



THIAGO FIALHO DE QUEIROZ LAFETÁ

**ENXAME DE PARTÍCULAS APLICADO AO
PROBLEMA GERAL DE
DIMENSIONAMENTO DE LOTES**

LAVRAS - MG

2010

THIAGO FIALHO DE QUEIROZ LAFETÁ

**EXAME DE PARTÍCULAS APLICADO AO PROBLEMA GERAL DE
DIMENSIONAMENTO DE LOTES**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Orientador:

Claudio Fabiano Motta Toledo

LAVRAS - MG

2010

THIAGO FIALHO DE QUEIROZ LAFETÁ

**EXAME DE PARTÍCULAS APLICADO AO PROBLEMA GERAL DE
DIMENSIONAMENTO DE LOTES**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

APROVADA em ____ de _____ de _____

Dr. Wilian Soares Lacerda UFLA

Dr. Raphael Winckler de Bettio UFLA

Claudio Fabiano Motta Toledo

Orientador

**LAVRAS - MG
2010**

RESUMO

O presente trabalho propõe aplicar uma metaheurística para solucionar o Problema Geral de Dimensionamento de Lotes (PGDL). Trata-se de um problema de dimensionamento de lotes monoestágio, com o objetivo de determinar a quantidade de um produto a ser produzido em diferentes períodos de tempo. O objetivo é encontrar uma configuração otimizada para o PGDL que minimize a soma dos custos de produção, preparação e estoque. A metaheurística escolhida foi a Otimização por Enxame de Partículas (OEP). O OEP tem como característica fazer com que as partículas vasculhem amplas áreas do espaço de soluções, influenciando umas as outras. Assim, o que há de melhor em cada partícula é mantido durante o processo de busca, conduzindo o enxame a sempre se mover para uma melhor região dentro do espaço de soluções do problema. A metaheurística será aplicada ao PGDL com máquinas simples e máquinas paralelas, considerando penalização para demandas não atendidas. Um conjunto de instâncias é selecionado para os testes computacionais e os resultados obtidos pelo OEP serão comparados aos obtidos por outras metaheurísticas.

Palavras-Chave: Otimização, Enxame de Partícula, Dimensionamento de Lotes.

ABSTRACT

The present work proposes to apply metaheuristic to solve the General Lot Sizing and Scheduling Problem (GLSP). It is a mono-stage lot sizing and scheduling problem that aims to determine products amount to be produced in different time periods. The objective is to find an optimized configuration for GLSP that minimizes production, setup and inventory costs. The metaheuristic chosen is a Particle Swarm Optimization (PSO). The PSO issue is to make particles search large solution space areas, changing information. Therefore, the best aspect in each particle is kept during the search process, guiding the swarm always to move for better regions on problem solution space. The metaheuristic is applied on GLSP with single and parallel machines, taking into account penalties for demands not satisfied. A set of instances is selected for computational tests and the results found for PSO are compared with those found by other metaheuristics.

Keywords: *Optimization, Particle Swarm, Lot sizing.*

LISTA DE FIGURAS

Figura 1: Modelo de sistema. Fonte: Tubino (2000)	14
Figura 2: Processo de Planejamento. Fontes: Côrrea (2004) e Tubino (2000) ..	17
Figura 3: Pseudocódigo OEP.....	22
Figura 4: Movimento de uma partícula.....	23
Figura 5: Estrutura do Horizonte de planejamento.....	26
Figura 6: Modelo matemático do PGLDPP.....	28
Figura 7: Representação da solução.	30
Figura 8: Exemplo de codificação.	31
Figura 9: Pseudo-código Método de Decodificação.....	33
Figura 10: Atualizar X_{ljt} , K_{lt} , D_j	34
Figura 11: Pseudocódigo para OEP.....	35
Figura 12: Pseudocódigo para o método Oscila.	35
Figura 13: Pseudocódigo para recombinação.	36
Figura 14: Possível contribuição de cada vetor	37
Figura 15: Exemplo de movimento de uma partícula.....	38
Figura 16: Desvio médio em S1, S2, S3 e S4.....	46
Figura 17: Desvio médio em P1, P2 e P3.	49

LISTA DE TABELAS

Tabela 1: Conjunto de instâncias para o PGDLPP com uma única máquina. ...	39
Tabela 2: Conjunto de instâncias para o PGDLPP com mais de uma máquina.	40
Tabela 3: Resultados obtidos para o PGDLPP com uma única máquina em S1.	42
Tabela 4: Resultados obtidos para o PGDLPP com uma única máquina em S2.	43
Tabela 5: Resultados obtidos para o PGDLPP com uma única máquina em S3.	43
Tabela 6: Resultados obtidos para o PGDLPP com uma única máquina em S4.	44
Tabela 7: Resultados obtidos para o PGDLPP com máquinas paralelas em P1.	47
Tabela 8: Resultados obtidos para o PGDLPP com máquinas paralelas em P2.	47
Tabela 9: Resultados obtidos em P3.	48

LISTA DE SIGLAS

PGDL:	Problema Geral de Dimensionamento de Lotes.
OEP:	Otimização por Enxame de Partículas.
PGDLPP:	Problema Geral de Direcionamento de Lotes e Programação da Produção.
PCP:	Planejamento e Controle da Produção.
AG:	Algoritmo Genético.
BT:	Busca Tabu.
OCF:	Otimização por Colônia de Formiga.
SA:	Simulated Annealing.
L:	Quantidade de linhas.
T:	Macro-períodos.
S:	Micro-períodos.
J:	Produtos.
Ct:	Capacidade da máquina.

SUMÁRIO

1. INTRODUÇÃO	10
1.1. Contextualização.....	10
1.2. Metodologia Científica	11
1.3. Objetivos do Trabalho	12
2. REFERENCIAL TEÓRICO	13
2.1. Problemas de Produção e Dimensionamento de lotes	13
2.2. Heurísticas e Metaheurísticas	20
2.3. Otimização por Enxame de Partículas (OEP).....	21
3. MATERIAIS E MÉTODOS	25
3.1. Problema Geral de Dimensionamento de lotes e Programação da Produção	25
3.2. Otimização por Exame de Partículas (OEP).....	30
3.2.1. Representação e Avaliação das soluções	30
3.2.2. Algoritmo Otimização por Enxame de Partículas (OEP)	34
4. RESULTADOS COMPUTACIONAIS	39
4.1. Parâmetros do Problema	39
4.2. Resultados para máquinas simples.....	40
4.3. Resultados para máquinas paralelas.....	46
5. CONCLUSÃO	50
6. REFERÊNCIAS BIBLIOGRÁFICAS.....	52

1. INTRODUÇÃO

1.1. Contextualização

Hoje em dia empresas competitivas possuem um sistema de produção que viabiliza a redução de custos, trazendo também flexibilidade em relação ao cliente e aumento da produtividade. O desafio que surge é conciliar maior flexibilidade com redução de custos e ofertar produtos de alta qualidade (Zattar, 2004). O planejamento bem estruturado e o total controle do sistema produtivo são fundamentais em um sistema produtivo lucrativo e eficiente.

O Problema Geral de Dimensionamento de Lotes e Programação da Produção (PGDLPP) tem como finalidade estabelecer o planejamento da produção de itens em um número limitado de máquinas ao longo de um horizonte de planejamento finito.

Pelo fato de existirem várias restrições ligadas a esse problema, como demanda a ser atendida, limitação de capacidade, tempos ou custos de ajuste de produtos, tempos de processamento variados; o problema pode ser tornar extremamente complexo.

Para resolver problemas combinatórios deste nível foram propostos métodos exatos e heurísticos. Os métodos exatos tendem a ser inadequados para problemas de elevada complexidade, pois geralmente não estabelecem uma solução de boa qualidade dentro de um tempo razoável. Por isso, este trabalho propõe utilizar um método heurístico. Métodos Heurísticos não garantem retornar a melhor solução, mas

costumam pelo menos obter soluções de boa qualidade em um tempo razoável.

Segundo Bueno (2009), métodos heurísticos são algoritmos exploratórios que buscam resolver problemas que, geralmente, não envolvem a implementação computacional de um conhecimento especializado. Por este motivo, muitas vezes, esses métodos são classificados como “busca cega”. Uma solução ótima de um problema nem sempre é o alvo dos métodos heurísticos uma vez que, tendo como ponto de partida uma solução viável, baseiam-se em sucessivas aproximações direcionadas a um ponto ótimo.

1.2. Metodologia Científica

A heurística Otimização por Enxame de Partículas (OEP) foi escolhida como método para resolver o PGDLPP. A escolha se baseou no fato de OEP ser um método que apresenta bons resultados em diversos trabalhos encontrados na literatura, além de ser uma metaheurística geral e fácil de implementar.

O OEP é uma técnica evolutiva desenvolvida por Kennedy e Eberhart em 1995. Assim como todas as outras abordagens de inteligência de enxame, o OEP também foi baseado em uma população de indivíduos que possuem a capacidade de interagir entre si. A OEP se diferencia de outras abordagens pelo fato de simular relações sociocognitivas, onde cada indivíduo possui a capacidade de avaliar as experiências que teve que enfrentar e as experiências que seus vizinhos passarão (Von Zuben 2008).

A pesquisa é classificada como exploratória por consistir na busca por um método mais eficiente que os já existentes na resolução do problema. Com base nos procedimentos, ela é uma pesquisa bibliográfica pelo fato da metaheurística utilizada no problema ser facilmente encontrada em artigos e livros; também é experimental, pois a qualidade dos resultados que a metaheurística usa varia de acordo com uma série de variáveis. Trata-se de uma pesquisa operacional por otimizar resultados. A pesquisa é classificada como de laboratório, pois variáveis envolvidas podem ser controladas. Para obter os resultados foram utilizados computadores Core 2 Duo, 2,66 GHz e 2 GB RAM. Neste trabalho foi utilizada a linguagem Java e foi utilizado o sistema operacional Microsoft Windows XP para a compilação e execução do método.

