destas soluções, exibe-se o valor da função objetivo (que é o custo ponderado das operações extracelulares) e, entre parênteses, a iteração que gerou a solução correspondente. Para cada uma destas soluções, nas células 1 e 2, é mostrado no quadro logo abaixo o número de

máquinas alocadas entre parênteses, e, na sequência, quais máquinas foram alocadas na célula correspondente.

A solução do problema de formação de células pode ser automatizada ou “manual”, onde cada passo de cada iteração é visualizável. Na figura 22, a janela “Forma_células” exibida à frente ilustra a “visão” da interface do Visual Basic no “modo manual” de solução do problema. Nela, a janela “Forma_células” apresenta qual máquina foi selecionada para mudar de célula e se a mudança foi melhor ou não (considerando-se a minimização do valor da função objetivo). Caso o movimento da máquina seja tabu, o valor exibido neste quadro é “*true*”; caso contrário, “*false*”.

4.3 Resultados da Solução do Problema de Formação de Células

Nesta seção são apresentados resultados da otimização aplicada apenas aos dados da empresa estudada.

Como o local da empresa estudada conta com 22 postos de trabalho (máquinas), seria possível aplicar a otimização do *layout* sem solucionar o problema de formação de célula. Para isso, bastaria realizar uma alocação “estimada” das demandas em cada posto. Este caso será analisado no estudo do *layout* (identificado como caso 3). O propósito da solução do problema de formação de células é, em primeiro lugar, tentar simplificar e melhorar os resultados do *layout*. Foram propostos dois casos para o problema de formação de células:

- a) Caso 1: Limite Máximo de 13 Postos;
- b) Caso 2: Limite Máximo de 12 Postos.

As figuras 23 e 24 apresentam, respectivamente, uma solução inicial e, após quatro iterações, a solução ótima para o caso 1. No caso 1, a solução ótima (de custo 0) implica a alocação de toda a demanda em uma das células. Isso ocorreu porque o período considerado, como comentado anteriormente neste capítulo, apresenta ocupação média inferior a 60%. Ainda assim, este cenário foi considerado interessante para saber-se o que ocorrerá com o *layout*.

Dados Busca Tabu SA Solução Inicial

Melhor solucao 375 (iter. 0) No. iterações 0

celula 1	celula 2
(10), 3, 5, 6, 7, 8, 9, 16, 19, 21, 22	(12), 1, 2, 4, 10, 11, 12, 13, 14, 15, 17, 18, 20
. 10 - 19, 11 - 21, 14 - 27, 16 - 31, 17 - 33, 18 - 35, 19 - 37, 20 - 39, 21 - 41, 22 - 43, 23 - 45, 24 - 47, 25 - 49, 26 - 51, 27 - 53, 28 - 55, 29 - 57,	. 1 - 2, 2 - 4, 3 - 6, 4 - 8, 5 - 10, 6 - 12, 7 - 14, 8 - 16, 9 - 18, 12 - 24, 13 - 26, 15 - 30, 31 - 62, 33 - 66, 34 - 68, 35 - 70, 36 - 72, 37 - 74, 38 - 76,

Sol. vigente

Nova solucao 375 (iter. 0)

celula 1	celula 2
(10), 3, 5, 6, 7, 8, 9, 16, 19, 21, 22	(12), 1, 2, 4, 10, 11, 12, 13, 14, 15, 17, 18, 20
. 10 - 19, 11 - 21, 14 - 27, 16 - 31, 17 - 33, 18 - 35, 19 - 37, 20 - 39, 21 - 41, 22 - 43, 23 - 45, 24 - 47, 25 - 49, 26 - 51, 27 - 53, 28 - 55, 29 - 57,	. 1 - 2, 2 - 4, 3 - 6, 4 - 8, 5 - 10, 6 - 12, 7 - 14, 8 - 16, 9 - 18, 12 - 24, 13 - 26, 15 - 30, 31 - 62, 33 - 66, 34 - 68, 35 - 70, 36 - 72, 37 - 74, 38 - 76,

Melhor candidato 375 (iter. 0)

celula 1	celula 2
(10), 3, 5, 6, 7, 8, 9, 16, 19, 21, 22	(12), 1, 2, 4, 10, 11, 12, 13, 14, 15, 17, 18, 20
. 10 - 19, 11 - 21, 14 - 27, 16 - 31, 17 - 33, 18 - 35, 19 - 37, 20 - 39, 21 - 41, 22 - 43, 23 - 45, 24 - 47, 25 -	. 1 - 2, 2 - 4, 3 - 6, 4 - 8, 5 - 10, 6 - 12, 7 - 14, 8 - 16, 9 - 18, 12 - 24, 13 - 26, 15 - 30, 31 - 62, 33 - 66, 34 -

Figura 23. Solução inicial do caso 1.

Dados Busca Tabu SA Solução Inicial

Melhor solucao 0 (iter. 4) No. iterações 4

celula 1	celula 2
(13), 1, 2, 3, 5, 6, 9, 11, 14, 16, 18, 20, 21, 22	(9), 4, 7, 8, 10, 12, 13, 15, 17, 19
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	

Sol. vigente 7 (iter. 3)

celula 1	celula 2
(12), 2, 3, 5, 6, 9, 11, 14, 16, 18, 20, 21, 22	(10), 1, 4, 7, 8, 10, 12, 13, 15, 17, 19
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	. 37 - 74, 40 - 80, 41 - 82, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Nova solucao 79 (iter. 4)

celula 1	celula 2
(11), 2, 3, 5, 6, 9, 11, 14, 16, 18, 20, 21	(11), 1, 4, 7, 8, 10, 12, 13, 15, 17, 19, 22
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	. 36 - 72, 39 - 78, 40 - 80, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Melhor candidato 0 (iter. 4)

celula 1	celula 2
(13), 1, 2, 3, 5, 6, 9, 11, 14, 16, 18, 20, 21, 22	(9), 4, 7, 8, 10, 12, 13, 15, 17, 19
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 -	

Figura 24. Solução ótima do caso 1.

Dados Busca Tabu SA Solução Inicial

Melhor solucao 0 (iter. 5) No. iterações 5

celula 1	celula 2
(13), 1, 5, 6, 7, 8, 9, 11, 13, 16, 19, 20, 21, 22	(9), 2, 3, 4, 10, 12, 14, 15, 17, 18
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	

Sol. vigente 7 (iter. 4)

celula 1	celula 2
(12), 5, 6, 7, 8, 9, 11, 13, 16, 19, 20, 21, 22	(10), 1, 2, 3, 4, 10, 12, 14, 15, 17, 18
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	. 37 - 74, 40 - 80, 41 - 82, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Nova solucao 79 (iter. 5)

celula 1	celula 2
(11), 5, 6, 7, 8, 9, 11, 13, 16, 19, 20, 21	(11), 1, 2, 3, 4, 10, 12, 14, 15, 17, 18, 22
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	. 36 - 72, 39 - 78, 40 - 80, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Melhor candidato 0 (iter. 5)

celula 1	celula 2
(13), 1, 5, 6, 7, 8, 9, 11, 13, 16, 19, 20, 21, 22	(9), 2, 3, 4, 10, 12, 14, 15, 17, 18
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 -	

Figura 25. Solução inicial do caso 2.

Dados Busca Tabu SA Solução Inicial

Melhor solucao 7 (iter. 4) No. iterações 4

celula 1	celula 2
(12), 1, 3, 5, 6, 10, 12, 13, 16, 18, 20, 21, 22	(10), 2, 4, 7, 8, 9, 11, 14, 15, 17, 19
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 - 29, 16 - 31, 17 - 33, 18 - 35, 19 - 37,	. 37 - 74, 40 - 80, 41 - 82, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Sol. vigente 24 (iter. 3)

celula 1	celula 2
(11), 1, 3, 5, 6, 10, 12, 13, 16, 18, 21, 22	(11), 2, 4, 7, 8, 9, 11, 14, 15, 17, 19, 20
. 11 - 21, 14 - 27, 19 - 37, 23 - 45, 27 - 53, 36 - 71, 38 - 75, 41 - 81, 42 - 83, 43 - 85, 44 - 87, 45 - 89, 46 - 91, 47 - 93, 48 - 95, 49 - 97	. 1 - 2, 2 - 4, 3 - 6, 4 - 8, 5 - 10, 6 - 12, 7 - 14, 8 - 16, 9 - 18, 10 - 20, 12 - 24, 13 - 26, 15 - 30, 16 - 32, 17 - 34, 18 - 36, 20 - 40, 21 - 42, 22 - 44,

Nova solucao 96 (iter. 4)

celula 1	celula 2
(10), 1, 3, 5, 6, 10, 12, 13, 16, 18, 21	(12), 2, 4, 7, 8, 9, 11, 14, 15, 17, 19, 20, 22
. 11 - 21, 14 - 27, 19 - 37, 23 - 45, 27 - 53, 36 - 71, 40 - 79, 41 - 81, 42 - 83, 43 - 85, 44 - 87, 45 - 89, 46 - 91, 47 - 93, 48 - 95, 49 - 97	. 1 - 2, 2 - 4, 3 - 6, 4 - 8, 5 - 10, 6 - 12, 7 - 14, 8 - 16, 9 - 18, 10 - 20, 12 - 24, 13 - 26, 15 - 30, 16 - 32, 17 - 34, 18 - 36, 20 - 40, 21 - 42, 22 - 44,

Melhor candidato 7 (iter. 4)

celula 1	celula 2
(12), 1, 3, 5, 6, 10, 12, 13, 16, 18, 20, 21, 22	(10), 2, 4, 7, 8, 9, 11, 14, 15, 17, 19
. 1 - 1, 2 - 3, 3 - 5, 4 - 7, 5 - 9, 6 - 11, 7 - 13, 8 - 15, 9 - 17, 10 - 19, 11 - 21, 12 - 23, 13 - 25, 14 - 27, 15 -	. 37 - 74, 40 - 80, 41 - 82, 42 - 84, 43 - 86, 44 - 88, 45 - 90, 46 - 92, 47 - 94, 48 - 96, 49 - 98

Figura 26. Solução ótima do caso 2.

