HEURÍSTICA E LIMITES DUAIS PARA O PROBLEMA DE DIMENSIONAMENTO DE LOTES E SEQUENCIAMENTO COM TEMPOS DE PREPARAÇÃO DEPENDENTES DA SEQUÊNCIA

LÍVIO ANTÔNIO MELO FREIRE

HEURÍSTICA E LIMITES DUAIS PARA O PROBLEMA DE DIMENSIONAMENTO DE LOTES E SEQUENCIAMENTO COM TEMPOS DE PREPARAÇÃO DEPENDENTES DA SEQUÊNCIA

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Geraldo Robson Mateus

Belo Horizonte

Maio de 2011

Melo Freire, Lívio Antônio

F866h Heurística e Limites Duais para o Problema de Dimensionamento de Lotes e Sequenciamento com Tempos de Preparação Dependentes da Sequência / Lívio Antônio Melo Freire. — Belo Horizonte, 2011

xxiii, 77 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais

Orientador: Geraldo Robson Mateus

1. Dimensionamento de Lotes. 2. Sequenciamento. 3. Heurísticas. 4. Limites Duais. I. Título.

CDU 519.6*61 (043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Heurística e limites duais para o problema de dimensionamento de lotes e sequenciamento com tempos de preparação dependentes da sequência

LÍVIO ANTÔNIO MELO FREIRE

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. GERALDO ROBSON MATEUS - Orientador
Departamento de Ciência da Computação - UFMG

PROF. MAURÍCIO CARDOSO DE SOUZA
Departamento de Engenharia de Produção - EE- UFMG

PROF. MARTÍN GOMEZ RAVETTI
Departamento de Engenharia de Produção - EE - UFMG

Thiago F. Noronha

PROF. THIAGO FERREIRA DE NORONHA
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 18 de junho de 2010.

À minha esposa Camila Freire

Agradecimentos

Ao meu orientador, Geraldo Robson Mateus, pela serenidade com que conduziu todo o processo de elaboração desta dissertação. Sem a sua compreensão e valiosa colaboração certamente este trabalho não teria sido finalizado.

À minha mãe, Socorro Freire, que me deu a vida e me criou com imensa dedicação e amor. Sou inteiramente grato a ela por ter me ensinado o valor real do saber.

Ao meu pai, Manoel Conegundes, pelo amor e amizade de todas as horas.

À minha esposa, Camila Freire, por ter sido paciente, companheira e amiga leal. Sem o seu amor e presença constante (apesar da distância física) não teria suportado tanto tempo longe de casa. Muito obrigado, meu amor!

À minha família, especialmente às minhas tias Regina, Lúcia e Maria Lúcia e ao meu tio Alcino pelo carinho e apoio irrestrito.

Aos professores e amigos Marcos Negreiros, Airton Xavier e Gustavo Campos, pelos conselhos e orientações durante minha graduação na Universidade Estadual do Ceará.

Aos meus sogros, Albenir e Mazé, por terem cuidado da minha esposa enquanto estive ausente e pelo carinho de sempre.

Ao amigo e professor Thiago Noronha, pela companhia e conversas, especialmente sobre otimização, que sem dúvida contribuíram para a minha formação

À Jordânia Maciel, grande amiga que fiz em Belo Horizonte, pela disponibilidade, companheirismo e carinho.

Aos amigos e ex-colegas de FITec Ricardo Brito (principalmente pela compreensão nos momentos em que tive de me ausentar por conta das atividades acadêmicas), Marcus Lima e Gustavo Morais pelo ambiente legal de trabalho e a troca de experiências.

Aos professores e funcionários do DCC que, direta ou indiretamente, contribuíram para a realização desta dissertação.

À Capes pelo auxílio financeiro.

"Eu sou de uma terra que o povo padece
Mas não esmorece e procura vencer.
Da terra querida, que a linda cabocla
De riso na boca zomba no sofrer
Não nego meu sangue, não nego meu nome
Olho para a fome, pergunto o que há?
Eu sou brasileiro, filho do Nordeste,
Sou cabra da Peste, sou do Ceará."
(Patativa do Assaré)

Resumo

Este trabalho aborda os Problemas de Dimensionamento de Lotes e Sequenciamento em sistemas de produção monoestágio, que consiste na determinação dos níveis de produção ao longo de diversos períodos de tempo para atender às demandas de produção de um conjunto de produtos. O objetivo do problema é obter um plano que minimize os custos de produção, estoque, atraso e preparação das máquinas, que são oriundos do processo fabril. Considera-se um parque industrial composto por máquinas paralelas não relacionadas com tempos de preparação dependentes da sequência.

Um método heurístico é proposto para resolver o problema. A ideia básica do algoritmo consiste em gerar soluções para um problema de dimensionamento não capacitado que considera apenas um recurso e aplicar procedimentos que restabelecem o sequenciamento e as limitações de tempo. Em seguida, essas soluções iniciais viáveis são submetidas a um processo de melhoria baseado na metaheurística Busca Tabu. Para medir a qualidade das soluções, é proposta uma relaxação combinatória da qual se obtém limites duais para o problema.

Experimentos computacionais são realizados sobre instâncias geradas artificialmente. São construídos cenários de testes que consideram instâncias de pequeno e grande porte. Os resultados computacionais comprovam tanto a eficiência da heurística quanto do procedimento de geração de limites duais.