1.3 Objetivos do Trabalho

O trabalho tem como objetivo específico estudar e propor novas abordagens para a OEP visando sua aplicação ao problema estudado. Estudar o PGDLPP e explorar suas características visando facilitar a busca por soluções.

Outros Objetivos:

- Comparar os resultados obtidos com outros resultados encontrados na literatura. No qual um destes deve ser um método exato.
- Obter resultados satisfatórios em um tempo computacional razoável.

2. REFERENCIAL TEÓRICO

2.1. Problemas de Produção e Dimensionamento de lotes

Pode ser chamado de sistema produtivo a maneira de como a empresa organiza e realiza suas operações de produção, de forma a manter uma dependência entre todas as etapas do processo produtivo. Segundo Moreira (1996) em Dutra (2006, p.23), “um sistema de produção é uma entidade abstrata, porém a definição do conceito é importante para se ter uma idéia da totalidade”.

O objetivo de uma programação em um sistema de produção é elaborar uma maneira de atribuir e seqüenciar o manejo de recursos compartilhados para atender e satisfazer as restrições de produção e minimizar os custos de produção. As demandas de produtos em sistemas de manufatura podem ser convertidas em lotes de itens que devem ser processados no chão de fábrica por uma ou mais máquinas. Prioridade entre os itens, prazos de entrega, restrições de custos, níveis de produção, restrições relacionados aos tamanhos dos lotes, disponibilidade e capacidade de máquinas, precedências de operações, requerimentos de recursos e disponibilidade de recursos são fatores que influenciam a programação da produção.

Segundo Harding (1981) em Erdmann (2000), os sistemas de produção podem ser definidos como “um conjunto de partes inter-

relacionadas, as quais quando ligadas atuam de acordo com padrões estabelecidos sobre *inputs* (entradas) no sentido de produzir *outputs* (saídas)”. A figura 1 ilustra um modelo de sistema de produção.

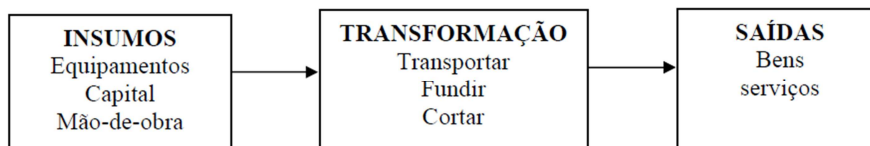


Figura 1: Modelo de sistema. Fonte: Tubino (2000).

Os insumos disponíveis (equipamentos, capital, mão-de-obra, etc) determinam restrições e custos relacionados à etapa de transformação dos insumos. Uma vez que os insumos são transformados, as saídas dos sistemas produtivos são obtidas na forma de bens e serviços. De acordo com Tubino (2000) há diferentes maneiras de qualificar um sistema produtivo que influenciarão diretamente no modo de se fazer o planejamento e controle da produção. Ele classifica o planejamento e controle de produção em três formas diferentes.

Primeiro, Tubino classifica por grau de padronização dos produtos, dividindo em produtos padronizados e produtos sob medidas. Produtos padronizados são fabricados de forma uniforme por todo sistema, normalmente produzidos em grande quantidade, estão sempre disponíveis aos clientes e representam uma facilidade da empresa em padronizar seus métodos de trabalho. Produtos sob medida, como o nome indica, são produtos produzidos especificamente para um cliente que o requisitou, apresentam certa dificuldade em padronizar os procedimentos de trabalho e normalmente são produtos mais caros.

A segunda forma considera os tipos de operações que são divididos em processos contínuos e discretos. Processos contínuos são aqueles que apresentam alto grau de uniformidade na produção, os processos são independentes entre si e o sistema não possui flexibilidade. Os processos discretos são processos em que os produtos podem ser isolados, individualmente ou em lotes, ou podem ser divididos em processo repetitivo em massa e em processo repetitivo em lote. Processo repetitivo em massa trabalha com grande quantidade de produtos padronizados, onde o sistema é pouco flexível, normalmente usado para montagem final do produto. Processo repetitivo em lote é caracterizado pela fabricação de produtos padronizados em certas quantidades (tamanho do lote), o sistema de produção é mais flexível a alterações.

A terceira e última classificação é definida pela natureza do produto que se divide em duas classes. O produto poderá ser um bem ou um serviço. A produção é orientada para o produto quando o produto é um bem e é orientada para a ação quando o produto é um serviço.

Segundo Tubino (2000, p. 31), o tipo de sistema produtivo utilizado pela empresa implica diretamente no nível de complexidade necessária para a execução do Planejamento e Controle da Produção (PCP). Chiavenato (1991) definiu que o PCP tem a função de planejar e programar a produção e as operações da empresa, controlando-as adequadamente com o objetivo de aumentar a eficiência.

Tubino (1987) acredita que o PCP é responsável por coordenar e aplicar os recursos produtivos para atender os planos estabelecidos em níveis estratégico, tático e operacional. Cada nível é descrito logo abaixo:

- Nível estratégico: o plano de produção se estabelece a longo prazo, de acordo com as previsões de venda, capacidade financeira e produtividade.
- Nível tático: o planejamento mestre de produção é estabelecido em médio prazo. Nesta etapa, o PCP tem como função analisar as necessidades de recursos produtivos para localizar gargalos na produção.
- Nível operacional: diz respeito à programação de produção, é estabelecido no curto prazo, tem como finalidade decidir quanto, quando e onde cada produto deverá ser produzido. Este nível depende de que os dois outros níveis tenham sucesso.

A figura 2 ilustra o processo de planejamento.

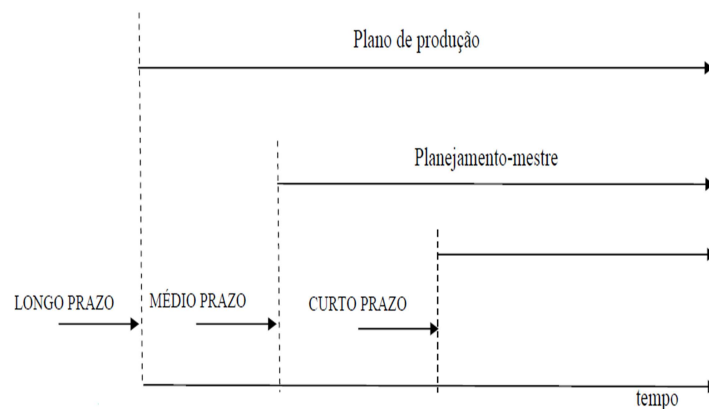


Figura 2: Processo de Planejamento. Fontes: Côrrea (2004) e Tubino (2000)

Zacarelli (1979) define o PCP como “um conjunto de funções inter-relacionadas que objetivam comandar o processo produtivo e coordená-lo com os demais setores administrativos”. Para Burbidge (1988) “o objetivo do PCP é proporcionar uma utilização adequada dos recursos, de forma que produtos específicos sejam produzidos por métodos específicos, para atender um plano de vendas aprovado”. Já para Plossl (1985) “o objetivo do PCP é fornecer informações necessárias para o dia a dia do sistema de manufatura, reduzindo os conflitos existentes entre vendas, finanças e chão de fábrica”.

Sendo assim o PCP determina quais os produtos a serem produzidos, quando será produzido, de que forma ocorrerá a produção e onde será produzido. O problema de dimensionamento de lotes tem como finalidade definir uma quantidade de itens a ser produzida em uma ou mais máquinas, em cada período ao longo de um horizonte de tempo finito. A finalidade deste planejamento é atender a demanda, podendo ou não estar sujeito a algumas restrições.

Alguns autores consideram o tempo de preparação na modelagem do problema e outros não. Alguns consideram que o tempo de preparação esta implícito nos custos de preparação (Maes e Van Wassenhove, 1991), assim não é necessário usá-lo no modelo. Outros autores acreditam que incorporar o tempo de preparação nos seus custos pode levar a uma falsa representação do consumo de recursos (Billington *et al*, 1983 e Kuik *et al*, 1994). Segundo Billington *et al* (1994) o tempo de preparação pode ser ignorado em algumas indústrias de processo, porém em sistemas que utilizam restrições de capacidade não pode ser desconsiderado. Um dos fatores mais críticos no problema de dimensionamento de lote é o tempo

de preparação e não seu custo. Segundo Baldo (2009), o problema de dimensionamento de lotes avalia três tipos de custos: o custo de produção, estoque e de preparação de máquinas (*setup cost*). Os itens têm a capacidade de dividir recursos que muitas vezes são escassos (capacidade limitada). Deve-se levar em conta também que os recursos são consumidos para produção e preparação de máquinas (*setup time*). De acordo com Karimi (2003), citado por Nascimento (2007), as seguintes características são relevantes na classificação de problemas de dimensionamento de lotes:

- 1. Horizonte de Planejamento:** Tempo total de produção. Normalmente é um tempo finito, mas pode ser infinito ou dividido em períodos.
- 2. Número de Níveis de Estágio:** Pode ser classificado em monoestágio ou multiestágio. Monoestágio seria quando a produção de um item não depende do outro. Multiestágio seria quando existe uma dependência entre os itens a serem produzidos, isto é, para produzir um item depende-se de outro item. O presente trabalho estuda um problema monoestágio.
- 3. Número de produtos a serem produzidos:** pode haver um ou mais produtos no planejamento de produção.
- 4. Capacidade de recursos:** Quantidade máxima que se pode produzir.
- 5. Demanda:** Há dois tipos de demanda: estática ou dinâmica. A demanda estática é aquela que não muda no

horizonte de planejamento. A demanda dinâmica é aquela que muda no horizonte de planejamento.

6. **Custos de configuração (*setup*):** São os tempos de preparação dos produtos ou itens no processo produtivo.
7. **Número de máquinas:** Pode-se usar mais de uma máquina para a produção e atendimento das demandas. Porém um maior número de máquinas torna o problema mais complexo.
8. **Número de micro-períodos:** Cada período em um horizonte de planejamento está restrito a um número máximo de configurações da máquina, onde cada configuração representa um micro-período.