As figuras 25 e 26 apresentam, respectivamente, uma solução inicial e a solução ótima para o caso 2. Diferentemente do caso 1, o limite de 12 postos de trabalho impôs no caso 2 a realização de operações extra-celulares no posto 17. No caso 2, a solução ótima tem custo 7, havendo alocação de demanda nas duas células.

A tabela 7 apresenta a evolução média das soluções da BT nos dois casos. Observa-se que, normalmente, em menos de 20 iterações ou um pouco mais, quando a BT fica presa num ponto de ótimo local. Apesar de aparentar uma convergência prematura, o que se observou foi que o problema tratado pode ser considerado de pequeno porte para a abordagem proposta. Assim, instâncias maiores serão testadas no futuro. Observou-se que a solução ótima foi identificada sempre em menos de 2 segundos.

Como resultado da solução ótima obtida, respectivamente, para os casos 1 (figura 24) e 2 (figura 26), obteve-se as demandas apresentadas nas tabelas 5 e 6.

Tabela 5 - Demanda caso 1.

origem	destino	demanda
maq1	.maq10	370
maq2	.maq10	368
maq3	.maq10	366
maq4	.maq10	363
maq5	.maq9	19
maq6	.maq9	16
maq7	.maq9	14
maq8	.maq9	12
maq9	.maq11	61
maq11	.maq13	61
maq13	.maq15	61
maq15	.maq17	61
maq17	.maq20	61
maq21	.maq22	141
maq10	.maq21	141
maq10	.maq12	1608
maq12	.maq14	1608
maq14	.maq16	1608
maq16	.maq18	809
maq16	.maq19	799
maq18	.maq20	809
maq19	.maq20	709

Tabela 6 - Demanda caso 2.

origem	destino	demanda
maq1	.maq10	388
maq2	.maq10	384
maq3	.maq10	380
maq4	.maq10	376
maq5	.maq9	5
maq6	.maq9	4
maq7	.maq9	3
maq8	.maq9	2
maq9	.maq11	10
maq11	.maq13	10
maq13	.maq15	10
maq15	.maq17	10
maq17	.maq20	10
maq21	.maq22	141
maq10	.maq21	141
maq10	.maq12	1749
maq12	.maq14	1749
maq14	.maq16	1749
maq16	.maq18	875
maq16	.maq19	874
maq18	.maq20	875
maq19	.maq20	874

Tabela 7 - Evolução da média da melhor solução encontrada.

Iteração	Caso 1	Caso 2
0	349,4	688,4
1	225,6	500,8
2	50,2	295
3	26,2	163,4
4	9,6	42,6
5	6,2	18
6	4,8	9
7	4,8	9
8	4,8	9
9	4,8	9
10	4,8	9
11	6,2	9
12	6,2	9
13	6,2	9
14	4,2	9
15	4,2	7
16	2,8	7
17	2,8	7
18	2,8	7
19	2,8	7
20	2,8	7

As figuras 27, 28 e 29 demonstram graficamente a convergência da solução para o caso 1 (4 linhas), caso 2 (4 linhas) e caso 1 (5 linhas), respectivamente.

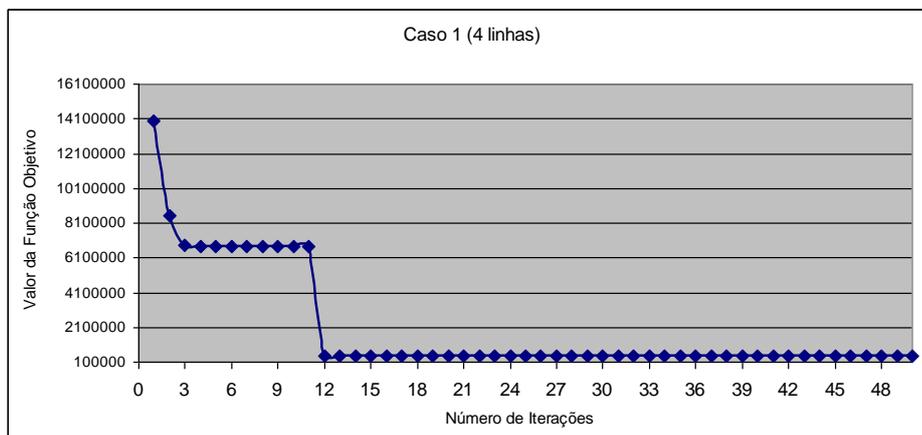


Figura 27 – Convergência da solução para o caso 1 – configuração de 4 linhas.

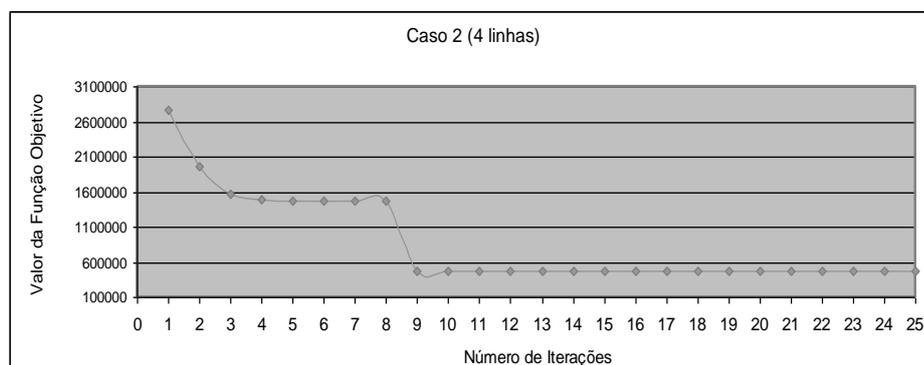


Figura 28 – Convergência da solução para o caso 2 – configuração de 4 linhas.

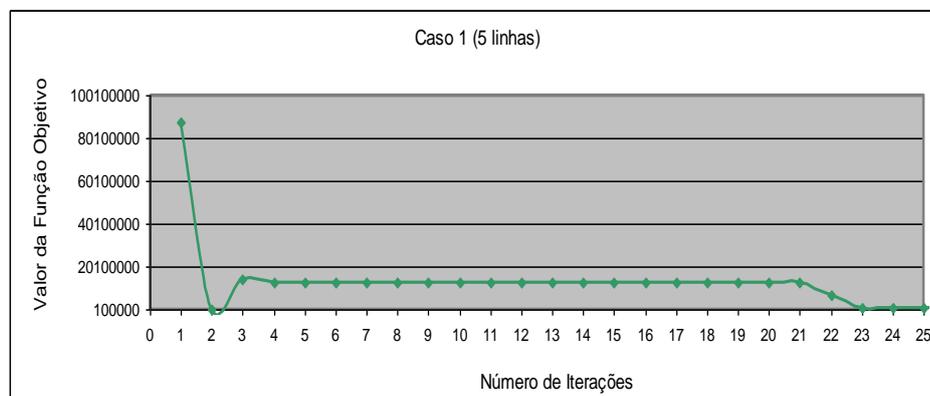


Figura 29 - Convergência da solução para o caso 1 – configuração de 5 linhas.

4.4 Resolução do Problema de Layout

Inicialmente, considerou-se a possibilidade de reaproveitar o trabalho de Weller (2008). Porém, os testes realizados como software GAMS/CPLEX foram capazes de solucionar o problema de *layout*. Assim, foram implementadas duas abordagens para a solução do *layout*. A primeira abordagem, apresentada na seção 3.2.1, é utilizar programação linear inteira mista (PLIM). A segunda abordagem é uma BT híbrida, que utiliza PLIM para definir as posições dos postos de trabalho. A seguir, apresenta-se um resumo da solução do problema de *layout*.

Como resultado da solução para o problema de *layout*, que seria o caso 3, obteve-se as demandas apresentadas na tabela 8.

Tabela 8 - Demanda caso 3

origem	destino	demanda
maq1	.maq9	191
maq2	.maq9	191
maq3	.maq9	191
maq4	.maq9	191
maq5	.maq10	191
maq6	.maq10	191
maq7	.maq10	191
maq8	.maq10	191
maq9	.maq11	764
maq11	.maq13	764
maq13	.maq15	764
maq15	.maq17	764
maq21	.maq22	141
maq10	.maq21	141
maq10	.maq12	905
maq12	.maq14	905
maq14	.maq16	905
maq16	.maq18	453
maq16	.maq19	452
maq18	.maq20	453
maq19	.maq20	452

4.4.1 Soluções Obtidas pela Abordagem PLIM

Inicialmente, serão apresentadas as soluções do modelo da seção 3.2.1, que representa uma abordagem de programação linear inteira mista. Foram geradas duas soluções para cada caso, considerando-se a possibilidade de organizar o *grid*

de linhas em quatro e, também em cinco linhas. A sequência desta seção apresenta um a um cada um dos três casos analisados. Assim, a análise que se pode fazer dos resultados é que, devido à complexidade do problema de *layout*, houve grande variação nos valores da função objetivo (identificado como `total.L` nos casos do PLIM). Para todos os casos, apresenta-se a área total a ser ocupada pela solução, identificada pelas variáveis `maior_X.L` e `maior_Y.L`. As variáveis de alocação dos postos numa linha e coluna, $pc_{m,l,c}$, são apresentados como `ordem_pc.L` nos casos abaixo.

a) Caso 1 (13 postos)

- **Grid de quatro linhas – PLIM**

A solução obtida foi (valor da função objetivo)::

```
----      266 VARIABLE total.L                =      875633.495  função objetivo
```

que ocupará uma área de:

```
----      270 VARIABLE maior_X.L              =           49.100
----      270 VARIABLE maior_Y.L              =           12.950
```

A figura 30 apresenta a alocação dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (`c1`, `c2`, `c3`, `c4`, `c5`, `c6`, `c7` e `c8`). A figura 29 apresenta as coordenadas da mediana dos postos na solução obtida (identificados pela linha *l* e a posição *c*).