Palavras-chave: Dimensionamento de Lotes, Sequenciamento, Heurísticas, Limites Duais.

Abstract

This work addresses the lot sizing and scheduling integrated problem in single-level manufacturing systems, which consists in determining levels of production over various periods of time to meet production demands for a range of products. The objective of the problem is to get a plan that minimizes production, inventory, backlog and setup costs. We consider an industrial park consisting of unrelated parallel machines with sequence dependent setup times.

A heuristic method is proposed to solve the problem. The basic idea of the algorithm is to generate solutions to an uncapacitated lot-sizing problem with only one resource and implement algorithms that restore the scheduling and time constraints. These initial feasible solutions are subjected to a process of improvement based on a Tabu Search metaheuristic. To measure the quality of solutions, we propose a combinatorial relaxation that generates dual bounds for the problem.

Computational experiments are performed on random instances. The test scenarios are built to consider instances of small and large sizes. The computational results prove the efficiency of both heuristics as the procedure for generation of dual bounds.

Keywords: Lot Sizing, Scheduling, Heuristics, Dual Bounds.

Lista de Figuras

1.1 Exemplo de configuração final 5
1.2 Problema de Dimensionamento de Lotes Capacitado com Atraso 5
1.3 Integração 6
1.4 Sequenciamento 7

4.1 Junção de Subproblemas 30
4.2 Redefinição 31
4.3 Níveis de Produção 31
4.4 Transferência Progressiva 32
4.5 Transferência Regressiva 32

5.1 Problema PDLCAMP 54
5.2 Função do problema dual Lagrangeano 57

Lista de Tabelas

6.1 Limites das Instâncias 63
6.2 Análise dos Resultados 64
6.3 Resultados para o Problema PIDLS 67
6.4 Resultados para o Problema PDLCAMP 69
6.5 Comporação com os resultados de Belisário et al. [2009] 70

Sumário

Agradecimentos **ix**

Resumo **xiii**

Abstract **xv**

Lista de Figuras **xvii**

Lista de Tabelas **xix**

1 Introdução **1**
1.1 Motivação 1
1.2 Descrições e premissas 3
1.3 Definições dos problemas 8
1.4 Objetivos do Problema 9
1.5 Estrutura do Texto 9

2 Revisão Bibliográfica **11**
2.1 Características do Problema 11
2.1.1 Horizonte de Planejamento 11
2.1.2 Número de Níveis 12
2.1.3 Número de Produtos 12
2.1.4 Capacidade 12
2.1.5 Demanda 12
2.1.6 Estrutura de Preparação 13
2.1.7 Máquinas 13
2.2 Aspectos históricos 13

3 Formulações Matemáticas **19**
3.1 Problema de Dimensionamento de Lotes Capacitado com Atraso (PDLCA) 19

3.2 Problema de Sequenciamento em Máquinas Paralelas (PSMP) 21
3.3 Problema Integrado de Dimensionamento de Lotes e Sequenciamento (PIDLS) 23
3.4 Problema de Dimensionamento de Lotes com Atraso (PDLA) 25
3.5 Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas (PDLCAMP) 26
3.6 Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas (PDLAMP) 28

4 Abordagem Heurística para o Problema 29
4.1 Considerações Iniciais 29
4.2 Descrição do Procedimento 30
4.3 Algoritmo Exato para o Problema de Dimensionamento de Lotes com Atraso - PDLA 33
4.4 Solução Inicial 35
4.5 Reconstrução 36
4.5.1 Definição das Sequências 36
4.5.2 Viabilização das Capacidades 37
4.6 Melhoria 45
4.6.1 Transferência Regressiva - N_1 47
4.6.2 Transferência Progressiva - N_2 47
4.6.3 Busca Tabu 47
4.6.4 Algoritmo de Melhoria 49

5 Limites Duais para o PIDLS 53
5.1 Considerações Iniciais 53
5.2 Limites Primais 54
5.3 Limites Duais 54
5.3.1 Relaxação Linear 55
5.3.2 Relaxação Combinatória 55
5.3.3 Relaxação Lagrangeana 55
5.4 Relaxação Lagrangeana para o PDLCAMP 58

6 Resultados computacionais 61
6.1 Geração de Instâncias 62
6.2 Cenários de Teste 65
6.2.1 Avaliação das Instâncias Geradas Artificialmente 65
6.2.2 Comparação com a Literatura 68

7 Conclusões e Trabalhos Futuros **71**

Referências Bibliográficas **73**

Capítulo 1

Introdução

Este capítulo tem como objetivo discutir sobre as origens do tema e as principais motivações que levaram ao desenvolvimento do trabalho (seção 1.1), descrição e premissas sobre o problema abordado (seção 1.3) e os objetivos do problema resolvido na pesquisa (seção 1.4). Finalmente, descreve-se, resumidamente, o conteúdo dos demais capítulos (seção 1.5).

1.1 Motivação

Parcelas significativas do custo final de mercadorias derivam dos gastos gerados durante o processo de fabricação. Decisões que envolvem o quanto deve ser investido em matéria-prima, ampliação ou troca de tecnologia das linhas de produção e contratação de mão-de-obra são determinantes para reduzir o preço da carteira de produtos que uma empresa oferece. Para obter competitividade e aumentar lucros, várias medidas são tomadas nos níveis de influência de uma estrutura organizacional.