2.2. Heurísticas e Metaheurísticas

Uma heurística pode ser definida com um algoritmo construído para encontrar uma solução viável, para um dado problema com determinada função objetivo em um tempo computacional aceitável (Diaz, 1996). Metaheurísticas são estruturas algorítmicas gerais de fácil adaptação a diversos problemas de otimização. Um algoritmo que não possui conhecimentos completos sobre seu comportamento é considerado uma metaheurística, ou seja, é um algoritmo com o objetivo de resolver um problema complexo utilizando uma quantidade de recursos com a finalidade de encontrar uma solução viável (Sucupira, 2004).

Entre as metaheurísticas conhecidas na literatura se destacam os Algoritmos Genéticos (AGs), a Busca Tabu (BT) e *Simulated Annealing* (SA). O primeiro Algoritmo Genético surgiu em 1975 proposto por Jonh Holland (Holland, 1975). Trata-se de um método baseado na evolução das espécies que busca a melhor solução a partir de uma população inicial. Cruzamentos dos indivíduos dessa população são efetuados com o objetivo de gerar filhos melhores para substituir os seus pais na população. Isso ocorre a cada iteração. Assim, a cada iteração é gerada uma população diferente e melhor adaptada que pode levar à convergência dos seus indivíduos para uma melhor solução (Fernandes, 2005).

A BT foi proposta por Fred Glover e Hansen (Glover, 1986 e Hansen, 1986). Tornou-se de forma muito rápida uma das heurísticas mais robustas para solucionar problemas combinatórios. A BT consiste em estabelecer regras que regem uma heurística de exploração do espaço de soluções de forma a limitar o espaço de busca durante a busca. Isso se deve pelo uso de uma memória de soluções ou movimentos e pela aceitação de soluções piores com o objetivo de se evitar retorno a soluções previamente visitadas (Fraga, 2006).

O SA foi baseado no processo de fundição de metal, em que o metal é aquecido a uma temperatura elevada e esfriado vagarosamente em seguida. O objetivo é retornar uma massa homogênea. Este processo é chamado de *annealing* (Haeser et al, 2008). O SA também pode ser visto como uma generalização do método Monte Carlo para analisar as equações de estado.

2.3. Otimização por Enxame de Partículas (OEP)

Otimização por Enxame de Partícula (OEP) é uma metaheurística baseada em populações. Trata-se de um método que utiliza a idéia de uma inteligência coletiva. Trata-se de uma técnica estocástica de otimização desenvolvida por Eberhart e Kennedy em 1995 (Eberhart e Kennedy, 1995), onde há uma população de indivíduos capazes de interagir entre si e com o meio ambiente. A OEP é inspirada em experimentos sociais, onde todos os indivíduos de uma população são capazes de absorverem conhecimento de seus vizinhos (Von Zuben, 2007). De acordo com Kennedy et al. (2001), há três princípios que resumem esse processo de adaptação cultural. O primeiro é o princípio da avaliação, onde cada indivíduo tem a capacidade de se avaliar. O segundo princípio é o da comparação, pois cada indivíduo tem a capacidade de utilizar os demais como material comparativo. O terceiro princípio é o da imitação que é a capacidade de obter melhorias aprendendo com outros indivíduos.

Cada partícula mantém guardada informações sobre seu percurso no espaço de busca, estas informações estão associados à melhor solução encontrada até o momento pela própria partícula (p_{Melhor}), à melhor posição local obtida até agora por alguma partícula próxima (l_{Melhor}) e à melhor posição global em relação a todo o enxame obtida por uma partícula do enxame (g_{Melhor}). Segue um pseudo código na figura 3.

```

1 Pseudo_Codigo_OEP(
2     Iniciar aleatoriamente N particulas;
3     Iniciar aleatoriamente os vetores velocidade
4         de cada particula;
5     Avaliar a funcao objetivo para cada particula;
6
7     Enquanto não atender a condição de parada(
8         Atualiza pMelhor;
9         Atualiza lMelhor;
10        Para cada particula(de 1 ate N){
11            Atualiza o vetor velocidade;
12            Efetua o movimento da particula;
13        }
14    }
15 }

```

Figura 3: Pseudocódigo OEP.

A “memória” dos movimentos exploratórios realizados pelas partículas é representado por pMelhor, a “cooperação” entre as partículas é disseminada por lMelhor e a inércia relacionada a partícula é armazenada por sua posição atual. Essas posições podem ser vistas como vetores, onde a soma vetorial resultante determina a nova posição de uma partícula X_i . A figura 4 exemplifica essa idéia.

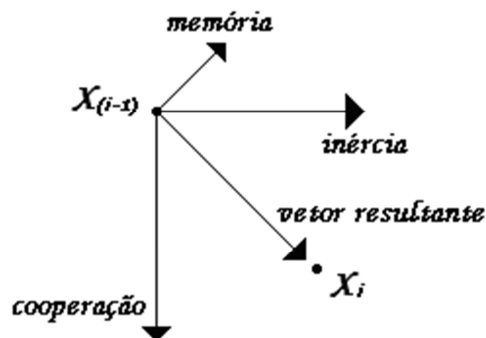


Figura 4: Movimento de uma partícula.

Nos últimos anos, OEP tem sido aplicado com sucesso em muitas áreas.

Ali & Kaelo (2008) utilizam OEP na otimização de funções multimodais. Chen & Zhao (2009) propõe uma OEP adaptativa, onde o tamanho da população varia segundo uma função do tipo escada. O método é aplicado na otimização de funções e no treinamento de redes neurais.

Um teste foi realizado por Siciliano (2007) para comparar o AG com o OEP usando problemas de otimização de funções não lineares. Uma função simples sem mínimos locais e outra com diversos mínimos e máximos locais foram utilizadas. Os resultados obtidos demonstraram que a OEP apresentou melhor desempenho em todos os testes. Segundo Prado (2005), que também realizou vários testes comparando OEP e AG em problemas de otimização de funções, observou-se que as duas técnicas convergiram para o mesmo resultado. No entanto, a OEP trabalhou com um tamanho de população menor que o AG, apresentando assim um esforço computacional menor.

Aloise *et al.*, (2006), baseado no fato de que a OEP clássica obtinha bons resultados para problemas contínuos de otimização combinatória, adaptou a OEP para um problema discreto. Os autores apresentaram algumas estratégias de otimização discreta aplicadas ao problema do caixeiro-viajante. A OEP alcançou resultados bastante eficientes.

3. MATERIAIS E MÉTODOS

3.1. Problema Geral de Dimensionamento de lotes e Programação da Produção

Neste trabalho atacamos o Problema Geral de Dimensionamento de Lotes e programação da Produção (PGDLPP). Este é um problema de produção bem conhecido na literatura, em que certa quantidade de produtos deve ser produzida por uma ou mais máquinas num horizonte de planejamento finito. Dentro de um horizonte de planejamento existem vários períodos, onde demandas por produtos devem ser atendidas ao final de cada período. Uma linha em um período pode produzir mais produtos que o necessário para atender a demanda. Todavia, a quantidade excedente será estocada ocorrendo num custo de estoque proporcional à quantidade e ao tipo de produto estocado. Cada período é segmentado em vários micro-períodos e cada micro-período é usado para produzir certa quantidade de produto. A quantidade de micro-períodos varia de período a período, mas todos os períodos possuem uma quantidade limitada de micro-períodos. O objetivo do problema é minimizar todos os custos de estoque e troca, a fim de atender as demandas respeitando a capacidade de produção das máquinas. Se a demanda não for atendida, há uma penalização associada aos produtos não produzidos.

O modelo matemático proposto nesta monografia para o PGDLPP considera máquinas paralelas e penalização por demandas não atendidas. Foi usada uma formulação apresentada por Meyr (2002) para o PGDLPP

com máquinas paralelas. Todavia, a formulação apresentada difere ao considerar variáveis que acumulam a quantidade de demanda não atendida para cada produto. Essas variáveis são incluídas na equação de balanço de estoque do primeiro período de produção e também são penalizadas na função objetivo do modelo.

Como foi dito anteriormente o horizonte de planejamento será dividido em T macro-períodos, onde cada macro-período $t \in T$ será dividido em S micro-períodos. O tamanho de cada micro-período s varia proporcionalmente ao produto alocado. De acordo com o modelo a ser apresentado, somente um produto pode ser produzido em cada micro-período. Logo, variáveis de dimensionamento de lotes e atribuição de produtos às linhas e períodos são indexadas por produtos e micro-períodos.

A figura 5 apresenta um exemplo no qual há um conjunto de macro-períodos $T = \{t1, t2\}$ e um conjunto de micro-períodos $S = \{s1, s2, s3, s4, s5\}$ em uma linha $L1$. Também são considerados três produtos $J = \{j1, j2, j3\}$. O tamanho dos micro-períodos varia já que é proporcional ao tempo de processamento daquele produto alocado ao micro período. Por exemplo, o micro período $s2$ em $t1$ é maior que $s1$ em $t1$ indicando que o tempo de processamento do produto $j2$ em $t1$ é maior do que o tempo de processamento do produto $j1$ em $s1$.

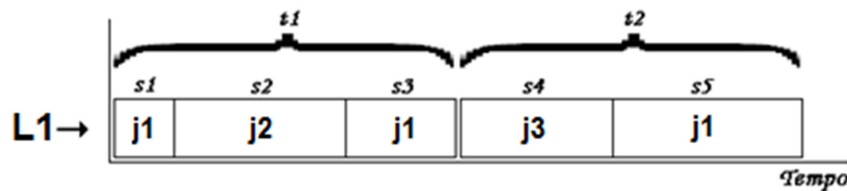


Figura 5: Estrutura do Horizonte de planejamento.

Os parâmetros utilizados na formulação matemática do PGDLPP são listados abaixo, onde são considerados J produtos e L linhas.