A figura 32 apresenta o resumo gerado pelo *solver* GAMS/CPLEX. A linha “RESOURCE USAGE, LIMIT” apresenta, respectivamente, o tempo de CPU (em segundos) utilizado e o limite máximo definido. Neste caso, o limite era 50.000 segundos, mas a busca durou 58823,745 segundos, devido ao critério de parada do CPLEX.

```

---- 270 VARIABLE ordem_pc.L

      c1      c2      c3      c4      c6      c7(Eixo Y)
(Eixo X)
maq1 .12                1.000
maq2 .11              1.000
maq3 .11                1.000
maq4 .11      1.000
maq6 .11                1.000
maq8 .12                1.000
maq9 .12              1.000
maq10.12             1.000
maq11.12      1.000
maq12.13             1.000
maq13.13      1.000
maq14.13                1.000
maq15.14                1.000
maq17.14             1.000
maq19.13                1.000
maq21.13          1.000

+      c8      c9      c10     c11

maq5 .11              1.000
maq7 .12                1.000
maq16.12     1.000
maq18.12             1.000
maq20.13             1.000
maq22.13                1.000

```

Figura 30. Alocação encontrada dos postos de trabalho.

```

---- 270 VARIABLE pos_X.L mediana do posto m

      c1      c2      c3      c4      c5      c6
11    4.550   14.200   24.400   30.500   31.500   32.500
12    4.550   14.200   24.400   30.500   31.500   32.500
13    4.550   14.200   24.400   30.500   31.500   32.500
14    4.550   14.200   24.400   30.500   31.500   32.500

+      c7      c8      c9      c10     c11
11    35.700   38.050   40.400   43.850   48.100
12    35.700   38.050   40.400   43.850   48.100
13    35.700   38.050   40.400   43.850   48.100
14    35.700   38.050   40.400   43.850   48.100

---- 270 VARIABLE pos_Y.L mediana do posto m

      c1      c2      c3      c4      c5      c6
11    1.300   1.300   1.300   1.300   1.300   1.300
12    4.750   4.750   4.750   4.750   4.750   4.750
13    8.900   8.900   8.900   8.900   8.900   8.900
14   11.800   11.800   11.800   11.800   11.800   11.800

+      c7      c8      c9      c10     c11
11    1.300   1.300   1.300   1.300   1.300
12    4.750   4.750   4.750   4.750   4.750
13    8.900   8.900   8.900   8.900   8.900
14   11.800   11.800   11.800   11.800   11.800

```

Figura 31. Coordenadas das medianas dos postos na solução encontrada.

Um aspecto importante a se chamar a atenção na figura 32 foi que o *gap* de integralidade (diferença entre a estimativa da melhor solução que pode ser encontrada e a melhor solução que foi encontrada). Neste caso, o *gap* de integralidade foi de 99,99% (Relative gap: 0.999949), indicando a dificuldade de o modelo matemático “guiar” a sua busca.

```

GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows 09/18/13 23:33:53 Page 5
Layout de máquinas
Solution Report SOLVE layout Using MIP From line 268

          S O L V E          S U M M A R Y

MODEL    layout                OBJECTIVE total
TYPE     MIP                    DIRECTION MINIMIZE
SOLVER   CPLEX                  FROM LINE 268

**** SOLVER STATUS      3 Resource Interrupt
**** MODEL STATUS      8 Integer Solution
**** OBJECTIVE VALUE    875633.4947

RESOURCE USAGE, LIMIT    30009.534    30000.000

Resource limit exceeded.
MIP Solution:            875633.507906    (691315 iterations, 1485 nodes)
Final Solve:            875633.494665    (1424 iterations)

Best possible:           27.849937
Absolute gap:            875605.657969
Relative gap:            0.999968

```

Figura 32. Resumo do relatório gerado pelo GAMS.

- **Grid de cinco linhas – PLIM**

A solução obtida foi (valor da função objetivo)::

```

---- 266 VARIABLE total.L = 687884.863 função objetivo

```

que ocupará uma área de:

```

---- 266 VARIABLE maior_X.L = 29.725
---- 266 VARIABLE maior_Y.L = 7.650

```

A figura 33 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 34 apresenta as coordenadas das medianas dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

A figura 35 apresenta o resumo gerado pelo *solver* GAMS/CPLEX. A linha “RESOURCE USAGE, LIMIT” apresenta, respectivamente, o tempo de CPU (em segundos) utilizado e o limite máximo definido. Neste caso, o limite era 50.000 segundos, mas a busca durou 58823,745 segundos, devido ao critério de parada do CPLEX.

É relevante mencionar, na figura 35, o *gap* de integralidade (diferença entre a estimativa da melhor solução e a melhor solução efetivamente encontrada), que foi de 99,99% (Relative gap: 0.999949), indicando a dificuldade do modelo matemático para “guiar” a sua busca.

```

GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows          09/13/13 12:31:06 Page 4
Layout de máquinas
Model Statistics      SOLVE layout Using MIP From line 264

MODEL STATISTICS

BLOCKS OF EQUATIONS      21      SINGLE EQUATIONS      169,461
BLOCKS OF VARIABLES     13      SINGLE VARIABLES      8,735
NON ZERO ELEMENTS      944,928      DISCRETE VARIABLES      2,420

              S O L V E      S U M M A R Y

MODEL  layout      OBJECTIVE  total
TYPE   MIP          DIRECTION  MINIMIZE
SOLVER CPLEX        FROM LINE  264

**** SOLVER STATUS      3 Resource Interrupt
**** MODEL STATUS      8 Integer Solution
**** OBJECTIVE VALUE      687884.8633

RESOURCE USAGE, LIMIT      10802.421      10800.000

Resource limit exceeded.

MIP Solution:      687884.863258      (1007741 iterations, 483 nodes)
Final Solve:      687884.863256      (3562 iterations)

Best possible:      19.617762
Absolute gap:      687865.245495
Relative gap:      0.999971

```

Figura 35. Resumo do relatório gerado pelo GAMS.

b) Caso 2 (12 postos)

- **Grid de quatro linhas – PLIM**

A solução obtida foi (valor da função objetivo):

```
----      265 VARIABLE total.L                =      320985.300  função objetivo
```

que ocupará uma área de:

```
----      265 VARIABLE maior_X.L              =              41.900
----      265 VARIABLE maior_Y.L              =              9.900
```

A figura 36 apresenta a alocação dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 37 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

```

---- 265 VARIABLE ordem_pc.L
      c1          c2          c3          c4          c5          c6 (Eixo Y)
(Eixo X)
maq1 .12          1.000
maq2 .12
maq3 .12          1.000
maq5 .14          1.000
maq9 .13          1.000
maq10.13         1.000
maq12.13
maq14.12          1.000
maq16.11          1.000
maq18.11
maq19.11          1.000
maq20.11          1.000
maq21.14          1.000
maq22.14          1.000
+      c7          c8          c9          c10         c11
maq4 .14
maq6 .12          1.000
maq7 .12          1.000
maq8 .12          1.000
maq11.12         1.000
maq13.12         1.000
maq15.11         1.000
maq17.11         1.000

```

Figura 36. Alocação encontrada dos postos de trabalho.

```

---- 265 VARIABLE pos_X.L mediana do posto m
      c1          c2          c3          c4          c5          c6
11    13.425     15.775     15.775     16.775     18.775     23.225
12    12.000     13.000     15.000     18.200     20.400     21.400
13     5.100     15.300     20.400     20.400     20.400     24.950
14    11.450     12.450     13.450     13.450     17.000     21.350
+      c7          c8          c9          c10         c11
11    26.675     26.675     29.025     31.375     32.375
12    23.400     25.400     27.400     30.600     37.350
13    29.500     29.500     29.500     29.500     29.500
14    22.150     22.150     22.150     22.150     23.150
---- 265 VARIABLE pos_Y.L mediana do posto m
      c1          c2          c3          c4          c5          c6
11     1.600     1.600     1.600     1.600     1.600     1.600
12     4.500     4.500     4.500     4.500     4.500     4.500
13     8.250     8.250     8.250     8.250     8.250     8.250
14     8.250     8.250     8.250     8.250     8.250     8.250
+      c7          c8          c9          c10         c11
11     1.600     1.600     1.600     1.600     1.600
12     4.500     4.500     4.500     4.500     4.500
13     8.250     8.250     8.250     8.250     8.250
14     8.250     8.250     8.250     8.250     8.250

```

Figura 37. Coordenadas das medianas dos postos na solução encontrada.

A figura 38 apresenta o resumo gerado pelo *solver* GAMS/CPLEX. A linha “RESOURCE USAGE, LIMIT” apresenta, respectivamente, o tempo de CPU (em segundos) utilizado e o limite máximo definido. Neste caso, o limite era 30.000 segundos, mas a busca durou 10914,84 segundos, devido ao critério de parada do CPLEX.

Um aspecto importante a se chamar a atenção na figura 38 foi que o *gap* de integralidade (diferença entre a estimativa da melhor solução que pode ser encontrada e a melhor solução que foi encontrada). Neste caso, o *gap* de integralidade foi de 99,99% (Relative gap: 0.999924), indicando a dificuldade de o modelo matemático “guiar” a sua busca.

```
GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows 09/20/13 01:38:08 Page 5
Layout de máquinas
Solution Report SOLVE layout Using MIP From line 269

          S O L V E          S U M M A R Y

MODEL    layout              OBJECTIVE total
TYPE     MIP                 DIRECTION MINIMIZE
SOLVER   CPLEX              FROM LINE 269

**** SOLVER STATUS          3 Resource Interrupt
**** MODEL STATUS          8 Integer Solution
**** OBJECTIVE VALUE       1103645.0332

RESOURCE USAGE, LIMIT      30013.266    30000.000

Resource limit exceeded.

MIP Solution:              1103645.036001    (1116110 iterations, 1260 nodes)
Final Solve:               1103645.033174    (1518 iterations)

Best possible:              27.686850
Absolute gap:               1103617.349151
Relative gap:               0.999975
```

Figura 38. Resumo do relatório gerado pelo GAMS.