Tipicamente, organizações determinam um conjunto de metas a longo prazo que devem ser alcançadas no decorrer de um período de trabalho. Tais objetivos são acompanhados por um conjunto de ações globais que visam alavancar o negócio. Esse nível de decisão, conhecido como estratégico, é o mais elevado do processo de planejamento e nele atuam os altos executivos da empresa.

Quando deve haver atividade de produção e quanto deve ser fabricado para atender as ordens de compras ou estimativas de vendas são determinações que afetam toda a cadeia produtiva. Essas decisões a médio prazo, tomadas no nível tático de planejamento, são adotadas lavando-se em conta diversos aspectos restritivos, dentre os quais: as datas para a distribuição dos produtos finais aos clientes, na maioria dos casos, são rígidas e o não cumprimento acarreta prejuízos; vários recursos com capacidade limitada de tempo são consumidos (mão-de-obra, máquinas, etc.). As datas de entrega são agrupadas em um conjunto de períodos e o plano de produção é definido sobre a estimativa dessas capacidades. No final desta etapa, os níveis de produção, agrupados em lotes, atendem às limitações impostas, considerando como consumo de capacidade o tempo necessário para fabricar cada unidade do produto.

É preciso lidar com compartilhamento dos recursos antes de iniciar o processo de produção. Em tal situação deve existir uma política para selecionar a ordem que os produtos são apresentados para evitar, por exemplo, atribuições simultâneas de itens às linhas de fabricação. Além disso, a passagem de dois produtos pelo mesmo recurso pode exigir um consumo adicional de tempo por conta de tempos de preparação (reconfiguração de máquinas, limpeza para prevenir contaminação, etc.). Essa etapa de decisão a curto prazo compõe o nível de planejamento operacional. Essencialmente, necessita-se escalonar um conjunto de tarefas relativas aos lotes de produção levando-se em conta a capacidade real do período.

Grande parte da inconsistência na estimativa de custos e data de entrega dos lotes fabricados decorre das decisões nos níveis tático e operacional serem tomadas de maneira independente. Os sistemas de apoio à decisão costumam separar o planejamento de produção em duas fases:

1. São definidos níveis de estoque e produção para todos os períodos. O dimensionamento do que deve ser produzido obedece à estimativa de tempo de utilização dos recursos. É possível que o sistema não consiga definir os lotes dentro dessa estimativa, sendo necessário permitir que parte das demandas de alguns períodos sejam transferidas para períodos posteriores.

2. Os produtos fabricados são agrupados em lotes para reduzir os custos que incorrem devido à preparação dos recursos. Nesta fase, o sequenciamento dos lotes é definido e não deve ultrapassar a capacidade disponível. Porém, o tempo necessário para efetuar operações de preparação dos recursos e a capacidade de utilização real não são considerados na etapa anterior, consequentemente as quantidades dimensionadas geralmente ultrapassam a capacidade de utilização.

A inviabilidade gerada na etapa 2 é resolvida através de métodos heurísticos, os quais adotam estratégias que alternam entre redimensionamento e transferência de lotes entre períodos e rotinas de sequenciamento. Tais procedimentos encontram planos de produção viáveis capazes de satisfazer, muitas vezes, as requisições dos clientes. Por outro lado, devido ao planejamento inadequado, a previsão de custos é comprometida, uma vez que os níveis de produção, estoque e demanda transferida e preparação dos recursos possuem custos associados.

Por razões econômicas, encontrar soluções apenas viáveis não é suficiente. Neste contexto, visando definir planos de produção mais estáveis, surge a ideia de incorporar decisões de sequenciamento na fase de dimensionamento dos lotes de produção. Dessa maneira, quando os tamanhos dos lotes são definidos as capacidades reais e os tempos de preparação são considerados.

Assim, a motivação para esta dissertação é obter planos de produção com custos inferiores àqueles encontrados através de métodos que tratam os dois problemas separadamente.

1.2 Descrições e premissas

Esta dissertação apresenta mecanismos para se encontrar respostas a três questões fundamentais que ocorrem em sistemas de planejamento produção: quanto produzir, quando produzir e onde produzir. Essas decisões são tomadas visando à minimização de custos associados à produção, estoque, atraso e preparação do ambiente produtivo. Custos de produção referem-se ao gasto necessário para produzir uma unidade de qualquer produto; custos de estoque incorrem quando a produção acontece antes do momento demandado; custos de atraso são decorrentes do não atendimento da produção requerida na data definida para a entrega; custos de preparação são gerados sempre que é necessário produzir. A quantidade a ser produzida é determinada para atender as ordens de produção disparadas quando há pedidos de compra dos clientes (ou estimativas de venda). Todos os pedidos são agrupados numa única demanda de produção para cada produto. Um período determina um intervalo de tempo no qual se recebe um conjunto de requisições de produção que geram diversas demandas que devem ser entregues ao fim desse intervalo. O tamanho do intervalo de tempo representa uma restrição para o sistema de produção. Isto é, toda a atividade produtiva deve estar concentrada entre o início e o fim desse intervalo. No cenário em questão, dispõem-se de um número fixo de linhas de produção capazes de serem utilizadas para produzir qualquer produto em todos os períodos de tempo. Essas linhas de produção representam todos os recursos necessários à atividade produtiva (máquinas, matéria prima, mão-de-obra, etc). Os diversos produtos concorrem pela utilização dessas linhas para produzir suas demandas e consomem parcelas do tempo disponível sempre que qualquer quantidade da demanda é fabricada. De maneira simplificada, as linhas de produção podem ser consideradas como máquinas com recursos de tempo limitados.