- C_t : Capacidade (tempo) disponível no macro-período t .
- TPl_j : Tempo de produção do produto j na linha l .
- MIN_j : A quantidade mínima que deverá ser produzida do produto j .
- H_j : Custo para estocar o produto j .
- CT_{ij} : Custo de troca do produto i para o produto j .
- D_{it} : Demanda exigida do produto i no macro-período t .
- I_{j0} : Estoque inicial do produto j no início do horizonte de planejamento.
- Y_{lj0} : Este parâmetro contém 1 se o produto j estiver ajustado inicialmente na linha l , contém 0 se não.
- M : Contém a penalização por unidade de demanda não atendida.

A seguir são definidas as variáveis do modelo.

- I_{jt} : Estoque do produto j no final do macro-período t . Este valor deve ser maior ou igual a zero.

- Q_{ljs} : Quantidade do produto j produzido no micro-período s na linha l . Este valor deve ser maior ou igual a zero.
- Q_j : Quantidade de demanda do produto j que não foi produzida. Este valor deve ser maior ou igual a zero.
- Y_{ljs} : Esta variável recebe 1 se o produto j é atribuído à linha l no micro-período s , recebe 0 caso não.
- Z_{lijs} : Se houver troca do produto i pelo produto j no micro-período s na linha l esta variável recebe 1, recebe 0 caso não. Este valor deve ser maior ou igual a zero.

O modelo matemático do PGLDPP está representado na figura 6.

$$\begin{aligned} \text{Minimize } & \sum_{j=1}^J \sum_{t=1}^T H_j I_{jt} + \sum_{i=1}^J \sum_{j=1}^J \sum_{l=1}^L \sum_{s=1}^{T.S} C T_{ij} Z_{lijs} + M \sum_{j=1}^J q_j^0 & (1) \\ I_{j1} = & I_{j,0} + q_j^0 + \sum_{l=1}^L \sum_{s=1}^S q_{ljs} - D_{j1} & j = 1, \dots, J & (2a) \\ I_{jt} = & I_{j,t-1} + \sum_{l=1}^L \sum_{s=(t-1)S+1}^{tS} q_{ljs} + q_j^0 - D_{jt} & j = 1, \dots, J, t=1 \dots T & (2b) \\ \sum_{j=1}^J \sum_{s=(t-1)S+1}^{tS} T P_{lj} q_{ljs} \leq & C_t & l = 1, \dots, L, t=1 \dots T & (3) \\ q_{ljs} \leq & \frac{C_t}{T P_{lj}} y_{ljs} & l = 1, \dots, L, j=1 \dots J, s=1 \dots T.S & (4) \\ q_{ljs} \geq & \text{Min}_j (y_{ljs} - y_{ljs-1}) & l = 1, \dots, L, j=1 \dots J, s=1, \dots, T.S & (5) \\ \sum_{j=1}^J y_{ljs} = & 1 & l = 1, \dots, L, s=1, \dots, T.S & (6) \\ z_{lijs} \geq & y_{li,s-1} + y_{ljs} - 1 & l = 1, \dots, L, i=l, \dots, J, j=l, \dots, J, s=1, \dots, T.S & (7) \end{aligned}$$

Figura 6: Modelo matemático do PGLDPP.

A função objetivo (1) minimiza os custos de estoque e de troca. Os custos de troca são independentes das linhas, mas dependentes da seqüência dos produtos. As restrições (2a) e (2b) determinam o estoque, onde a demanda não atendida é acumulada na equação de estoque (2a). A variável q_j^0 acumula as demandas não atendidas que é penalizada na função objetivo (1). Essa representação foi utilizada por Toledo *et al* (2009) em um modelo integrado de dimensionamento e programação da produção em fábricas de bebidas.

A restrição (3) garante que dentro do período t a capacidade disponível não é violada. A capacidade disponível de um macro-período é a mesma para todas as linhas, porém há diferentes tempos de processamento $TP_{l,j}$ para os produtos. A restrição (4) define que não haverá produção de um produto que não esteja atribuído a um micro-período. Também impõe o tamanho máximo de um produto em um micro-período.

A restrição (5) garante que um lote mínimo será produzido quando ocorre a atribuição de um novo produto a um micro-período. A equação (6) restringe que somente um produto seja produzido em um micro-período. As variáveis de trocas de produtos são estabelecidas pela restrição (7).

3.2. Proposta de Otimização por Enxame de Partículas (OEP)

3.2.1. Representação e Avaliação das soluções

Toledo *et al* (2009) definiu uma representação para indivíduos em um Algoritmo Genético (AG) que é uma extensão para máquinas paralelas da representação proposta originalmente em Fleischmann e Meyr (1997) para o PGDLPP com uma única máquina. Esta representação consiste em uma lista de máquinas ou linhas, onde cada linha armazena uma matriz com informações sobre sua seqüência de produção. Essa matriz é do tipo $T \times S$, onde $t \in T$ se refere ao número de macro-períodos e $s \in S$ ao número de micro-períodos. Esta foi a representação adotada neste trabalho e está ilustrada na figura 7.

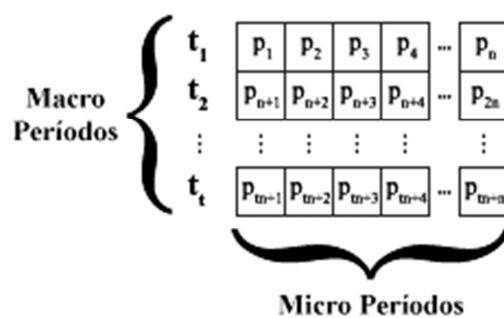


Figura 7: Representação da solução.

Um produto ocupa um determinado micro-período dentro de um macro-período (t,s) . Uma regra a ser respeitada é que dois micro-períodos consecutivos não podem conter o mesmo produto. A única exceção a esta regra ocorre caso o último micro-período de um macro-período t tenha produto igual ao produto do primeiro micro-período do macro-período $t+1$. A figura 8 mostra uma representação da solução considerando um problema com duas linhas, dois macro-períodos com quatro micro-períodos e quatro produtos.

		<i>Sequência</i>				
$L.1$	{	$T1 \rightarrow$	$P3$	$P1$	$P4$	
	}	$T2 \rightarrow$	$P2$	$P4$		
$L.2$	{	$T1 \rightarrow$	$P2$			
	}	$T2 \rightarrow$	$P4$	$P3$	$P1$	

Figura 8: Exemplo de codificação.

A avaliação de uma solução consiste no cálculo do valor da função objetivo (1) apresentada na seção 3.1. Esse processo exige a determinação dos custos de troca, estoque e a penalização por demanda não atendida. Os custos de troca são calculados a partir da sequência de produtos existentes em cada linha na representação da solução. Os custos de estoque e a penalidade por demanda não atendida exigem a determinação do dimensionamento dos lotes a partir das informações codificadas na representação da solução. Esse dimensionamento é feito através do método apresentado no pseudocódigo a seguir (Figura 9). Esse método é baseado no algoritmo *greedy-mod* proposto em Fleischmann &

Meyr (1997) para o PGDLPP com máquinas simples. O método de decodificação é determinístico e define o tamanho dos lotes, minimizando o custo de estoque. Os parâmetros do algoritmo são descritos abaixo:

- X_{ljt} - armazena o lote do produto j produzido na linha l no macro-período t .
- Y_{ljt} - número de lotes do produto j no macro-período t na linha l .
- D_j - demanda acumulada (total) do produto j .
- AC_{jt} - demanda acumulada do produto j até o macro-período anterior ao t .
- LS_{jt} - limite superior estimado considerando que toda a capacidade disponível fosse utilizada para produzir o produto j em períodos anteriores ao t .

$$LS_{jt} = 0; \text{ para } t=1 \text{ e } LS_{jt} = LS_{j(t-1)} + \sum_{l=1}^L \frac{\text{Mínimo}(1, Y_{ljt}) * K_{lt}}{TP_{lj}} ; \text{ para } t > 1$$

- X_{ljs} - lote do produto j produzido na linha l no micro-período s .
- Q_j - quantia de demanda do produto j não alocada pelo método de decodificação.

O algoritmo trabalha preenchendo as quantias dos lotes em ordem decrescente de macro-período. Primeiro aloca o lote mínimo para os produtos atribuídos a micro-períodos na matriz de sequência das linhas. Em seguida, dimensiona os lotes para produtos que, se não forem produzidos no macro-período corrente, não haverá capacidade disponível para sua produção ou o produto não aparecerá em outra posição na matriz

de sequência das linhas. O objetivo é evitar infactibilidades. Por último, todos os produtos alocados na matriz de sequência das linhas têm seus lotes redimensionados, seguindo a ordem decrescente do quociente H_j/TPl_j .

```

Método de dimensionamento de lotes
t=T
Enquanto (t≥1) faça
  //----Garantindo lote mínimo-----
  Para toda linha l
    Para todo produto com (Yljt>0)
      p = Mínimo( Minj * Yljt ; Dj - ACjt );           //Calcula a parcela a ser
      produzida
      Atualiza Xljt, Klt, Dj;
      Se Klt < 0 então Penalizar capacidade violada

  //----Evitando infactibilidades-----
  Para todo produto j
    Q = Dj - LSjt;
    Para toda linha l com (Yljt>0 e Q > 0)
      p = Mínimo( Dj - ACjt ; Klt/TPlj ; Q );
      Atualiza Xljt, Klt, Dj;

  //-----Distribui em uma ordem gulosa-----
  Para toda linha l e produto j com (Yljt > 0 ) em ordem decrescente de Hj/TPlj
    p = Mínimo( Dj - ACjt ; Klt/TPlj );
    Atualiza Xljt, Klt, Dj;
t = t-1
fim Enquanto

Qj = Dj, para todo j.
//----Distribui as demandas dos macro-período para os micro-períodos-----
Para todo l,j,t e s pertencente a St
  Se Yljt > 0 então
    Xljs = Xljt / Yljt;
  Senão
    Xljs = 0;
fim do método.

```

Figura 9: Pseudo-código Método de Decodificação.

A Figura 10 abaixo demonstra como são atualizadas as quantias X_{ljt} , K_{lt} e D_j pela parcela p de demanda que será produzida.