- **Grid de cinco linhas – PLIM**

A solução obtida foi (valor da função objetivo)::

```
---- 271 VARIABLE total.L = 1103645.033 função objetivo
```

que ocupará uma área de:

```
---- 265 VARIABLE maior_X.L = 30.400
```

```
---- 265 VARIABLE maior_Y.L = 15.850
```

A figura 39 apresenta a alocação ótima dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 40 apresenta as coordenadas da mediana dos postos na solução ótima (identificados pela linha / e a posição c).

```

---- 265 VARIABLE ordem_pc.L

(Eixo X)
maq3 .15
maq4 .14
maq6 .12
maq7 .12
maq8 .13
maq9 .15
maq10.13
maq11.14
maq13.13
maq15.13
maq16.11
maq17.14
maq18.12
maq19.11
maq20.11
maq21.12
maq22.12

+
c10
c11

maq1 .14
maq2 .15
maq5 .12
maq12.13
maq14.12

```

	c1	c2	c3	c4	c8	c9 (Eixo Y)
maq3 .15						1.000
maq4 .14			1.000			
maq6 .12						1.000
maq7 .12				1.000		
maq8 .13			1.000			
maq9 .15	1.000					
maq10.13				1.000		
maq11.14	1.000					
maq13.13	1.000					
maq15.13		1.000				
maq16.11						1.000
maq17.14		1.000				
maq18.12			1.000			
maq19.11				1.000		
maq20.11	1.000					
maq21.12		1.000				
maq22.12					1.000	
+						
c10						
c11						
maq1 .14	1.000					
maq2 .15		1.000				
maq5 .12	1.000					
maq12.13		1.000				
maq14.12		1.000				

Figura 39. Alocação encontrada dos postos de trabalho.

A figura 41 apresenta o resumo gerado pelo *solver* GAMS/CPLEX. A linha “RESOURCE USAGE, LIMIT” apresenta, respectivamente, o tempo de CPU (em segundos) utilizado e o limite máximo definido. Neste caso, o limite era 10.800 segundos, mas a busca durou 10805,4 segundos, devido ao critério de parada do CPLEX.

É interessante mencionar que, conforme figura 41, o *gap* de integralidade foi de 99,99% (Relative gap: 0.999926). Isso indica a dificuldade do modelo matemático para “guiar” a sua busca.

----	271 VARIABLE pos_X.L mediana do posto m					
	c1	c2	c3	c4	c5	c6
11	4.550	10.100	11.100	14.550	19.100	21.300
12	4.550	10.100	11.100	14.550	19.100	21.300
13	4.550	10.100	11.100	14.550	19.100	21.300
14	4.550	10.100	11.100	14.550	19.100	21.300
+	c7	c8	c9	c10	c11	
11	22.300	23.300	28.400	33.500	35.700	
12	22.300	23.300	28.400	33.500	35.700	
13	22.300	23.300	28.400	33.500	35.700	
14	22.300	23.300	28.400	33.500	35.700	
----	265 VARIABLE pos_Y.L mediana do posto m					
	c1	c2	c3	c4	c5	c6
11	2.150	2.150	2.150	2.150	2.150	2.150
12	6.300	6.300	6.300	6.300	6.300	6.300
13	8.300	8.300	8.300	8.300	8.300	8.300
14	9.300	9.300	9.300	9.300	9.300	9.300
15	10.300	10.300	10.300	10.300	10.300	10.300
+	c7	c8	c9	c10	c11	
11	2.150	2.150	2.150	2.150	2.150	
12	6.300	6.300	6.300	6.300	6.300	
13	8.300	8.300	8.300	8.300	8.300	
14	9.300	9.300	9.300	9.300	9.300	
15	10.300	10.300	10.300	10.300	10.300	

Figura 40. Coordenada da mediana dos postos na solução ótima.

```

GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows          09/13/13 13:10:36 Page 4
Layout de máquinas
Model Statistics      SOLVE layout Using MIP From line 263

MODEL STATISTICS

BLOCKS OF EQUATIONS      21      SINGLE EQUATIONS      168,911
BLOCKS OF VARIABLES     13      SINGLE VARIABLES      8,680
NON ZERO ELEMENTS      948,998      DISCRETE VARIABLES    2,420

          S O L V E      S U M M A R Y

MODEL  layout          OBJECTIVE total
TYPE   MIP             DIRECTION MINIMIZE
SOLVER CPLEX          FROM LINE 263

**** SOLVER STATUS      3 Resource Interrupt
**** MODEL STATUS      8 Integer Solution
**** OBJECTIVE VALUE    264699.4637

RESOURCE USAGE, LIMIT    10805.400    10800.000

Resource limit exceeded.

MIP Solution:            264699.463683    (1031425 iterations, 312 nodes)
Final Solve:            264699.463693    (4391 iterations)

Best possible:          19.617762
Absolute gap:          264679.845921
Relative gap:           0.999926

```

Figura 41. Resumo do relatório gerado pelo GAMS.

c) Caso 3 (Problema *layout*)

- **Grid de quatro linhas – PLIM**

A solução obtida foi (valor da função objetivo)::

```
----      266 VARIABLE total.L                =      482937.400  função objetivo
```

que ocupará uma área de:

```
----      266 VARIABLE maior_X.L              =              30.950
----      266 VARIABLE maior_Y.L              =              10.950
```

A figura 42 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 43 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

---- 266 VARIABLE ordem_pc.L		c1	c2	c4	c5	c6	c8 (Eixo Y)
(Eixo X)							
maq1 .14				1.000			
maq2 .14		1.000					
maq3 .13							1.000
maq4 .13						1.000	
maq5 .12					1.000		
maq7 .12						1.000	
maq9 .13		1.000					
maq11.12		1.000					
maq13.11					1.000		
maq15.11			1.000				
maq17.11		1.000					
maq19.14						1.000	
maq20.14					1.000		
	+	c9	c10	c11			
maq6 .11			1.000				
maq8 .11		1.000					
maq10.12		1.000					
maq12.13		1.000					
maq14.13				1.000			
maq16.14				1.000			
maq18.14			1.000				
maq21.12				1.000			
maq22.11				1.000			

Figura 42. Alocação encontrada dos postos de trabalho.

---- 266 VARIABLE pos_X.L mediana do posto m						
	c1	c2	c3	c4	c5	c6
11	3.950	7.300	9.650	9.650	11.850	14.050
12	5.100	9.650	9.650	9.650	10.650	12.650
13	5.100	10.200	10.200	10.200	10.200	11.200
14	5.100	8.200	8.200	9.200	15.700	20.150
+	c7	c8	c9	c10	c11	
11	14.050	14.050	18.250	21.700	26.000	
12	13.650	13.650	18.750	23.850	27.400	
13	12.200	13.200	18.750	23.300	25.500	
14	21.150	21.150	21.150	22.150	25.500	
---- 266 VARIABLE pos_Y.L mediana do posto m						
	c1	c2	c3	c4	c5	c6
11	1.750	1.750	1.750	1.750	1.750	1.750
12	1.750	1.750	1.750	1.750	1.750	1.750
13	5.900	5.900	5.900	5.900	5.900	5.900
14	9.350	9.350	9.350	9.350	9.350	9.350
+	c7	c8	c9	c10	c11	
11	1.750	1.750	1.750	1.750	1.750	
12	1.750	1.750	1.750	1.750	1.750	
13	5.900	5.900	5.900	5.900	5.900	
14	9.350	9.350	9.350	9.350	9.350	

Figura 43. Coordenada da mediana dos postos na solução encontrada.

A figura 44 apresenta o resumo gerado pelo *solver* GAMS/CPLEX. A linha “RESOURCE USAGE, LIMIT” apresenta, respectivamente, o tempo de CPU (em segundos) utilizado e o limite máximo definido. Neste caso, o limite era 50.000 segundos, mas a busca durou 58823.745 segundos, devido ao critério de parada do CPLEX.

No caso da figura 44, o *gap* de integralidade foi de 99,99% (Relative gap: 0.999949).

```

GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows          09/06/13 15:28:40 Page 5
Layout de máquinas
Solution Report      SOLVE layout Using MIP From line 264

                S O L V E      S U M M A R Y

MODEL  layout                OBJECTIVE  total
TYPE   MIP                   DIRECTION MINIMIZE
SOLVER CPLEX                 FROM LINE 264

**** SOLVER STATUS      3 Resource Interrupt
**** MODEL STATUS      8 Integer Solution
**** OBJECTIVE VALUE   482937.4000

RESOURCE USAGE, LIMIT      58823.745      50000.000

Resource limit exceeded.

MIP Solution:      482937.400001      (2866988 iterations, 5582 nodes)
Final Solve:      482937.400000      (3368 iterations)

Best possible:      24.535915
Absolute gap:      482912.864085
Relative gap:      0.999949

```

Figura 44. Resumo do relatório gerado pelo GAMS.

- **Grid de cinco linhas – PLIM**

A solução obtida foi (valor da função objetivo)::

```

----      266 VARIABLE total.L                =      608991.678  função objetivo

```

que ocupará uma área de:

```

----      266 VARIABLE maior_X.L              =      24.962
----      266 VARIABLE maior_Y.L              =      4.300

```

A figura 45 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 46 apresenta as coordenadas da mediana dos postos na solução ótima (identificados pela linha *l* e a posição *c*).

```

---- 266 VARIABLE ordem_pc.L

      c1          c2          c3          c4          c5          c6 (Eixo Y)
(Eixo X)
maq3 .14                                1.000
maq4 .14          1.000
maq5 .12          1.000
maq7 .12                                1.000
maq10.11                             1.000
maq11.15                                1.000
maq12.11          1.000
maq13.14                                1.000
maq14.12                                1.000
maq16.13                                1.000
maq18.13                                1.000
maq19.13                                1.000
maq20.14                                1.000
maq21.12                                1.000
maq22.13          1.000

      +          c7          c8          c9          c10

maq1 .14                                1.000
maq2 .14          1.000
maq6 .12                                1.000
maq8 .12          1.000
maq9 .15          1.000
maq15.13                                1.000
maq17.13          1.000

```

Figura 45. Alocação encontrada dos postos de trabalho.

```

---- 266 VARIABLE pos_X.L mediana do posto m

      c1          c2          c3          c4          c5          c6
11      3.950      7.300      9.650      9.650      11.850      14.050
12      5.100      9.650      9.650      9.650      10.650      12.650
13      5.100      10.200     10.200     10.200     10.200      11.200
14      5.100      8.200      8.200      9.200     15.700     20.150

      +          c7          c8          c9          c10          c11
11      14.050     14.050     18.250     21.700     26.000
12      13.650     13.650     18.750     23.850     27.400
13      12.200     13.200     18.750     23.300     25.500
14      21.150     21.150     21.150     22.150     25.500

---- 266 VARIABLE pos_Y.L mediana do posto m

      c1          c2          c3          c4          c5          c6
11      1.750      1.750      1.750      1.750      1.750      1.750
12      1.750      1.750      1.750      1.750      1.750      1.750
13      5.900      5.900      5.900      5.900      5.900      5.900
14      9.350      9.350      9.350      9.350      9.350      9.350

      +          c7          c8          c9          c10          c11
11      1.750      1.750      1.750      1.750      1.750
12      1.750      1.750      1.750      1.750      1.750
13      5.900      5.900      5.900      5.900      5.900
14      9.350      9.350      9.350      9.350      9.350

```

Figura 46. Coordenadas das medianas dos postos na solução encontrada.