As máquinas devem passar por uma etapa de preparação, que também consome recursos de tempo, antes de produzir qualquer item. A decisão de quando produzir deve ocorrer no período que for mais lucrativo para o sistema. Ou seja, pode acontecer da produção para atender certas demandas ser programada para ser fabricada em períodos anteriores àqueles onde as demandas foram geradas. Nessas situações, as demandas são satisfeitas através do estoque disponível no período. A realização da produção em períodos posteriores à geração da demanda é uma situação que ocorre, na maioria dos casos, apenas quando não há recursos disponíveis, já que atrasos no atendimento das demandas são situações indesejadas e devidamente penalizadas. Quando há escassez de tempo devido a utilização dos recursos, a produção pode ser direcionada a períodos anteriores ou posteriores, mesmo que custos adicionais incorram ao sistema. A quantidade produzida de fato, levando em conta o estoque ou atraso da produção, determina o tamanho dos lotes de produção que são enviados às máquinas para serem fabricados. Onde produzir refere-se à decisão de qual máquina deve receber a quantidade a ser fabricada. Máquinas diferentes possuem tempos e custos de produção e preparação diversificados, portanto a eleição das máquinas para produzir leva em conta esses parâmetros. Além disso, os tempos de preparação dependem também da sequência de alocação dos produtos. Sendo assim, onde produzir é uma decisão que envolve a escolha da máquina e sequência de produção. A figura 1.1 mostra um exemplo de configuração final das máquinas, períodos e lotes de produção quando essas três questões são solucionadas. São considerados 4 produtos, 3 períodos e 2 máquinas. A área escura representa o tempo gasto durante a preparação e a área clara refere-se ao tempo gasto durante a produção.

Muitas das estratégias adotadas para resolver o problema são baseadas na separação das decisões de quanto produzir e quando produzir das decisões de onde produzir. Neste contexto, é definido o *Problema de Dimensionamento de Lotes Capacitado com Atraso - PDLCA*, que oferece resposta às questões de quanto e quando produzir. A figura 1.2 mostra um exemplo do plano de produção definido para uma instância do PDLCA com 4 produtos e 3 períodos. O PDLCA considera apenas uma máquina com recursos finitos. Também são considerados tempos de preparação (área escura), que independem da sequência, e produção (área clara). Como há apenas uma máquina e não há sequenciamento, os tempos de preparação e produção do problema original precisam ser estimados quando se resolve o PDLCA e são, portanto, os mesmos em todos os períodos e independentes da sequência. Cabe também observar que é um problema dinâmico que considera diversos períodos.

O *Problema de Sequenciamento em Máquinas Paralelas - PSMP* resolve o problema de decidir onde produzir. Dessa forma, os lotes de produção definidos no pro-

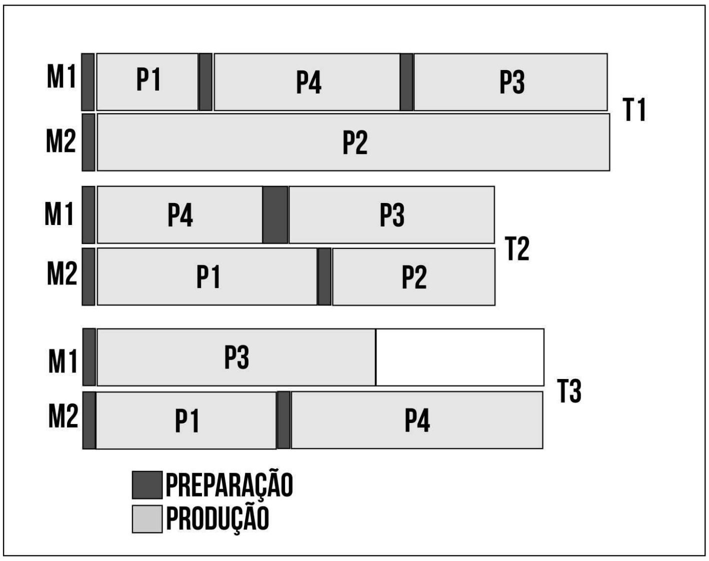

Figura 1.1. Exemplo de configuração final

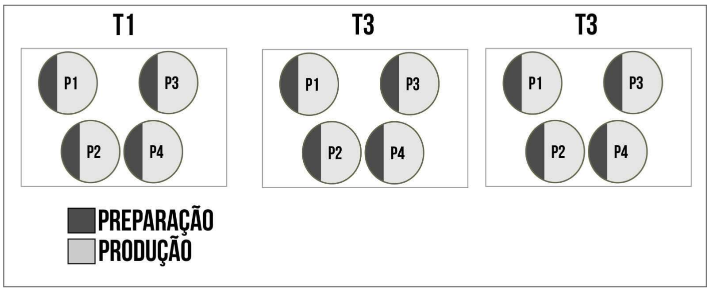

Figura 1.2. Problema de Dimensionamento de Lotes Capacitado com Atraso

blema PDLCA podem ser sequenciados, em todos os períodos, em alguma das máquinas disponíveis, levando-se em conta os tempos de preparação dependentes da sequência e das máquinas. A figura 1.3 mostra uma integração entre os problemas PDLCA e PSMP para a mesma instância com 4 produtos, 3 períodos e 2 máquinas.