Atualiza X_{ljt} , K_{lt} , D_j dada uma parcela de demanda p calculada	
$X_{ljt} = X_{ljt} + p;$	//Adiciona a parcela, a quantia a ser produzida
$K_{lt} = K_{lt} - p * TPlj;$	//Subtrai a capacidade para produzir a quantia p .
$D_j = D_j - p;$	//Nova demanda do produto que deverá ser atendida.

Figura 10: Atualizar X_{ljt} , K_{lt} , D_j .

3.2.2. Algoritmo de Otimização por Enxame de Partículas (OEP)

As figuras 11 e 12 apresentam os pseudocódigos para a OEP proposta. Uma partícula pode assumir diversas representações da solução durante a execução do método. Neste contexto, cada representação da solução de uma partícula é a posição assumida por esta no espaço de busca. No pseudocódigo, P_i é o conjunto com a posição (representação da solução) atual das partículas i . M_i é o conjunto que armazena a melhor posição onde as partículas P_i já estiveram. O pseudo-código força que essa melhor posição seja reinicializada após um certo número de iterações. O objetivo é não ficar preso na mesma melhor posição durante todo o código. P_g é a melhor posição alcançada pelo enxame de partículas. P_{rand} é uma partícula aleatória, que pode ser totalmente aleatória ou gerada pela recombinação de partículas existentes através de mudanças aleatórias. $Pesos(j)$ define a probabilidade de escolher a partícula j para recombinação. A atualização dos pesos é dinâmica, ou seja, fica variando entre um valor mínimo e máximo. Essa variação é ditada pelo parâmetro

d(j) que determina a quantia adicionada ou subtraída de Peso(j):

- o $\text{PesoMáximo}(j) \geq \text{Pesos}(j) \geq \text{PesoMínimo}(j)$

```
OEP  
Inicializa as partículas Pi, Mi.  
Pg = Melhor(Pi);  
  
Enquanto critério não for atingido  
Escolhe um tipo Prand usada na recombinação.  
Restarta Mi  
Para cont=1 até 20  
    Pi = Oscila(Pi,Mi,Pg, Prand);           // para toda partícula  
    Atualiza Pg e Mi;  
    Atualiza os pesos(j) dos vetores de recombinação.  
Fim OEP
```

Figura 11: Pseudocódigo para OEP.

```
Oscila(Pi)  
    m = Recombinação(Pi, Mi, Pg, Prand);   //para toda partícula  
    Para cont = 1 até 40  
        aux = Recombinação(Pi, Mi, Pg, Prand); //para toda partícula  
        Se avaliação(aux) < avaliação (m)           //para toda partícula  
            m = aux;                               //armazena partícula com melhor valor em  
    Pi  
    Retorne m;  
Fim Oscila
```

Figura 12: Pseudocódigo para o método Oscila.

Cada partícula oscila em torno de um ponto antes de efetuar um movimento. O objetivo é garantir que seu próximo movimento tenha um ganho significativo ou garantir pelo menos que o movimento não piore muito a posição atual da partícula. Para efetuar o movimento de uma partícula, o método proposto utiliza uma recombinação envolvendo quatro vetores que armazenam posições (representações da solução) de partículas. O primeiro vetor armazena a posição atual das partículas (Pi), o segundo vetor armazena a melhor posição alcançada pelas partículas (Mi), o terceiro vetor armazena a melhor posição já obtida por todo o

enxame (Pg) e o quarto vetor armazena uma posição aleatória (Prand). A recombinação gera novas posições para as partículas baseada em seqüências de produtos herdadas das representações armazenadas nos quatro vetores (Figura 13). A seleção de qual vetor irá contribuir segue uma probabilidade de escolha definida como:

$$Probabilidade(j) = \frac{Pesos(j)}{\sum_{j=1}^4 Pesos(j)}$$

Recombinação(Pi, Mi, Pg, Prand)
Para cada partícula
Para cada linha-l
Para cada macro-periodo-t
 Seleciona um vetor j de acordo com *Probabilidade(j)*.
 Preenche uma nova partícula com a sequência em l e t do vetor selecionado..
Atualiza partícula em Pi com nova partícula;
Retorna Pi com partículas atualizadas.
Fim Recombinação

Figura 13: Pseudocódigo para recombinação.

A figura 14 ilustra uma possível faixa de participação para cada vetor de posição.

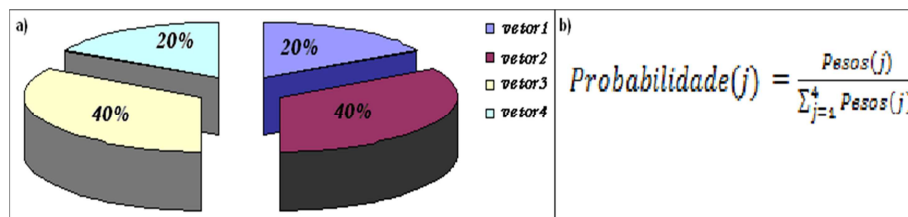


Figura 14: Possível contribuição de cada vetor.

Duas abordagens são propostas para execução da OEP. A primeira chamada OEP1 permite que os valores dos pesos $Pesos(j)$ variem no

decorrer da execução do método. Para isso, valores máximos e mínimos são ajustados para cada peso. Também são ajustados o valor inicial dos pesos e a taxa de decréscimo ou acréscimo relacionada. Durante a execução, os pesos irão aumentar ou diminuir sem ultrapassar os valores máximos e mínimos estabelecidos. Na segunda abordagem chamada OEP2, não há variação dos pesos e o método é executado com o valor inicialmente atribuído a cada $Peso(j)$.

Para uma partícula efetuar um movimento, o método proposto utiliza uma recombinação de quatro vetores, cada vetor é representado por uma posição já avaliada, ou seja, os quatro vetores são compostos por uma partícula do conjunto P_i , por uma partícula do conjunto M_i , pela partícula P_g e pela partícula P_{rand} . A recombinação gera uma nova posição para a partícula, onde cada vetor contribui com uma parte na nova posição gerada. Antes de uma partícula efetuar um movimento ela oscila em volta de sua posição, o objetivo é garantir que seu movimento tenha um ganho significativo, no caso, mesmo que o melhor movimento não melhore a partícula ele é aceito. Durante a recombinação, cada vetor contribui com uma parte de sua informação respeitando o peso de sua informação.

A figura 15 é um exemplo de como acontece o movimento de uma partícula. A idéia é atribuir pedaços de informações de cada vetor para formar a nova posição. Como pode ser visto o vetor 3 contribui mais que os outros vetores por possuir um peso maior.

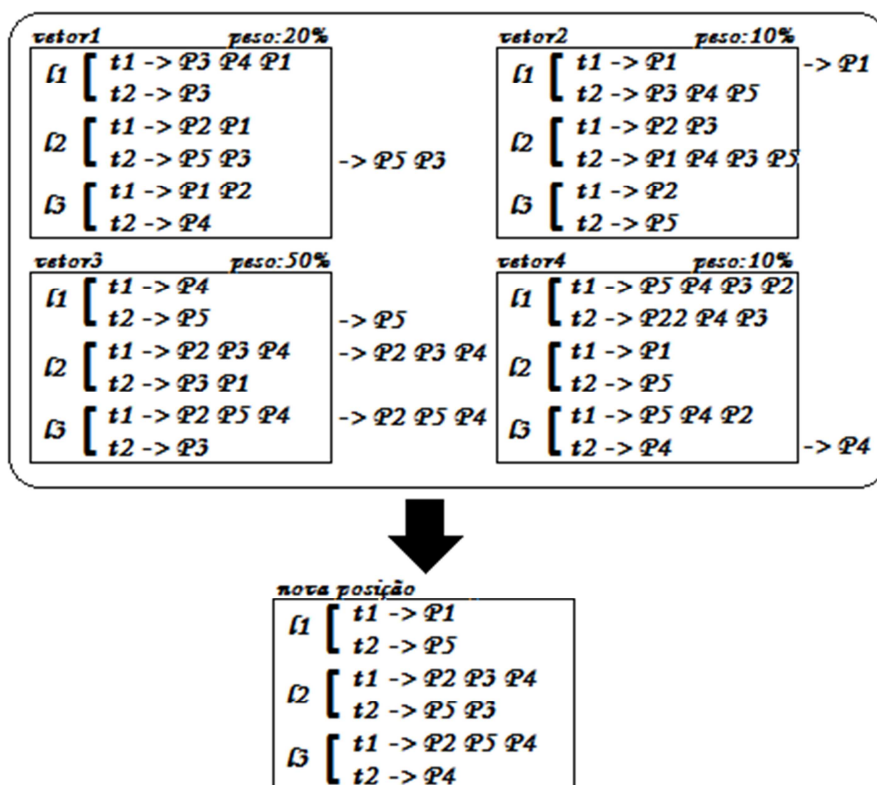


Figura 15: Exemplo de movimento de uma partícula.

4. RESULTADOS COMPUTACIONAIS

4.1. Parâmetros do Problema

Os parâmetros de quatro conjuntos de instâncias para o PGDLPP com uma única máquina (máquina simples) estão sendo apresentados na tabela 1 são os mesmos parâmetros usados por Haase (1996). Foi gerado um total de 10 instâncias para cada conjunto. A tabela apresenta a quantidade de macro-períodos (T), micro-períodos (S), produtos (J), capacidade da máquina (Ct) e a porcentagem de utilização da capacidade disponível (U%).

Tabela 1: Conjunto de instâncias para o PGDLPP com uma única máquina.

Conjuntos	Macro-Períodos (T)	Micro-Períodos (S)	Produtos (J)	Capacidade (Ct)	U%
S1	6	4	4	200	80
S2	6	4	4	200	90
S3	5	4	4	200	90
S4	5	5	5	250	90

Os demais parâmetros do problema também foram ajustados segundo Haase (1996), onde $d_{it} \in \{0,1,\dots,100\}$, $H_j=1, TP_{1j}=1$, $CT_{ij} \in \{100,\dots,200\}$ para $i \neq j$ e $CT_{ij}=0$ para $i=j$, $M=10000$ e $Min_j = 1$.