Conforme Figura 47, o tempo limite de CPU (em segundos) utilizado e o limite máximo definido foram, respectivamente, 58496.114 e 50000 segundos, devido ao critério de parada do CPLEX. Nesta figura, o *gap* de integralidade 99,99% (Relative gap: 0.999968).

```

GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows 09/06/13 15:33:46 Page 4
Layout de máquinas
Model Statistics SOLVE layout Using MIP From line 264

MODEL STATISTICS

BLOCKS OF EQUATIONS      21      SINGLE EQUATIONS      169,461
BLOCKS OF VARIABLES      13      SINGLE VARIABLES      8,735
NON ZERO ELEMENTS      944,928      DISCRETE VARIABLES      2,420

GENERATION TIME      =      2.750 SECONDS      54 Mb WIN233-233 Dec 15, 2009

EXECUTION TIME      =      2.750 SECONDS      54 Mb WIN233-233 Dec 15, 2009
GAMS Rev 233 WEX-VIS 23.3.3 x86/MS Windows 09/06/13 15:33:46 Page 5
Layout de máquinas
Solution Report SOLVE layout Using MIP From line 264

                S O L V E      S U M M A R Y

MODEL layout      OBJECTIVE total
TYPE MIP          DIRECTION MINIMIZE
SOLVER CPLEX      FROM LINE 264

**** SOLVER STATUS      3 Resource Interrupt
**** MODEL STATUS      8 Integer Solution
**** OBJECTIVE VALUE      608991.6780

RESOURCE USAGE, LIMIT      58496.114      50000.000

Resource limit exceeded.

MIP Solution:      608992.132581      (1968400 iterations, 609 nodes)
Final Solve:      608991.678012      (4353 iterations)

Best possible:      19.617762
Absolute gap:      608972.514818
Relative gap:      0.999968

```

Figura 47. Resumo do relatório gerado pelo GAMS.

4.4.2 Soluções Obtidas pela Abordagem BT Híbrida

Nesta seção serão apresentadas as soluções do modelo da seção 3.2.2, que representa uma abordagem BT híbrida. Foram geradas duas soluções para cada caso, considerando-se a possibilidade de organizar o *grid* de linhas em quatro e, também em cinco linhas. Ao contrário da seção anterior, não será apresentado o resumo do GAMS/CPLEX, já que a BT foi implementada dentro deste *solver*, executando-o inúmeras vezes. De fato, o relatório de cada caso tem, aproximadamente, 1 Gigabyte. Os tempos de execução para 100 iterações foram de duas horas.

a) Caso 1 (13 postos)

- **Grid de quatro linhas – BT híbrida**

A solução obtida foi (valor da função objetivo)::

```
----      526 PARAMETER tot_otimo              =      448406.700
```

que ocupará uma área de:

```
----      526 PARAMETER maior_X_otimo          =          42.150
          PARAMETER maior_Y_otimo            =          11.050
```

A figura 48 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 49 apresenta as coordenadas da mediana dos postos na solução ótima (identificados pela linha *l* e a posição *c*).

```

---- 526 PARAMETER otimo
      c1          c2          c3          c4          c5          c6 (EixoY)
(Eixo X)
maq1 .12                1.000
maq2 .13                1.000
maq4 .12          1.000
maq6 .12                1.000
maq9 .13                1.000
maq10.13             1.000
maq11.14             1.000
maq12.14          1.000
maq13.14                1.000
maq14.14             1.000
maq16.13                1.000
maq18.12                1.000
maq19.13          1.000
maq20.12                1.000
maq21.12                1.000
maq22.13                1.000
      +          c7          c8          c9
maq3 .12          1.000
maq5 .12                1.000
maq7 .12                1.000
maq8 .13          1.000
maq15.13             1.000
maq17.13             1.000

```

Figura 48 Alocação encontrada dos postos de trabalho.

```

---- 526 PARAMETER pos_X_otimo mediana do posto m
      c1          c2          c3          c4          c5          c6
11      5.100      5.100
12     28.700      5.100     30.700     23.250     18.800     38.600
13     19.900     30.700      7.100     23.250     36.800     38.600
14     30.700     23.250      7.100     14.550      5.100      5.100
      +          c7          c8          c9
12     32.700      9.100      7.100
13      1.000     14.550     17.900
14      5.100

---- 526 PARAMETER pos_Y_otimo mediana do posto m
      c1          c2          c3          c4          c5          c6
11      2.150      1.650
12      9.750      9.600      9.750      7.300      7.300      7.300
13      4.900      6.300      6.150      5.050      6.300      4.650
14      2.150      2.150      2.000      2.000      2.150      1.000
      +          c7          c8          c9
12      9.750      9.600      9.600
13      6.150      4.900      4.900
14      1.000

```

Figura 49. Coordenadas das medianas dos postos na solução encontrada.

- **Grid de cinco linhas**

A solução obtida foi (valor da função objetivo)::

```
---- 532 PARAMETER tot_otimo = 454380.050
```

que ocupará uma área de:

```
---- 526 PARAMETER maior_X_otimo = 36.900
      PARAMETER maior_Y_otimo = 17.150
```

A figura 50 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 51 apresenta as coordenadas da mediana dos postos na solução obtida (identificados pela linha / e a posição c).

---- 526 PARAMETER otimo						
(Eixo X)	c1	c2	c3	c4	c5	c6 (Eixo Y)
maq1 .12					1.000	
maq2 .12		1.000				
maq3 .13	1.000					
maq4 .12						1.000
maq5 .11		1.000				
maq6 .11			1.000			
maq7 .11	1.000					
maq8 .12				1.000		
maq9 .12	1.000					
maq10.13		1.000				
maq11.13					1.000	
maq12.14	1.000					
maq13.14				1.000		
maq14.15	1.000					
maq15.14					1.000	
maq16.14		1.000				
maq17.14						1.000
maq18.13			1.000			
maq19.14			1.000			
maq20.13				1.000		
maq22.13						1.000
	+	c7				
maq21.14	1.000					

Figura 50. Alocação encontrada dos postos de trabalho.

---- 526 PARAMETER pos_X_otimo mediana do posto m						
	c1	c2	c3	c4	c5	c6
11	3.100	5.100	7.100	5.100	5.100	
12	5.100	27.250		11.200	25.250	23.250
13	31.350	25.250	18.350	13.900	5.100	33.350
14	25.250	18.350	15.000	5.100	9.650	13.000
15	18.500	5.100				
+ c7						
12	5.100					
13	5.150					
14	33.350					
---- 526 PARAMETER pos_Y_otimo mediana do posto m						
	c1	c2	c3	c4	c5	c6
11	15.850	15.850	15.850	1.650	1.650	
12	12.400	10.350	9.550	12.400	10.350	10.350
13	6.900	6.900	6.900	6.900	8.250	8.550
14	2.750	4.650	4.500	4.500	4.500	4.500
15	1.750	2.150				
+ c7						
12	2.150					
13	9.950					
14	5.900					

Figura 51. Coordenadas das medianas dos postos na solução encontrada.

b) Caso 2 (12 postos)

Foram geradas duas soluções usando BT híbrida para o caso 2, considerando-se a possibilidade de organizar o *grid* de linhas em quatro e, também em cinco linhas.

- **Grid de quatro linhas**

A solução obtida foi (valor da função objetivo)::

```
---- 532 PARAMETER tot_otimo = 471417.350
```

que ocupará uma área de:

```
---- 532 PARAMETER maior_X_otimo = 36.450
PARAMETER maior_Y_otimo = 14.400
```

A figura 52 apresenta a alocação encontrada dos postos de trabalho no *grid*

compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 53 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha / e a posição c).

```

---- 532 PARAMETER otimo
(Eixo X)      c1      c2      c3      c4      c5      c6 (Eixo Y)
maq3 .13      1.000
maq4 .12      1.000
maq5 .14              1.000
maq6 .14              1.000
maq7 .14              1.000
maq9 .13              1.000
maq10.13             1.000
maq11.12             1.000
maq12.14      1.000
maq13.11             1.000
maq14.14             1.000
maq15.11             1.000
maq16.13              1.000
maq17.11      1.000
maq18.12              1.000
maq19.13              1.000
maq20.12              1.000
maq21.12              1.000
maq22.13              1.000
+      c7      c8
maq1 .12              1.000
maq2 .12      1.000
maq8 .13      1.000

```

Figura 52. Alocação encontrada dos postos de trabalho.

```

532 PARAMETER pos_X_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11    13.100   9.750   5.200  34.450   5.100   5.100
12    27.000   5.100              17.550  13.100  32.900
13    31.100  25.000   5.100   17.550  13.200  32.900
14    25.000  17.550   5.100   3.100   7.100   5.100
+      c7      c8      c9      c10     c11
12    23.000  25.000   3.000  39.350  38.350
13    11.200  37.450
14     5.100
---- 532 PARAMETER pos_Y_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11    12.650  12.650  12.650   1.300   2.150   2.150
12    11.350   8.900   8.900   9.050   9.050   8.900
13     7.900   7.900   4.750   6.650   6.650   6.250
14     3.750   3.750   1.300   1.300   1.300   1.000
+      c7      c8      c9      c10     c11
12    11.350  11.350   3.750   5.500   7.150
13     4.750   4.750
14     1.000

```

Figura 53. Coordenadas das medianas dos postos na solução encontrada.