Porém, pode acontecer que na resolução do problema PSMP a configuração final das máquinas não atenda as restrições de tempo do período considerado. Assim, após o sequenciamento de todos os períodos algumas máquinas podem apresentar excesso de produção. Existir excesso de produção significa que a dimensão dos lotes deve ser revista. Ou seja, é necessário transferir partes desses lotes para outro período com máquinas sem excesso. Pode acontecer de não existir nenhuma máquina com capacidade disponível para receber o lote. Nesse caso, o lote de produção deve ser transferido para um período posterior ou é cancelado se ocorre no último período (atrasado até o último período).

A figura 1.4 mostra uma configuração possível do produto T1 após o sequencia-

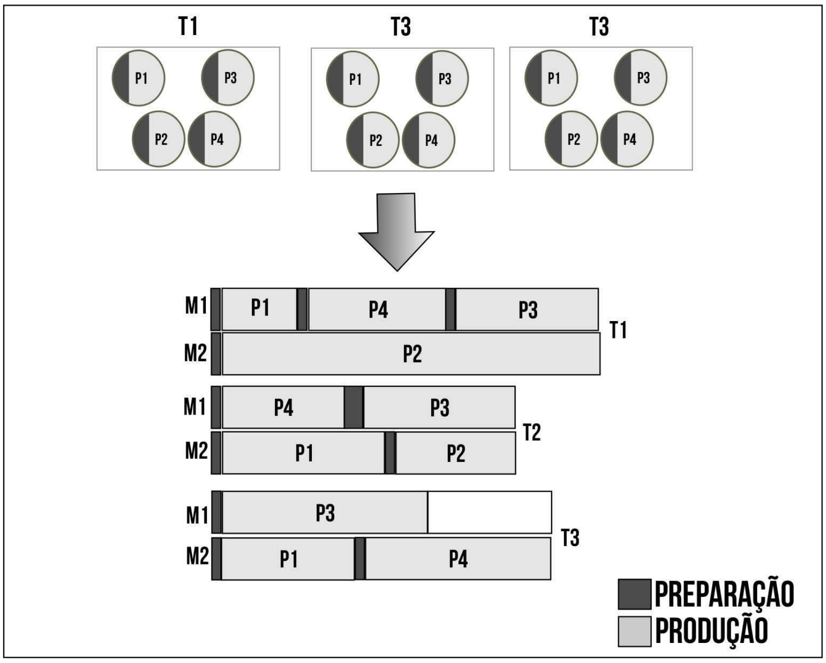

Figura 1.3. Integração

mento. Os produtos P3 e P4 apresentam excesso de produção, então os lotes de dos produtos das máquinas M1 e M2 devem ser transferidos até que o excesso seja anulado. Por fim, após a viabilização de todas as máquinas, o plano de produção está de acordo com as restrições do problema.

Esta dissertação apresenta a proposta de um procedimento integrado para resolver este problema. Dessa maneira, as decisões de quanto, quando e onde produzir são tomadas em conjunto. Ou seja, não há uma separação entre as decisões durante a geração do plano com menor custo. No método integrado, os tempos reais de produção e preparação são considerados em vez de estimativas. Assim, a capacidade real de tempo dos recursos é considerada durante todo o processo decisório. Espera-se que a utilização do procedimento integrado gere planos de produção com custos menores. Nesse contexto, é definido o *Problema integrado de Dimensionamento e Sequenciamento - PIDLS*, que combina os problemas PDLCA e PSMP.

Outros problemas são abordados nesta dissertação com objetivo de auxiliar na obtenção de soluções para o PIDLS. Esses problemas são descritos a seguir.

O *Problema de Dimensionamento de Lotes com Atraso - PDLA* é uma versão simplificada do PDLCA que considera máquinas com recursos infinitos. Nesse problema, as decisões de quando e quanto produzir são tomadas para cada produto de maneira independente. Como há recursos infinitos, não existem tempos de produção e preparação. O PDLA tem pouco interesse prático e uma das suas aplicações é servir como

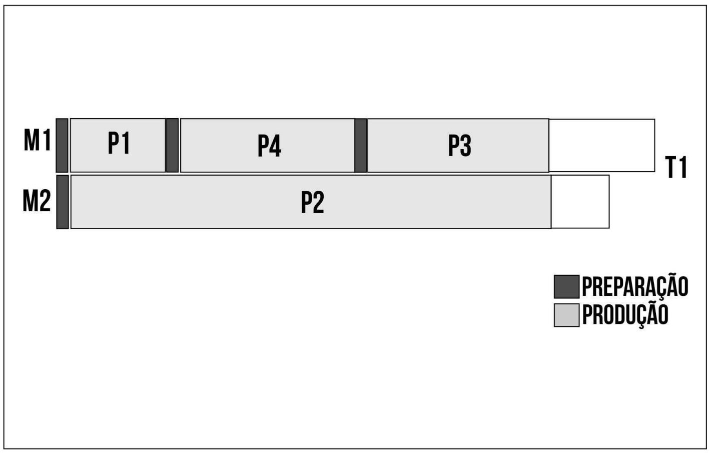

Figura 1.4. Sequenciamento

solução inicial durante a resolução de problemas mais complexos. Nesta dissertação, o PDLA é empregado como solução de partida no procedimento proposto para encontrar soluções viáveis para o PIDLS.