Três conjuntos de instâncias foram gerados para avaliar o PGDLPP com máquinas paralelas. Os parâmetros utilizados estão apresentados na tabela 2. Os demais parâmetros permanecem iguais aos das máquinas simples exceto o $TP_{ij} \in [0,1]$. Um total de cinco instâncias foram gerados para cada um desses conjuntos. A tabela apresenta a quantidade de macro-períodos (T), micro-períodos (S), produtos (J), a capacidade da máquina (Ct), a quantidade de linhas utilizadas (L) e a porcentagem de utilização da capacidade disponível (U%).

Tabela 2: Conjunto de instâncias para o PGDLPP com mais de uma máquina.

Conjuntos	Macro-Períodos (T)	Micro-Períodos (S)	Produtos (J)	Capacidade (Ct)	Linha (L)	U%
P1	6	4	4	200	2	80
P2	6	4	4	200	3	80
P3	6	4	4	200	4	80

4.2. Resultados em máquinas simples

O modelo matemático da seção 3.1 e as instâncias geradas foram codificados utilizando o pacote computacional AMPL e solucionados pelo solver CPLEX 11.0 com o tempo de execução de 1 hora. Foram utilizadas as duas abordagens propostas para OEP, uma com os pesos de cada vetor variando durante a execução do método, como foi definido na

seção 3.2.2, denominado OEP1 e a outra abordagem, sem esta variação dos pesos dos vetores, denominado OEP2.

Para analisar melhor o desempenho do método proposto foram utilizados os resultados da metaheurística Algoritmo Genético (AG) obtidos para o mesmo problema por Toledo *et al* (2009). O AG foi executado utilizando uma população com 13 indivíduos, 2,5 de taxa de recombinação (*crossover*) e 0,7 de taxa de mutação.

OEP1 foi executada com 30 partículas, onde o vetor com a posição atual foi ajustado com peso inicial 0, taxa de crescimento 0,8 e limitante superior para o valor do peso em 10. O vetor com a melhor posição da partícula recebeu peso inicial 90, taxa de decaimento 1,6 e limitante inferior 0. O vetor com a melhor posição de todo o enxame recebeu peso inicial 0, taxa de crescimento 0,8 e limitante superior 80. O vetor aleatório recebeu valor inicial 10 e era ajustado durante a execução de forma que a soma dos pesos resultasse em 100. OEP2 foi executada todo o tempo com o mesmo valor inicial atribuído aos pesos da OEP1.

Todos os testes foram realizados em um processador Intel Corel 2 Duo, 2,66 GHz e 1,95 GB RAM. Cada instância foi executada 10 vezes pelas metaheurísticas para poder tirar o desempenho médio. Cada execução foi limitada a 0,5 horas, ou seja metade do tempo do método exato (AMPL/CPLEX). Usamos metade do tempo nas heurísticas em relação ao método exato, pois se a heurística, que é um método que não garante o melhor resultado, consegue o mesmo resultado no mesmo tempo que o método exato, que é um método que garante sempre encontrar o mesmo resultado no espaço de tempo que lhe foi oferecido, indica que a melhor opção seria o método exato.

Os resultados obtidos pelo AMPL/CPLEX e a média dos resultados das metaheurísticas em 10 execuções por instância de uma única máquina estão apresentados nas tabelas 3, 4, 5 e 6. Também é apresentado o desvio percentual entre o método exato e as metaheurísticas:

$$\text{DesvAMPL}(\%)=100*(Z-Z^*)/Z^*,$$

onde Z se refere ao valor médio da solução final obtida pela metaheurística e Z^* é o valor da solução final encontrado pelo AMPL/CPLEX. Na coluna AMPL, valores com * indicam que o AMPL/CPLEX alcançou o valor ótimo. Como pode ser visto nas tabelas o AMPL/CPLEX alcançou o valor ótimo em todas as execuções em S1, S2 e S3, mas no conjunto de instancias S4 é retornada apenas a melhor solução factível encontrada pelo AMPL/CPLEX. Para as duas abordagens foi colocado também o desvio padrão (DP) em relação aos 10 resultados obtidos de cada instancia.

Tabela 3: Resultados obtidos para S1.

Métodos	AMPL	AG	OEP1	OEP2
S1-0 DesvAMPL(%) DP	1658*	1658 0.0	1658 0.0 0.0	1658 0.0 0.0
S1-1 DesvAMPL (%) DP	1421*	1421 0.0	1421 0.0 0.0	1421 0.0 0.0
S1-2 DesvAMPL (%) DP	1700*	1700 0.0	1703 0.2 0.0	1703 0.2 0.0
S1-3 DesvAMPL (%) DP	1459*	1459 0.0	1471 0.8 0.0	1471 0.8 0.0
S1-4 DesvAMPL (%) DP	1543*	1551,4 0.5	1543 0.0 0.0	1543 0.0 0.0
S1-5 DesvAMPL (%) DP	1402*	1402 0.0	1402 0.0 0.0	1402 0.0 0.0
S1-6 DesvAMPL (%) DP	1477*	1486,5 0.6	1480 0.2 0.0	1480 0.2 0.0
S1-7 DesvAMPL (%) DP	1463*	1463 0.0	1463 0.0 0.0	1463 0.0 0.0
S1-8 DesvAMPL (%) DP	1480*	1487 0.5	1487 0.5 0.0	1487 0.5 0.0
S1-9 DesvAMPL (%) DP	1626*	1633,4 0.5	1628 0.1 0.0	1628 0.1 0.0

Tabela 4: Resultados obtidos para S2.

Métodos	AMPL	AG	OEP1	OEP2
S2-0	1729*	1746	1729	1729
DesvAMPL (%)		1.0	0.0	0.0
DP			0.0	0.0
S2-1	1758*	1758	1758	1758
DesvAMPL (%)		0.0	0.0	0.0
DP			0.0	0.0
S2-2	2162*	2162	2162	2162
DesvAMPL (%)		0.0	0.0	0.0
DP			0.0	0.0
S2-3	1747*	1748,2	1747	1747
DesvAMPL (%)		0.1	0.0	0.0
DP			0.0	0.0
S2-4	1638*	1638	1638	1638
DesvAMPL (%)		0.0	0.0	0.0
DP			0.0	0.0
S2-5	1534*	1534	1539	1539
DesvAMPL (%)		0.0	0.3	0.3
DP			0.0	0.0
S2-6	1533*	1533	1533	1533
DesvAMPL (%)		0.0	0.0	0.0
DP			0.0	0.0
S2-7	1822*	1833,8	1822	1822
DesvAMPL (%)		0.6	0.0	0.0
DP			0.0	0.0
S2-8	1863*	1864,7	1863	1863
DesvAMPL (%)		0.1	0.0	0.0
DP			0.0	0.0
S2-9	1751*	1751	1751	1751
DesvAMPL (%)		0.0	0.0	0.0
DP			0.0	0.0

Tabela 5: Resultados obtidos para S3.

Métodos	AMPL	AG	OEP1	OEP2
S3-0 DesvAMPL (%) DP	1269*	1269 0,0	1279 0.8 0.0	1279 0.8 0.0
S3-1 DesvAMPL (%) DP	1491*	1491 0,0	1491 0.0 0.0	1491 0.0 0.0
S3-2 DesvAMPL (%) DP	1338*	1338 0,0	1338 0.0 0.0	1338 0.0 0.0
S3-3 DesvAMPL (%) DP	1530*	1530 0,0	1530 0.0 0.0	1530 0.0 0.0
S3-4 DesvAMPL (%) DP	1663*	1667 0,3	1663 0.0 0.0	1663.4 0.02 1.26
S3-5 DesvAMPL (%) DP	1542*	1542 0,0	1542 0.0 0.0	1542 0.0 0.0
S3-6 DesvAMPL (%) DP	1553*	1553 0,0	1553 0.0 0.0	1553 0.0 0.0
S3-7 DesvAMPL (%) DP	1518*	1532,1 0,9	1518 0.0 0.0	1518 0.0 0.0
S3-8 DesvAMPL (%) DP	1526*	1526 0,0	1526 0.0 0.0	1526 0.0 0.0
S3-9 DesvAMPL (%) DP	1441*	1441 0,0	1441 0.0 0.0	1441 0.0 0.0

Tabela 6: Resultados obtidos para S4.

Métodos	AMPL	AG	OEP1	OEP2
S4-0	1833	1861,5	1833	1833
DesvAMPL (%)		1,6	0.0	0.0
DP			0.0	0.0
S4-1	1986	1989,6	1986	1986
DesvAMPL (%)		0,2	0.0	0.0
DP			0.0	0.0
S4-2	1825	1827,2	1805	1806,9
DesvAMPL (%)		0,1	-1.1	-1.0
DP			0.0	4.0
S4-3	1674	1681,8	1779	1779
DesvAMPL (%)		0,5	6.3	6.3
DP			0.0	0.0
S4-4	1974	1974	1974	1974
DesvAMPL (%)		0,0	0.0	0.0
DP			0.0	0.0
S4-5	1994	2011,3	1994	1995,5
DesvAMPL (%)		0,9	0.0	0.1
DP			0.0	4.7
S4-6	1790	1793,3	1790	1790
DesvAMPL (%)		0,2	0.0	0.0
DP			0.0	0.0
S4-7	1786	1784,8	1777	1777
DesvAMPL (%)		-0,1	-0.5	-0.5
DP			0.0	0.0
S4-8	2150	2195,4	2139	2139
DesvAMPL (%)		2,1	-0.5	-0.5
DP			0.0	0.0
S4-9	1782	1788,7	1782	1782
DesvAMPL (%)		0,4	0.0	0.0
DP			0.0	0.0

As duas abordagens OEP1 e OEP2 foram capazes de encontrar a solução ótima na maioria das instâncias em S1, S2 e S3. Os valores negativos de alguns desvios em S4 indicam que as duas abordagens

propostas retornaram, em média, soluções melhores que as encontradas pelo método exato. O AG também apresentou uma média de resultados melhores que o método exato.