- **Grid de cinco linhas**

A solução obtida foi (valor da função objetivo)::

```
----- 532 PARAMETER tot_otimo           = 454380.050
```

que ocupará uma área de:

```
----- 526 PARAMETER maior_X_otimo       = 36.900
        PARAMETER maior_Y_otimo         = 17.150
```

A figura 52 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 53 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

```
----- 526 PARAMETER otimo
```

	c1	c2	c3	c4	c5	c6 (Eixo Y)
(Eixo X)						
maq1 .12					1.000	
maq2 .12		1.000				
maq3 .13	1.000					
maq4 .12						1.000
maq5 .11		1.000				
maq6 .11			1.000			
maq7 .11	1.000					
maq8 .12				1.000		
maq9 .12	1.000					
maq10.13		1.000				
maq11.13					1.000	
maq12.14	1.000					
maq13.14				1.000		
maq14.15	1.000					
maq15.14					1.000	
maq16.14		1.000				
maq17.14						1.000
maq18.13			1.000			
maq19.14			1.000			
maq20.13				1.000		
maq22.13						1.000
	+	c7				
maq21.14	1.000					

Figura 54. Alocação encontrada dos postos de trabalho.

```

----- 526 PARAMETER pos_X_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11      3.100      5.100      7.100      5.100      5.100
12      5.100      27.250      11.200      25.250      23.250
13      31.350      25.250      18.350      13.900      5.100      33.350
14      25.250      18.350      15.000      5.100      9.650      13.000
15      18.500      5.100
+      c7
12      5.100
13      5.150
14      33.350

----- 526 PARAMETER pos_Y_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11      15.850      15.850      15.850      1.650      1.650
12      12.400      10.350      9.550      12.400      10.350      10.350
13      6.900      6.900      6.900      6.900      8.250      8.550
14      2.750      4.650      4.500      4.500      4.500      4.500
15      1.750      2.150
+      c7
12      2.150
13      9.950
14      5.900

```

Figura 55. Coordenadas das medianas dos postos na solução encontrada.

c) Caso 3 (Layout)

Foram geradas duas soluções usando BT híbrida para o caso 2, considerando-se a possibilidade de organizar o *grid* de linhas em quatro e, também em cinco linhas.

- **Grid de quatro linhas**

A solução obtida foi (valor da função objetivo)::

```

----- 532 PARAMETER tot_otimo = 471417.350

```

que ocupará uma área de:

```

----- 532 PARAMETER maior_X_otimo = 36.450
          PARAMETER maior_Y_otimo = 14.400

```

A figura 54 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 55 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

```

---- 532 PARAMETER otimo
(Eixo X)      c1      c2      c3      c4      c5      c6 (Eixo Y)
maq3 .13      1.000
maq4 .12      1.000
maq5 .14              1.000
maq6 .14              1.000
maq7 .14              1.000
maq9 .13              1.000
maq10.13             1.000
maq11.12             1.000
maq12.14      1.000
maq13.11              1.000
maq14.14             1.000
maq15.11             1.000
maq16.13              1.000
maq17.11      1.000
maq18.12              1.000
maq19.13              1.000
maq20.12              1.000
maq21.12              1.000
maq22.13              1.000
+
maq1 .12      c7      c8
maq2 .12      1.000      1.000
maq8 .13      1.000

```

Figura 56. Alocação encontrada dos postos de trabalho.

```

532 PARAMETER pos_X_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11     13.100     9.750     5.200     34.450     5.100     5.100
12     27.000     5.100              17.550     13.100     32.900
13     31.100     25.000     5.100     17.550     13.200     32.900
14     25.000     17.550     5.100     3.100     7.100     5.100
+
12     23.000     25.000     3.000     39.350     38.350
13     11.200     37.450
14     5.100
---- 532 PARAMETER pos_Y_otimo mediana do posto m
      c1      c2      c3      c4      c5      c6
11     12.650     12.650     12.650     1.300     2.150     2.150
12     11.350     8.900     8.900     9.050     9.050     8.900
13     7.900     7.900     4.750     6.650     6.650     6.250
14     3.750     3.750     1.300     1.300     1.300     1.000
+
12     11.350     11.350     3.750     5.500     7.150
13     4.750     4.750
14     1.000

```

Figura 57. Coordenadas das medianas dos postos na solução encontrada.

- **Grid de cinco linhas**

A solução obtida foi (valor da função objetivo)::

```
---- 520 PARAMETER tot_otimo = 372793.800
```

que ocupará uma área de:

```
---- 520 PARAMETER maior_X_otimo = 26.600
PARAMETER maior_Y_otimo = 16.700
```

A figura 56 apresenta a alocação encontrada dos postos de trabalho no *grid* compostos de linhas (11, 12, 13 e 14), cada uma com sua lista ordenada (c1, c2, c3, c4, c5, c6, c7 e c8). A figura 57 apresenta as coordenadas da mediana dos postos na solução encontrada (identificados pela linha *l* e a posição *c*).

---- 520 PARAMETER otimo						
	c1	c2	c3	c4	c5	c6 (Eixo Y)
(Eixo X)						
maq1 .15				1.000		
maq2 .15						1.000
maq3 .14					1.000	
maq4 .15		1.000				
maq5 .11			1.000			
maq6 .12				1.000		
maq7 .11	1.000					
maq8 .11				1.000		
maq9 .14				1.000		
maq10 .12		1.000				
maq11 .13		1.000				
maq12 .13	1.000					
maq13 .12	1.000					
maq14 .14	1.000					
maq15 .11					1.000	
maq16 .15					1.000	
maq17 .11		1.000				
maq18 .15			1.000			
maq19 .15	1.000					
maq20 .14			1.000			
maq21 .12			1.000			
maq22 .11						1.000

Figura 58. Alocação encontrada dos postos de trabalho.

A Figura 58 exibe o resumo dos resultados obtidos pela abordagem PLIM conforme já debatido, sendo verificado que a melhor solução ótima encontrada dentre todas as abordagens (valor da função objetivo) foi para o caso 1, grid de 4 linhas. Já em termos de área (considerando-se a melhor área como a melhor solução neste caso), a melhor solução obtida ocorreu para o caso 3, para o grid de 4 linhas.

A Figura 59 demonstra o resumo dos resultados conseguidos através da BT, sendo

a melhor solução ótima (função objetivo) conseguida foi obtida no caso 3, para um grid de 5 linhas; já em termos de menor área encontrada, foi conseguido no caso 3, para um grid de 5 linhas.

```

---- 520 PARAMETER pos_X_otimo mediana do posto m
      c1          c2          c3          c4          c5          c6
11    16.400      8.450      14.400      12.400      5.100      23.050
12     5.100     14.400     23.050      8.300      5.100
13    14.400      5.100       1.000     22.500
14    14.400     20.050      5.100     11.200     15.400
15    20.050      3.100     18.050      5.100     14.400      7.100

+      c7
11     5.100
15     5.100

---- 520 PARAMETER pos_Y_otimo mediana do posto m
      c1          c2          c3          c4          c5          c6
11    15.400     15.550     15.400     15.400     15.550     14.600
12    12.650     11.950     11.950     11.950      1.300
13     7.800      8.900      5.750     10.950
14     4.050      2.300      3.550      4.750      4.750     11.950
15     1.150      1.300      1.150      1.300      1.150      1.300

+      c7
11     1.300
15     1.000

```

Figura 59. Coordenada da mediana dos postos na solução encontrada.

PLIM		SOL. ÓTIMA	DIMENSÕES		ÁREA	TEMPO
Caso 1	Grid de 4 linhas	875633,495	49,100	12,950	635,845	30009,534
Caso 1	Grid de 5 linhas	1028603,150	33,500	17,650	591,275	30012,822
Caso 2	Grid de 4 linhas	1103645,033	37,900	15,200	576,080	30013,266
Caso 2	Grid de 5 linhas	1857166,100	62,800	10,300	646,840	30013,735
Caso 3	Grid de 4 linhas	1078305,564	35,650	15,050	536,533	30012,905
Caso 3	Grid de 5 linhas	999455,500	40,000	15,500	620,000	30012,455

Figura 60. Resumo das soluções obtidas pela abordagem PLIM.

BT		SOL. ÓTIMA	DIMENSÕES		ÁREA	TEMPO
Caso 1	Grid de 4 linhas	448406,700	42,150	11,050	465,758	7200,000
Caso 1	Grid de 5 linhas	454380,050	36,900	17,150	632,835	7200,000
Caso 2	Grid de 4 linhas	471417,350	36,450	14,400	524,880	7200,000
Caso 2	Grid de 5 linhas	454380,050	36,900	17,150	632,835	7200,000
Caso 3	Grid de 4 linhas	471417,350	36,450	14,400	524,880	7200,000
Caso 3	Grid de 5 linhas	372793,800	26,600	16,700	444,220	7200,000

Figura 61. Resumo das soluções obtidas pela abordagem BT.