O *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas - PDLACMP* difere do PDLCA ao considerar máquinas paralelas e custos de preparação dependentes da máquina. Nesse problema, são consideradas estimativas dos tempos de preparação, uma vez que não há a definição da sequência de produção. Dessa forma, dependendo de como essa estimativa é definida, as soluções do PDLCAMP podem apresentar custos sempre inferiores aos da solução do PIDLS para uma mesma instância. Nesta dissertação, o PDLCAMP é empregado para obtenção das tais soluções (limites) inferiores.

O *Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas - PDLAMP* estende o PDLCA ao incorporar decisões de onde produzir. O PDLAMP também pode ser visto como uma versão do PDLCAMP com recursos infinitos, já que no PDLAMP também são consideradas máquinas paralelas e custos de preparação dependentes das máquinas. Porém, como também não há limitação de recursos nas máquinas, a decisão de onde produzir leva em conta apenas os custos de preparação. Assim como o PDLA, o PDLAMP também possui aplicações práticas limitadas e seu uso se restringe, basicamente, ao auxílio na resolução de problemas maiores. Nesta dissertação, o PDLAMP é empregado em um processo para obtenção de limites inferiores para o PDLCAMP e o PIDLS.

1.3 Definições dos problemas

Nesta dissertação, o problema tratado atende à necessidade de incorporação de decisões de sequenciamento durante a etapa de dimensionamento de lotes de produção. Considera-se o Problema de Dimensionamento de Lotes Capacitado com Atraso (PDLCA) integrado ao Problema de Sequenciamento em Máquinas Paralelas (PSMP). O problema integrado recebe a denominação de Problema Integrado de Dimensionamento Lotes e Sequenciamento (PIDLS).

São estabelecidas algumas premissas para o problema integrado:

- Há uma oferta de P produtos que podem ser produzidos;
- O parque industrial é formado por M máquinas aptas a fabricar qualquer produto;
- As entregas são realizadas apenas no final de cada um dos T períodos que dividem o horizonte de planejamento. A atividade produtiva deve estar contida entre o início e o fim de cada período.
- Há um conjunto de demandas para os produtos, cujos valores são conhecidos antes da definição do planejamento;
- Não há restrições quanto ao número de itens que podem ser fabricados no mesmo período;
- Uma unidade do produto final é obtida sem passos intermediários. Ou seja, a fabricação de um item começa e termina em apenas uma operação;
- As máquinas podem apresentar tecnologias diferentes. Portanto, os tempos de processamento são dependentes das máquinas;
- As máquinas precisam ser preparadas antes de iniciar a fabricação de um produto. Tal procedimento consome parte do tempo disponível e depende diretamente da natureza do último item que passou pela máquina. Também são considerados tempos que independem da sequência para os primeiros produtos fabricados no período;
- Há custos de preparação que influenciam a escolha da máquina para fabricar um item. Esses custos dependem do item e das peculiaridades da máquina;
- Os estoques e atrasos iniciais são considerados nulos;
- No final de um período, há níveis de estoque, produção e demanda transferida.

- Os custos de estoque são contabilizados de acordo com o número de períodos que os lotes ficam armazenados antes de serem entregues.

- A fabricação de um item consome recursos (energia, mão-de-obra, matéria-prima, etc.) cujos custos são contabilizados de acordo com o tamanho dos lotes;

- As demandas devem ser satisfeitas, preferencialmente, pelos níveis de produção ou estoque acumulado no período anterior. No entanto, devido às limitações de capacidade de tempo, pode ser necessário transferir parte da demanda para períodos posteriores. Todavia, essa situação pode gerar penalidade, onerando o custo final em função do número de períodos que o atraso se estende.

1.4 Objetivos do Problema

O problema tem como objetivo encontrar um plano de produção que atenda as demandas pelos produtos e não viole as limitações de capacidade dos períodos, levando-se em conta os tempos decorrentes das preparações das máquinas e de produção dos lotes. Os custos relativos à produção, estoque e demanda transferida, bem como preparação, devem ser minimizados.

1.5 Estrutura do Texto

Este texto está organizado como descrito a seguir. No capítulo 2 são apresentados alguns conceitos fundamentais necessários ao entendimento do restante da dissertação. Este capítulo também apresenta uma contextualização histórica dos problemas de Dimensionamento de Lotes e Sequenciamento.

No capítulo 3 o *Problema Integrado de Dimensionamento de Lotes e Sequênciamento (PIDLS)* é formulado a partir da incorporação ao modelo clássico para o *Problema de Dimensionamento de Lotes Capacitado com Atraso* das decisões de sequenciamento da formulação para o *Problema de Sequenciamento em Máquinas Paralelas (PSMP)*. Esse modelo, proposto inicialmente por Mateus et al. [2010], é ligeiramente modificado para se adequar às premissas do problema. No modelo resultante, são considerados tempos de preparação positivos para os primeiros produtos fabricados em cada período, situação não abordada no modelo original.