Todavia, OEP1 e OEP2 superam os resultados médios do AG em 3 instâncias no conjunto S1, sendo um desses resultados (S1-4) com valor médio no ótimo ($\text{Desv}(\%)=0.0$). Nas instâncias do conjunto S2, OEP1 e OEP2 superam os resultados do AG em 4 instâncias com 4 médias no valor ótimo (S2-0, S2-3, S2-7 e S2-8). O AG supera OEP1 e OEP2 em apenas 1 instância tanto em S1 quanto em S2. Nos demais resultados os métodos obtiveram os mesmos resultados. OEP1 e OEP2 superam AG em S3 com 2 resultados (S3-4 e S3-7) no valor ótimo. O AG obtém o melhor resultado em 1 instância no conjunto S3. Não há instâncias com solução ótima no conjunto S4, onde OEP1 e OEP2 superam AG em 7 instâncias. AG supera esses métodos novamente em uma única instância. Todos os 7 melhores resultados obtidos por OEP1 e OEP2 são iguais ou melhores que aqueles obtidos pelo AMPL/CPLEX.

A Figura 16 apresenta o desvio total médio obtido por cada método considerando os conjuntos de instâncias. Observe que OEP1 e OEP2 apresentam praticamente o mesmo desvio em todos os conjuntos e ambos são menores que o desvio apresentado pelo AG.

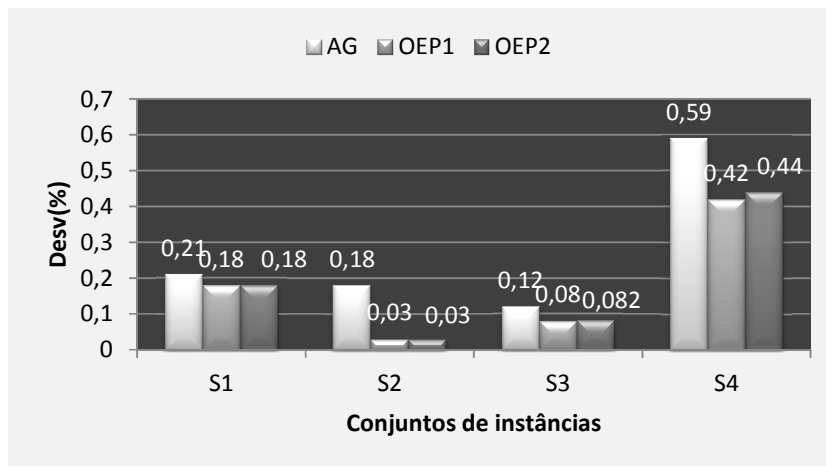


Figura 16: Desvio médio em S1, S2, S3 e S4.

4.3. Resultados em máquinas paralelas

O AMPL/CPLEX também foi utilizado em instâncias de máquinas paralelas com tempo de execução de 1 hora. As metaheurísticas foram executadas em 0.5 horas, ou seja, assim com nas máquinas simples pela metade do tempo do método exato. Um total de 10 execuções foi novamente efetuada sobre cada instância. As tabelas 7, 8 e 8 apresentam os resultados obtidos.

Tabela 7: Resultados obtidos para P1.

P1	AMPL	AG	OEP1	OEP2
P1-0	5819,8	4869,1	4241,3	4291,4
DesvAMPL(%)		-16.3	-27.1	-26.3
DP			44.39	45.6
P1-1	5561,0	4653,4	4169,6	4245,2
DesvAMPL(%)		-16.3	-25.0	-23.7
DP			80.5	84.5
P1-2	4371,5	4052,3	3650,5	3673,7
DesvAMPL(%)		-7.3	-16.5	-16.0
DP			34.6	36.39
P1-3	5043,8	4380,5	3998,4	4019,9
DesvAMPL(%)		-13.1	-20.7	-20.3
DP			45.47	45.81
P1-4	4536,9	4411,8	4153,3	4161,4
DesvAMPL(%)		-2.7	-8.5	-8.3
DP			67.72	69.12

Tabela 8: Resultados obtidos para P2.

P2	AMPL	AG	OEP1	OEP2
P2-0 DesvAMPL(%) DP	5730,2	4711,4 -17.8	3793 -33.8 118.85	38114 -33.5 97.56
P2-1 DesvAMPL(%) DP	5364,3	4913,7 -8.4	3899,6 -27.3 57.11	3900,4 -27.3 57.12
P2-2 DesvAMPL(%) DP	6375,3	5183,6 -18.7	4244,8 -33.4 58.82	4302 -32.5 58.78
P2-3 DesvAMPL(%) DP	7045,4	5326,4 -24.4	4091 -41.9 64.92	4126 -41.4 61.54
P2-4 DesvAMPL(%) DP	6666,0	5552,1 -16.7	4186,1 -37.2 105.68	4258,9 -36.1 101.67

Tabela 9: Resultados obtidos para P3.

P3	AMPL	AG	OEP1	OEP2
P3-0	5679,6	4711,4	3711,8	3797,7
DesvAMPL(%)		-17.0	-34.6	-33.1
DP			124.72	120.34
P3-1	7713,1	4913,7	3726,5	3863,3
DesvAMPL(%)		-36.3	-51.7	-49.9
DP			137.23	135.25
P3-2	7806,1	5183,6	3695,3	3865,9
DesvAMPL(%)		-33.6	-52.7	-50.5
DP			101.06	97.23
P3-4	7836,7	5552,1	3798,7	3973,4
DesvAMPL(%)		-29.2	-51.5	-49.3
DP			116.19	110.45
P3-5	7130,8	5530,8	3690,5	3771,9
DesvAMPL(%)		-22.4	-48.2	-47.1
DP			109.65	108.27

O AMPL/CPLEX não foi capaz de retornar soluções ótimas ao final de 1 hora de execução em P1, P2 e P3. Por outro lado, as metaheurísticas propostas superaram os valores das soluções finais obtidas pelo AMPL/CPLEX na maioria das instâncias. OEP1 e OEP2 superaram o AG em todas as instâncias do PGDLPP com máquinas paralelas. OEP1 obtém os melhores resultados em todos os conjuntos.

A Figura 17 apresenta os resultados médios para cada conjunto. O desempenho médio de OEP1 não é muito diferente do obtido por OEP2. Isso indica que a abordagem de ajuste dos pesos, durante a execução, representou uma pequena melhoria no desempenho do método OEP1. Por outro lado, as duas abordagens mativeram desempenho bastante superior ao AG.

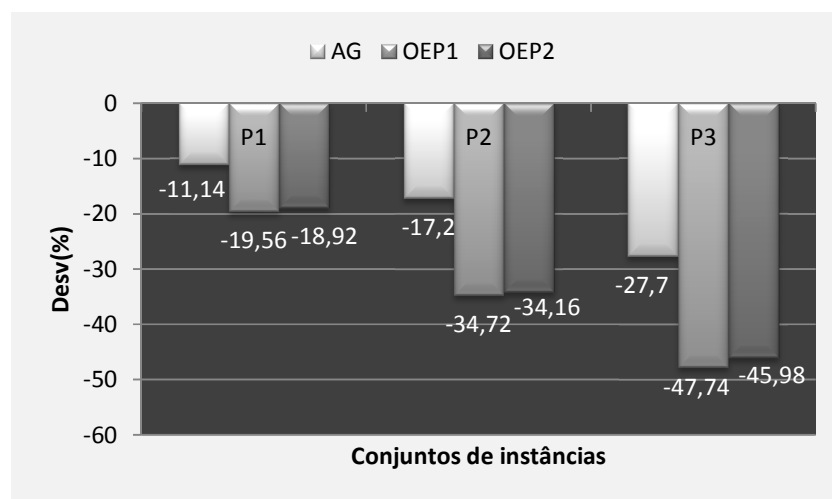


Figura 17 – Desvio médio em P1, P2 e P3.

5. CONCLUSÃO

O presente trabalho de conclusão de curso propôs métodos de resolução para o Problema Geral de Dimensionamento de Lotes e Programação da Produção (PGDLPP) com e sem máquinas em paralelo, e com penalização por demandas não atendidas.

Um modelo matemático foi apresentado e duas abordagens baseadas em otimização por enxame de partículas foram propostas. A primeira utiliza um ajuste dinâmico dos pesos dos vetores de posição das partículas (OEP1). A segunda é executada todo o tempo com os mesmos pesos. Um total de 7 conjuntos de instâncias são solucionados e comparados aos resultados obtidos pelo AMPL/CPLEX e por um algoritmo genético.

Nas instâncias do PGDLPP com máquinas simples, tanto OEP1 quanto OEP2 apresentam desvios abaixo de 0.5% em relação ao AMPL/CPLEX. Os métodos superam ou igualam a maioria dos resultados obtidos pelo AG. Não há uma diferença relevante de desempenho médio entre OEP1 e OEP2 nessas instâncias. Isso se repete durante a resolução dos conjuntos de instâncias do PGDLPP com máquinas paralelas. OEP1 consegue melhores resultados que OEP2 na maioria das instâncias, mas a diferença desses resultados na média de cada conjunto é pequena.

Dessa forma, o ajuste de pesos durante a execução melhorou o desempenho do método, mas não de forma expressiva. Os resultados obtidos permitem concluir que as duas abordagens baseadas em otimização por enxame de partícula conseguem melhores resultados que o

AG, anteriormente proposto para o mesmo problema. Além disso, o desempenho médio de OEP1 e OEP2 foi bastante superior ao AG principalmente nos conjuntos de instâncias mais complexos.

Trabalhos Futuros:

- Utilizar o OEP para determinar o dimensionamento dos lotes no.
- Acrescentar novas estratégias evolutivas às abordagens OEP1 e OEP2.

6. REFERÊNCIAS BIBLIOGRÁFICAS

ALI, M.M. & KAELO, P. *Improved particle swarm algorithms for global optimization.* Applied Mathematics and Computation Vol. 196, p.578-593, 2008.

ALOISE, D.J., OLIVEIRA, M.C.S. & SILVA, T.L. Otimização discreta por nuvem de partículas aplicada ao problema do caixeiro viajante. GEPROS – Gestão da Produção, Operações e Sistemas, Vol 2, p 87-95, 2006.