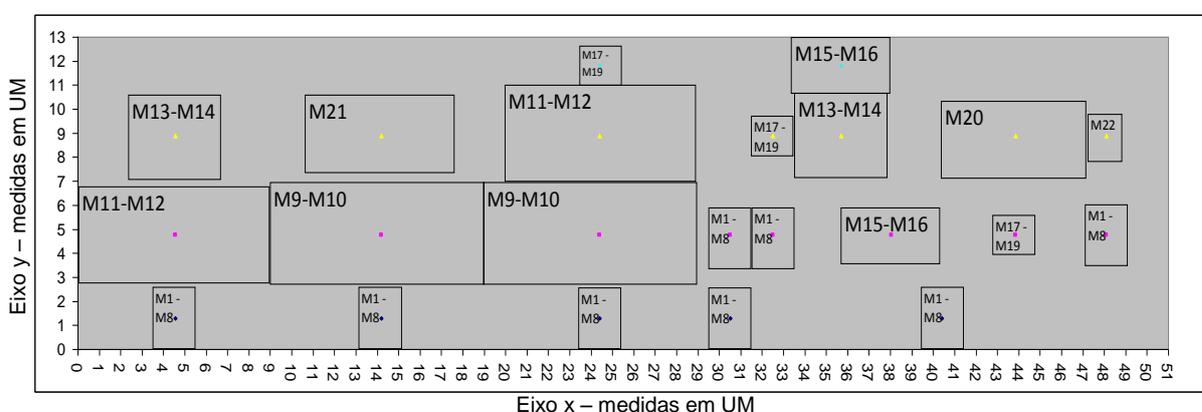


Figura 62. Layout resultante, para o caso 1, 4 linhas, abordagem PLIM.

Já a Figura 60 demonstra, de forma gráfica, o *layout* resultante para uma das soluções obtidas - o caso 1, para 4 linhas, utilizando a abordagem PLIM. Esta figura foi montada a partir dos dados fornecidos pelas Figuras 30 (Alocação dos postos de trabalho) e 31 (Coordenada da mediana dos postos na solução encontrada). Lembrando que o espaço total ocupado pelas máquinas é de 49,1 UM no eixo x, e 12,95 UM no eixo Y (área total de 635,85 UM²), o que é ainda um problema, pois a área disponível informada é de 13,7UM no Eixo y e 23,6UM no Eixo x (área total 323,32 UM²), portanto, as máquinas extrapolam o espaço disponível neste caso. Pode-se perceber na Figura 60 os resultados de análise de sobreposição (conforme Figura 14) de forma a impedir realmente que as máquinas se sobreponham, tanto no eixo X quanto no eixo Y. Outra questão importante é que a

5 CONCLUSÕES E TRABALHOS FUTUROS

O trabalho em questão visou estudar, a partir de dados reais de produção, a otimização do *layout* fabril de uma empresa de autopeças situada na cidade de Curitiba, PR, através do uso dos métodos heurísticos (busca tabu) e programação linear inteira mista. O objeto de estudo é a linha de montagem de um dos componentes mais modernos produzidos na unidade. Os volumes de fornecimento por número de tipo de parte do ano de 2011 foram levantados, assim como dimensional e quantidade de máquinas na linha de produção que é objeto deste estudo, entre outros dados relevantes (ex. plano de fabricação com sequencial de operações).

Como parte deste trabalho, foi publicado um capítulo de livro (RODRIGUES *et al.*, 2013) para formação de células. Nesta dissertação, foram geradas duas células, a partir da atribuição das máquinas existentes a estas, considerando-se aspectos de restrição de capacidade. Após a formação de células de manufatura, as peças são atribuídas às respectivas células de manufatura através da BT Híbrida e BB com poda no espaço de busca, de modo a propiciar uma busca em profundidade e diminuir o tempo computacional. A função objetivo aqui é a minimização dos movimentos extracelulares. Inclusive, o método BB (utilizado quando ocorre alocação de máquinas às células) é com certeza a principal contribuição dessa dissertação (inclusive, foi publicado em Rodrigues *et al.* (2013). Para isso, foi necessário realizar uma preparação do BB (utilizando-se o recurso da lista *EXPAND_n*), realizando uma poda para favorecer a busca em profundidade. Isso se faz necessário, pois por outro lado, caso o BB seja mal preparado, são gerados altos tempos computacionais.

Após a formação das células, na segunda parte do problema, é utilizada a BT Híbrida e Programação Linear Inteira Mista, de modo a realizar o *Layout* das máquinas. A função objetivo aqui é de minimização do custo de movimentação das peças entre operações.

O objetivo de formar as células de manufatura é, entre outros, o de dar uma “carga” menor para a formação propriamente dita do *Layout* através da tentativa de diminuir os tempos computacionais e de uma busca de maior qualidade na solução a ser encontrada.

O problema de *layout* mostrou-se bastante desafiador. A análise positiva desta dissertação é que ela foi capaz de abordar um problema real da indústria com resultados positivos. Os inúmeros casos analisados apresentam grande variação nos resultados, indicando que este é um problema de grande complexidade e de difícil convergência. Assim, como trabalho futuro, as duas abordagens de *layout* propostas deverão ser aperfeiçoadas. Já a abordagem de formação de células será validada para problemas maiores (os problemas encontrados na literatura não são “grandes”). Na sequência, será gerado um artigo para uma revista internacional. Este estudo será encaminhado para as pessoas responsáveis na empresa pela elaboração do *layout* fabril, de modo que analisem o mesmo.

Antes de apresentar os resultados, o autor e seu orientador reconhecem que mais testes devem ser feitos para validar a abordagem proposta de BT híbrida, como uma abordagem genérica para a solução do problema de Formação de Célula.

6 REVISÃO BIBLIOGRÁFICA

AARTS E., KORST J., (1989). *Simulated annealing and Boltzmann Machines*, Wiley, New York.

AHMED, N.U., (1995). *A design and implementation model for life cycle cost management system*, *Information and Management*, 28, 261-269.

AIELLO, G., LASCALIA, G., ENEA, M., (2013). *A non dominated ranking Multi Objective Genetic Algorithm and electre method for unequal area facility layout problems*, *Expert Systems with Applications* 40, 4812–4819.

ALBADAWI, Z., BASHIR, H.A., CHEN, M., (2005). *A mathematical approach for the formation of manufacturing cells*. *Computers and Industrial Engineering*, 48, 3–21.

ALHOURANI, F., SEIFODDINI, H., (2007). *Machine cell formation for production management in cellular manufacturing systems*. *International Journal of Production Research*, 45(4), 913–934.

AMARAL, A.R.S., (2006). *Discrete Optimization on the exact solution of a facility layout problem*, *European Journal of Operational Research* 173, 508–518.

AMARAL, A.R.S.. (2009). *A new lower bound for the single row facility layout problem* *Discrete Applied Mathematics* 157 , 183-190.

BROOKE, A., KENDRICK, D., MEERAUS, A., (1992). *GAMS: A user's guide. Release 2.25*, *The Scientific Press*, EUA.

BURBIDGE, J.L., (1992). *Change to group technology: Process organization is obsolete*. *International Journal of Production Research* 30 (5), 1209–1219.

CAUX, C., BRUNIAUX, R., PIERREVAL, H., (2000). *Cell formation with alternative process plans and machine capacity constraints: A new combined approach*. *Int J Production Economics*, 64:279-284.

CHIANG, W.-C., CHIANG, C. (1998). *Intelligent local search strategies for solving facility layout problems with the quadratic assignment problem formulation* *European Journal of Operational Research* 106, 457-488.

CHIANG, C.-P., LEE S.-D, (2004). *Joint determination of machine cells and linear intercell layout*, *Computers & Operations Research* 31, 1603–1619.

CHUNG, S.H., WU, T.S., CHANG, C.C , (2011). *An efficient tabu search algorithm to the cell formation problem with alternative routings and machine reliability considerations*. *Computers e Industrial Engineering* 60 7–15.

CHWIF, L., BARRETTO, M.R.P., MOSCATO, L.A., (1998). *A solution to the facility layout problem using simulated annealing*, *Computers in Industry* 36 _1998. 125–132.

CLÍMACO, R. R., (2003). Tecnologia de grupo e manufatura celular aplicadas ao projeto de leiaute industrial para pequenas e médias empresas: simplificação do fluxo de produção de uma empresa metal mecânica. XXIII Encontro Nac. de Eng. de Produção - Ouro Preto, MG, Brasil, 21 a 24 de out de 2003 - ENEGEP 2003 ABEPRO 1.

DEFERSHA, F.M., CHEN, M., (2006). *Machine cell formation using a mathematical model and a genetic-algorithm-based heuristic*. *International Journal of Production Research* 44 (12), 2421–2444.

DRIRA, A., PIERREVAL, H., HAJRI-GABOUJ, S., (2007). *Facility layout problems: A survey* *Annual Reviews in Control* 31, 255–267.

EGLESE, R.W., (1990). *Simulated annealing: a tool for operational research*, *European Journal of Operational Research*, 46 271-281.

GARCIA-HERNANDEZ, L., PIERREVAL, H., SALAS-MORERA, L., ARAUZO-AZOFRA, A., (2013). Handling qualitative aspects in Unequal Area Facility Layout Problem: An Interactive Genetic Algorithm; *Applied Soft Computing* 13, 1718–1727.

GEN, M., CHENG, R., (2000). *Genetic Algorithm and Engineering Optimization*. John Wiley and Sons. pp. 8–16.

GEORGIADIS, M.C., SCHILLING, G., ROTSTEIN, G.E., MACCHIETTO, S., (1999). *A general mathematical programming approach for process plant Layout*, *Computers and Chemical Engineering* 23, 823–840.

GLOVER, F., (1990). *Tabu search—Part II*. *ORSA Journal on Computing*, 2(1), 4–32.

GLOVER, F., LAGUNA, M., (1997). *Tabu Search*. Kluwer Academic Publisher.

GONÇALVES, J.F., RESENDE, M.G.C., (2004). *An evolutionary algorithm for manufacturing cell formation*. *Computers and Industrial Engineering* 47, 247–273.

HERAGU, S.S., GUPTA, Y.P., (1994). *A heuristic for designing cellular manufacturing facilities*. *International Journal of Production Research* 32 (1), 125–140.

HOLLAND, J.H., (1975). *Adaptation in Natural and Artificial System*, Univ. of Michigan Press, Ann Arbor, MI.

IRANI, S., (1999). *Handbook of Cellular Manufacturing Systems*. Nova York, EUA: John Wiley.

JAMES, T.L., BROWN, E.C., KEELING, K.B., (2007). *A hybrid grouping genetic algorithm for the cell formation problem*. *Computers and Operations Research* 34, 2059– 2079.