No capítulo 4 é apresentado um procedimento heurístico para resolver o PIDLS. Os passos da heurística são explicados detalhadamente, considerando etapas para a geração de uma solução inicial a partir de uma relaxação combinatória do problema, reincorporação das restrições removidas para gerar o problema relaxado, viabilização e busca local.

No capítulo 5 são apresentadas relaxações para o problema. Uma atenção especial é dada ao *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas (PDLCAMP)* por este ser uma relaxação combinatória para o PIDLS. Foi também desenvolvida uma Relaxação Lagrangeana para o PDLCAMP que gera limites duais para os dois problemas. O algoritmo para resolver essa relaxação, que é baseado no Método Subgradiente, é discutido em detalhes.

O capítulo 6 apresenta os resultados computacionais para os cenários de teste considerados. Para cada um desses cenários, seguem-se comentários sobre desempenho dos procedimentos desenvolvidos.

Capítulo 2

Revisão Bibliográfica

Este capítulo apresenta um conjunto de características úteis à classificação dos problemas de dimensionamento de lotes e sequenciamento quanto às dificuldades de resolução ou proximidade com a realidade (seção 2.1). Na seção 2.2 é apresentada a evolução histórica desses problemas, destacando-se o detalhamento dos problemas PDLA e PDLAMP (definidos na seção 1.2).

2.1 Características do Problema

Na literatura, encontra-se uma grande variedade de modelos para o problema de dimensionamento de lotes e sequenciamento. Algumas características determinam a complexidade desses problemas, além de aproximá-los à realidade ou dar a eles um enfoque mais teórico. Diante disso, [Karimi et al., 2003] e [Allahverdi et al., 2008] enumeram e discutem um conjunto de características para os problemas de dimensionamento de lotes e sequenciamento, respectivamente. Algumas dessas características são descritas nas seções a seguir.

Considerando as características abaixo descritas, o problema tratado trabalha com **horizonte de planejamento** finito, **períodos de tempo** grandes (*big bucket*), um **nível**, múltiplos **itens**, **capacidade** limitada, **demandas** dinâmicas, **estrutura de preparação** complexa e **máquinas** paralelas e não relacionadas.

2.1.1 Horizonte de Planejamento

É o espaço de tempo no qual a atividade produtiva se estende. Estes podem ser finitos ou infinitos. Os modelos de dimensionamento, quanto ao tamanho dos períodos, podem ser classificados em duas categorias: problemas de *small bucket* ou *big bucket*.

Problemas de *big buckets* são aqueles em que o período de tempo é grande o suficiente para produzir muitos produtos, enquanto para *small buckets* o período é tão pequeno que somente um produto – em algumas situações, dois produtos – pode ser produzido em cada período de tempo.

2.1.2 Número de Níveis

Sistemas de produção podem ser de um nível ou multi-níveis. Em sistemas de um nível, a produção do produto final costuma ser simples e realizada em apenas uma operação. Para esse tipo de sistema, as demandas são independentes, sendo oriundas diretamente de pedidos de clientes ou previsão de vendas. Em sistemas multi-níveis, há uma relação hierárquica entre os diversos componentes do produto final. As matérias primas são submetidas a várias operações até que o produto seja fabricado. A saída de uma operação (nível) alimenta a entrada de outra operação. Dessa maneira, as demandas de um nível dependem das demandas dos níveis que estão hierarquicamente acima. Nesse caso, a demanda é dita ser dependente. Problemas multi-níveis são mais difíceis de resolver que problemas de apenas um nível.

2.1.3 Número de Produtos

Em termos de número de produtos, os problemas de dimensionamento de lotes podem lidar com apenas um item ou com múltiplos itens. A complexidade de problemas multi-itens é maior que a complexidade daqueles com apenas um item.

2.1.4 Capacidade

As restrições de capacidade estão relacionadas com escassez de recursos (mão-de-obra, máquinas, energia elétrica, etc.). Quando os recursos são infinitos o problema é dito ser *não capacitado* e quando há limitações dos recursos o problema é dito ser *capacitado*. Problemas do primeiro tipo possuem relevância teórica ou surgem como subproblemas quando problemas do segundo tipo são resolvidos.

2.1.5 Demanda

São ordens de compra ou previsões de vendas que motivam a atividade produtiva. Há dois tipos de demanda: *estática*, quando os valores são constantes ou estacionários (mudam a uma taxa constante) ou *dinâmica*, quando os valores variam com o tempo. Se as demandas são conhecidas antes do planejamento, então são ditas *determinísticas*.

Caso contrário, se seus valores dependem de alguma distribuição de probabilidade, são ditas *probabilísticas*.

2.1.6 Estrutura de Preparação

Quando há alternância de itens durante a produção podem ocorrer custos e/ou tempos de preparação. Se esses tempos ou custos não dependem de itens ou períodos anteriores, a preparação apresenta uma estrutura simplificada; por outro lado, se há alguma dependência, a estrutura é dita ser complexa. O estado de preparação de uma máquina corresponde aos ajustes necessários para a máquina fabricar um certo produto.