BAHL, H. C., RITZMAN, L. P., GUPTA, J. N. D. *Determining Lot Sizes and Resource Requirements: A Review.* Operational Research Society of America, v. 35, n. 3, p. 329-345, 1987.

BALDO, T. A. *Geração de colunas para o problema de dimensionamento de lotes de produção com limitações de capacidade.* Serviço de Pós-Graduação do ICMC-USP, 2009.

BILLINGTON, P. J., MCCLAIN, J. O., THOMAS, L. J. *Mathematical Programming Approaches to Capacity MRP Systems: Review, Formulation and Problem Reduction.* Management Science, v. 29, n. 10, p. 1126-1141, 1983.

BILLINGTON, P. J., BLACKBURN, J. D., MAES, J., MILLEN, R. A., VAN WASSEHOF, L. *Multi-Item Lotsizing in Capacitated Multi-stage Serial Systems.* IIE Transactions, v. 26, n. 2, p. 12-18, 1994.

BREVE, F. A. *Otimização por Colônia de Formigas (ASO) aplicadas ao problema do Caixeiro Viajante.* Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo, 2007.

BUENO, F. *Métodos Heurísticos – Teoria e Implementação,* IFSC, Araranguá, 2009.

BURBIDGE, J. L. *Planejamento e controle da produção*. 2.ed. São Paulo, Atlas. 556p, 1988.

CHEN, D., ZHAO, C. *Data-driven fuzzy clustering based on maximum entropy principle and PSO*. Pergamon Press, v36, pg:625-633, 2009.

CHIAVENATO, I. *Iniciação à Administração da Produção*. São Paulo: Makron, McGraw- Hill, Cap. 5, pp. 81-100, 1991.

COELHO, L. S., NETO, R. F. T. *Colônia de Formigas: Abordagem Promissora para Aplicações de Atribuição Quadrática e Projeto de Layout*. XXIV ENEGEP Florianópolis, SC. PUCPR, 2004.

CORDENONSI, A. Z., MULLER, F. M. & BASTOS, F. P. O Ensino de heurísticas e Metaheurísticas na área de Pesquisa Operacional sob a ótica da Educação Dialógica Problematizadora. CINTED-UFRGS, v3 num 1, maio, 2005.

CÔRREA, H. L., CÔRREA, C. A. *Administração de Produção e operações*. São Paulo: Atlas, 2004.

DIAZ, A., GLOVER, F., GHAZIRI, H. M., GONZÁLEZ, J. L., LAGUNA, M., MOSCATO, P., TSENG, F. T. *Optimización Heurística y Redes Neuronales*. Editorial Paraninfo, Espanha. 1996.

DREXL, A. & KIMMS, A. *Lot Sizing and Scheduling – Survey and Extentions*, European Journal of Operational Research, Vol. 99, p. 221-235, 1997.

DUTRA, F. A. F. *Análise do planejamento e controle da produção visto como um sistema complexo adaptativo*. Florianópolis, 2006. Disponível em:

<http://www.dominio.gov.br/pesquisa/PesquisaObraForm.do?select_action=&co_autor=8734> Acesso em: 15 janeiro de 2005.

EBERHART, R. C., KENNEDY, J. A. *New Optimizer Using Particles Swarm Theory*. Proc. Sixth International Symposium on Micro

Machine and Human Science. IEEE Service Center, Piscataway, NJ, 39-43, Nagoya, Japan, 1995.

ERDMANN, R. H. *Administração da produção: planejamento, programação e controle.* Florianópolis: Papa livro, 2000.

FERNANDES, A.M.R. *Inteligência Artificial: noções gerais.* 2 imp. Florianópolis: VisualBooks, 2005.

FILITTO, D. *Algoritmos Genéticos: Uma Visão Explanatória.* Revista Multidisciplinar da UNIESP, 2008

FLEISCHMANN, B. & MEYR, H. *The general lotsizing and scheduling problem.* ORSpektrum Vol. 19, p.11-21, 1997.

FRAGA, M. C. P. *Uma Metodologia Híbrida Colônia de Formigas - Busca Tabu - Reconexão por Caminhos para Resolução do Problema de Roteamento de Veículos com Janelas de Tempo.* Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), 2006.

GAMBARDELLA, L.M.; TAILLARD, É.D. & DORIGO, M. *Ant colonies for the quadratic assignment problem.* Journal of the Operational Research Society, Vol. 50, p. 167-176. 1999.

GELDERS L. F., NAMBIAR, J. M., et al. "A Large Scale Location-Allocation Problem in the Natural Rubber Industry". European Journal of Operational Research **6**: 183-189, 1981.

Glover, F. *Future paths for Integer Programming and Links to Artificial Intelligence.* Computers & Operations Research, v. 5, p. 553-549, 1986.

HAASE, K. *Capacitated lot-sizing with sequence dependent setup costs.* ORSpektrum Vol. 8, p.51-59, 1996.

HAESER, G, GOMES-RUGGIERO, M. *Aspectos Teóricos de Simulated Annealing e um Algoritmo duas Fases em Otimização Global.* TEMA Tend. Mat. Apl. Comput., 9, Num 3, 395-404. 2008.

Hansen, P. *The steepest ascent mildest descent heuristic for combinatorial programming.* Proceedings of Congress on Numerical Methods in Combinatorial Optimization, Capri, Itália. 1986.

Holland, J. H. *Adaptation in Natural and Artificial System.* University of Michigan Press. 1975.

JOSÉ, D, OLIVEIRA, M. C. S., SILVA, T. L. *Otimização discreta por nuvem de partículas aplicado ao problema do caixeiro viajante.* GEPROS num 2, p. 87-95.UFRN, 2006.

KANG, Y. & CULLEN, B. R. *The human Tap protein is a nuclear mRNA export factor that contains novel RNA-binding and nucleocytoplasmic transport sequences.* Genes Dev 13: 1126-1139, 1999.

Kennedy, J., Eberhart, R., Shi, Y., *Swarm Intelligence.* Academic Press, Inc. 2001.

KUBO, M. & KAKAZU, Y. *Simulating a competition for foods between ant colonies as a coordinated model of autonomous agents.* Proceedings of International Conference on Systems, Man, and Cybernetics, La Touquet, France, Vol. 5, p. 142-148. 1993.

KUIK, R., SALOMON, M., VAN WASSENHOVE, L. N. *Batching Decisions: Structure and Models.* European Journal of Operational Research, v. 75, p. 243-263, 1994.

MAES, J., VAN WASSENHOVE, L. N. *Capacitated Dynamic Lotsizing Heuristics for Serial Systems.* International Journal of Production Research, v. 29, n. 6, p. 1235-1249, 1991.

MEYR, H. *Simultaneous lotsizing and scheduling on parallel machines.* European Journal of Operational Research Vol. 139, p. 277-292, 2002.

MULLER, G. I., gÓMEZ, A. T. *Utilização da Busca Tabu para a Geração de um Modelo Aplicado ao Job-shpo Sheduling Problem Considerando um Sistema de Manufatura Flexível.* Programa

Internacional de Pós-Graduação em Computação Aplicada (PIPICA). UNISINOS, 2006.

NASCIMENTO, M. C. V. *Uma heurística GRASP para o problema de dimensionamento de lotes com múltiplas plantas*. 66 f. Dissertação (Mestrado em Ciências da Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo. 2007.

PLOSSL, G. W. *Production and inventory control - principles and techniques*. 2. ed. Englewood Cliffs, N. J., Prentice-Hall, Inc. 443p, 1985.

PRADO, J. R., SARAMAGO, S. F. P. *Otimização por Colônia de Partículas*. FAMAT em Revista - num 4. Universidade Federal de Uberlândia (UFU), 2005.

SCHOPF, E. C., SCHEPKE, C., SILVA, M. L. SILVA, P. F. *Avaliação de heurísticas de Melhoria e da Metaheurística Busca Tabu para Solução de PRV*. Centro de Eletrônica e Tecnologia, UFSM, 2004.

SICILIANO, A. V. *Algoritmo Genéticos e Particle Swarm Optimization e suas aplicações problemas de Guerra Eletrônica*. Diretoria de Sistemas de Armas da Marinha (DSAM). ITA (PPGAO), 2007.

SUCUPIRA, I. R. *Métodos Heurísticos Genéricos: Meta-Heurística e Hiper-Heurística*. Instituto de Matemática e Estatística, IME-USP.

disponível para acesso em 21/05/2010 endereço em <http://www.ime.usp.br/~igorrs/monografias/metahiper.pdf>

TIMÓTEO, G. T. S. *Desenvolvimento de um algoritmo genético para a resolução de timetabling*. 76 f. Monografia (Ciência da Computação) –

Departamento de Ciência da Computação, Universidade Federal de Lavras, Lavras, 2002.

TOLEDO, C.F.M., FERREIRA, J. E., SIMEONE, F. & ROSA, G.P. *Metaheurísticas Aplicadas ao Problema Geral de Dimensionamento de Lotes e Programação da Produção*. In: XLI SBPO Simpósio Brasileiro de Pesquisa Operacional, Porto Seguro, BA. 2009.

TUBINO, D. F. Manual de planejamento e controle da produção. 2. ed. São Paulo: Atlas, 2000.

TUBINO, D. F. *Manual de Planejamento e Controle da Produção*. São Paulo: Atlas. Cap. 1, pp. 23-32, 1987.

VON ZUBEN, F. J. *Computação Evolutiva: Uma Abordagem Pragmática*, DCA/FEEC/UNICAMP disponível para acesso em 08/01/2008 endereço em <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/tutorial/tutorialEC.pdf>.

ZACARELLI, S.B. *Programação e controle da Produção*. 5 ed. São Paulo, Livraria Pioneira. 292p, 1979.

ZATTAR, I. C. *Análise da Aplicação dos Sistemas Baseados no Conceito de Capacidade Finita nos Diversos Níveis da Administração da Manufatura*, Dissertação de Mestrado, UFSC, 2004.