KHOO, L.P., LEE, S.G., YIN, X.F., (2003). *Multiple-objective optimization of machine cell layout using genetic algorithm*. *Int. J. Comput.-Integr. Manuf.*, 16, 140–155.

KIRPATRICK, S., GELATT, C.D., VECCHI, M.P., (1983). *Optimization by Simulated Annealing*. *Science*, 220, 671-679.

KONAK, S.K., (2012). *A linear programming embedded probabilistic tabu search for the unequal-area facility layout problem with flexible bays* , *European Journal of Operational Research* 223, 614–625.

KOTHARI, R., GHOSH, D., (2013). *Tabu search for the single row facility layout problem using exhaustive 2-opt and insertion neighborhoods*; *European Journal of Operational Research* 224, 93–100.

KOUVELIS, P., CHIANG, W.C., (1992). *A Simulated Annealing procedure for single row layout problems in flexible manufacturing systems*. *Int. J. Prod. Res.*, 30, 717-732.

KUSIAK, A., (1987). *The generalized group technology concept*. *International Journal of Production Research*, 25, 561–569.

KUSIAK, A., CHO, M., (1992). *Similarity coefficient algorithms for solving the group technology problem*. *International Journal of Production Research*, 30, 2633–2646.

LEI, D., WU, Z., (2005). *Tabu search-based approach to multi-objective machine-part cell formation*. *International Journal of Production Research*, Vol. 43, No. 24, 15, 5241–5252.

LEI, D., WU, Z., (2006). *Tabu search for multiple-criteria manufacturing cell design*. *International Journal of Advanced Manufacturing Technology* 28, 950–956.

LOGENDRAN, R., KARIM, Y., (2003). *Design of manufacturing cells in the presence of alternative cell locations and material transporters*. *Journal of the Operational Research Society* 54, 1059–1075.

LOZANO, S., CANCA, D., GUERRERO, F., GARCIA, J.M., (2001). *Machine grouping using sequence-based similarity coefficients and neural networks*. *Robotics and Computer Integrated Manufacturing* 17, 399–404.

MAK, K.L., WONG, Y.S., WANG, X.X., (2000). *An adaptive genetic algorithm for manufacturing cell formation. International Journal of Advanced Manufacturing Technology* 16, 491–497.

METROPOLIS, N., ROSENBLUTH, A. W., TELLER, A. H., (1953). *Equation of state calculations by fast computing machines. Journal of Chemical Physics*, 1087–1092.

MORABITO, R., MORALES, S., (1998). *A simple and effective recursive procedure for the manufacturer's pallet loading problem. Journal of the Operational Research Society*, 49, 819-828.

OHNO, T., (1988). *Toyota production system: beyond large-scale production, Productivity, press, ISBN 8573071702.*

OU-YANG, C.; UTAMIMA, A., (2013). *Hybrid Estimation of Distribution Algorithm for solving Single Row Facility Layout Problem. Computers & Industrial Engineering* 66, 95–103.

PAPAIIOANNOU, G., WILSON, J.M., (2008). *Fuzzy extensions to integer programming models of cell-formation problems in machine scheduling. Annals of Operations Research (Available online 10.1007/s10479-008-0423-1).*

PAPAIIOANNOU, G., WILSON, J. M., (2010). *The evolution of cell formation problem methodologies based on recent studies (1997–2008): Review and directions for future research, European Journal of Operational Research*, 206, 509-521.

PEARL, J., (1984). *Some Recent Results in Heuristic Search Theory, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-6, NO. 1, January.*

PIRLOT, M., (1996). *General local search methods, European Journal of Operational Research*, 92, 493-511.

PUREZA, V., MORABITO, R., (2003). *Uma Heurística de busca tabu simples para o problema de carregamento de paletes do produtor. Pesquisa Operacional*, v.23, n.2, p.359-378, Maio a Agosto, 359-379.

RODRIGUES, L., BALAU, A., WELLER, T., (2013). *Problema de formação de células de manufatura com roteiros alternativos e considerações de capacidade, Capítulo 27, Meta-Heurísticas em Pesquisa Operacional, Omnipax Editora.*

SAFAEI, N., SAIDI-MEHRABAD, M., TAVAKKOLI-MOGHADDAM, R., SASSANI, F., (2008). *A fuzzy programming approach for a cell formation problem with dynamic and uncertain conditions. Fuzzy Sets and Systems* 159 (2), 215–236.

SAHIN KEMAL H., CIRIC AMY R., (1998). *A dual temperature simulated annealing approach for solving bilevel programming problems*, *Computers and Chemical Engineering*, 23,11-25

SCHEITHAUER, G., TERNO, J., (1996). *The G4-heuristic for the pallet loading problem*. *Journal of the Operational Research Society*, 47, 511-522.

SELIM, H.M., ASKIN, R.G., VAKHARIA, A.J., (1998). *Cell formation in group technology: Review, evaluation and directions for future research*. *Computers and Industrial Engineering*, 34, 3–20.

Site <http://www.artigonal.com/administracao-artigos/os-sete-desperdicios-lean-manufacturing-908342.html> - acessado em 14.03.11 às 07:43.

SLACK, N., CHAMBERS, S., HARLAND, C., HARRISON, A.; JOHNSTON, R., (1999). *Administração da Produção*. Ed. Atlas S.A.

SOFIANOPOULOU, S., (1999). *Manufacturing cells design with alternative process plans and/or replicate machines*. *International Journal of Production Research* 37 (3), 707–720.

SOKE, A., BINGUL, Z. (2006). *Hybrid genetic algorithm and simulated annealing for two-dimensional non-guillotine rectangular packing problems*. *Engineering Applications of Artificial Intelligence*, 19, 557–567.

SASSANI, F., TAVAKKOLI-MOGHADDAM, R., SAFAEI, N., (2008). *A new solution for a dynamic cell formation problem with alternative routing and machine costs using simulated annealing*. *Journal of the Operational Research Society* 23, 916–924.

TALBI, E.-G., (2002). *A taxonomy of hybrid metaheuristics*. *Journal of Heuristics* 8, 541–564.

TAVAKKOLI-MOGHADDAM, R., SAFAEI, N., SASSANI, F., (2008). *A new solution for a dynamic cell formation problem with alternative routing and machine costs using simulated annealing*. *Journal of the Operational Research Society* 23, 916–924.

TOMPKINS, J.A., WHITE, J. A, (1984). *Facilities planning*. Wiley, New York.

TOMPKINS, J.A., WHITE, J.A., BOZER, Y.A., FRAZELLE, E.H., TANCHOCO, J.M.A., TREVINO, J., (1996). *Facilities Planning*, Wiley, New York.

VAKHARIA, A.J., CHANG, Y.-L., (1997). *Cell formation in group technology: a combinatorial search approach*. *International Journal of Production Research* 35 (7), 2025–2043.

VENUGOPAL, V. NARENDRAN, T.T., (1992). *A genetic algorithm approach to the machine-component grouping problem with multiple objectives. Comput. Ind. Eng.*, 22, 469–480.

WANG, T.Y., LIN, H-C, WU, K.-B. (1998). *An improved Simulated Annealing for facility layout problems in cellular manufacturing systems, Computers Ind. Eng.*, 34, 309-319.

WANG, T.Y., WU, K.B., LIU, Y.W. (2001). *A simulated annealing algorithm for facility layout problems under variable demand in Cellular Manufacturing Systems. Computers in Industry*, 46, 181-188.

WANG, M.-J., HU, M. H., KU, M.-Y., (2005). *A solution to the unequal area facilities layout problem by genetic algorithm, Computers in Industry* 56, 207–220.

WELLER, T. R., (2008). Proposta de uma abordagem de busca tabu para o layout de células de manufatura, Dissertação de Mestrado em Engenharia de Manufatura - Programa de Pós-graduação em Engenharia Mecânica e de Materiais, Universidade Tecnológica Federal do Paraná, Curitiba, 149 p.

WEMMERLOV, U., HYER, N.L., (1989). *Cell manufacturing in the US industry: a survey of users. International Journal of Production Research* 27, 1511–1530.

WON, Y. K., KIM, S.H., (1997). *Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings. Computers and Industrial Engineering*, 32, 207–220.

WONG, S.Y.W., (2001). *Hybrid simulated annealing/genetic algorithm approach to short-term hydro-thermal scheduling with multiple thermal plants. International Journal of Electrical Power and Energy Systems*, 23, 565–575.

WU, T.-H., LOW, C., WU, W.-T., (2004). *A tabu search approach to the cell formation problem. International Journal of Advanced Manufacturing Technology* 23, 916–924.

WU, X., CHU C.-H., WANG Y., YAN W., (2006). *A genetic algorithm for cellular manufacturing design and layout. European Journal of Operational Research*, 1-12.

WU, X., CHAO-HSIEN, C., WANG, Y., YAN, W., (2007). *A genetic algorithm for cellular manufacturing design and layout. European Journal of Operational Research* 181, 156–167.

WU, T.-H., CHANG, C.-C., CHUNG, S.-H., (2008). *A simulated annealing algorithm to manufacturing cell formation problems. Expert Systems with Applications*, 34, 1609–1617.

- WU, T.-H., CHUNG, S.-H., CHANG C.-C., (2009). *Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routings*, *Expert Systems with Applications*, 36, 3652-3661.
- WU, T.-H., CHUNG, S.-H., CHANG, C.-C., (2010). *A water flow-like algorithm for manufacturing cell formation problems*, *European Journal of Operational Research* 205, 346–360.
- XAMBRE, A.R., VILARINHO, P.M., (2003). *A simulated annealing approach for manufacturing cell formation with multiple identical machines*, *European Journal of Operational Research* 151, 434-446.
- YIN, Y., YASUDA, K., (2006). *Similarity coefficient methods applied to the cell formation problem: A taxonomy and review*. *International Journal of Production Economics* 101, 329–352.
- YU-HSIN CHEN, G., (2013). *A new data structure of solution representation in hybrid ant colony optimization for large dynamic facility layout problems* *Int. J. Production Economics* 142, 362–371.