2.1.7 Máquinas

As máquinas que formam o ambiente produtivo, no que se refere à velocidade para produzir um item, podem ser: *idênticas*, situação em que todas as máquinas são capazes de processar um determinado produto na mesma velocidade; ter velocidades diferentes, porém *uniformes*, nas quais a razão dos tempos para processar dois produtos quaisquer em um determinado par de máquinas representa um número constante; ou completamente *não relacionadas*, situação em que não há nenhuma relação conhecida entre as velocidades de processamento. Quanto à organização do ambiente produtivo, as maquinas podem ser: *dedicadas*, situação em que há apenas um subconjunto de máquinas capazes de processar cada um dos produtos; ou *paralelas*, em que todas as máquinas podem processar qualquer produto.

2.2 Aspectos históricos

O estudo sobre os problemas de dimensionamento de lotes iniciou-se com o modelo Quantidade Econômica do Pedido (QEP). O QEP considera o planejamento da produção de apenas um item, necessitando apenas uma operação (único nível) e sem restrições de capacidade (recursos infinitos). As demandas para cada item são estacionárias (ocorrem a taxas constantes) e acontecem continuamente durante um horizonte de planejamento infinito. A solução ótima para o problema pode ser encontrada sem muitas dificuldades [Erlenkotter, 1990; Harris, 1990].

Posteriormente, surgiram outros modelos a partir do QEP incorporando elementos para aproximá-los a situações mais realistas, tais como: horizonte de planejamento finito, limitação de capacidade, múltiplas máquinas, produção de vários itens no mesmo período e demanda dinâmica.

O problema de Programação de Lotes Econômico (PPLE) é um deles [Elmaghraby, 1978]. No PPLE há restrições de capacidade (recursos limitados) e produção de vários itens (multi-item) considerando operações atômicas (único nível). Entretanto, o PPLE também é um modelo contínuo com demandas estacionárias e horizonte de planejamento infinito. Encontrar a solução ótima para o PPLE é NP-difícil [Hsu, 1983] e métodos heurísticos são amplamente empregados [Dobson, 1987; Khouja et al., 1998].

O problema clássico de dimensionamento Wagner-Whitin (WW) diferencia-se do QEP por assumir demandas dinâmicas e horizonte de planejamento finito. No entanto, não há restrições de capacidade. O WW pode ser visto como um problema do caminho de custo mínimo numa rede, tal que os vértices e arcos estão associados, respectivamente, aos períodos e aos custos de produção entre os períodos. Sendo assim, a solução ótima para o WW pode ser encontrada por algoritmos polinomiais [Aggarwal & Park, 1993; Federgruen & Tzur, 1991; Wagelmans et al., 1992].

O problema de Dimensionamento de Lotes Capacitado (PDLC) estende o problema WW adicionando restrições de capacidade. O PDLC tem como objetivo minimizar a soma de custos de preparação e estoque. O PDLC é um problema *big bucket*, uma vez que é possível produzir vários produtos no mesmo período. Encontrar uma solução ótima para o PDLC é NP-difícil. Se tempos de configuração são incorporados ao modelo, decidir se há uma solução viável para o PDLC torna-se NP-completo [Maes et al., 1991]. Uma variação desse modelo é apresentada na seção 3.1.

O problema de Dimensionamento e Programação de Lotes Discreto (PDPLD) subdivide os períodos do PDLC em vários períodos menores. Se uma produção inicia-se em algum desses micro-períodos, esta deve consumir toda a capacidade disponível. Como só é possível fabricar um item por micro-período, o PDPLD é um problema de períodos curtos (*small bucket*). O PDPLD conserva o estado de preparação se ocorre a produção do mesmo produto em períodos vizinhos. Obter uma solução ótima para o PDPLD é NP-difícil [Brüggemann & Jahnke, 2000].

O problema de Dimensionamento com Preparação Contínua (PDPC) surge tendo como principal diferença para o PDPLD a não obrigação da produção consumir toda a capacidade do período. Diante disso, enquanto no PDPLD o estado de preparação não é conservado se houver micro-períodos ociosos, no PDPC o estado preparação só é alterado se houver produção de um produto de natureza diferente a do último produzido. Comparado ao PDPLD, o PDPC tem atraído pouco interesse em pesquisa [Drexl & Kimms, 1997]. Alguns estudos sobre o PDPC podem ser encontrados em [Karmarkar et al., 1987; Bitran & Matsuo, 1986]

O problema de Dimensionamento e Programação de Lotes Proporcional (PDPLP) [Drexl & Haase, 1995] é semelhante ao PDPC. No PDPLP a capacidade remanescente do período pode ser utilizada para produzir um segundo item. O PDPLP introduz sequenciamento a um modelo de período curto (*small bucket*), já que a ordem em que os produtos são produzidos no período deve estar clara devido à conservação do estado de preparação.

O Problema Geral de Dimensionamento e Sequenciamento (PGDS) integra os problemas de dimensionamento e sequenciamento de vários produtos em uma máquina com capacidade de tempo restrita, cujo objetivo é minimizar custos de estoque e preparação. Um estudo detalhado sobre o problema pode ser encontrado em [Fleischmann & Meyr, 1997]. Neste trabalho, são apresentados dois modelos, em que o primeiro não conserva o estado de preparação da máquina em situações em que ocorrem micro-períodos ociosos, enquanto essa propriedade é estabelecida no segundo modelo. No que diz respeito à complexidade computacional, no mesmo trabalho também foi provado que encontrar a solução ótima para esse problema é NP-difícil. Mais do que isso, provaram que decidir se há solução viável para o problema é NP-completo. Os autores também desenvolveram um procedimento heurístico baseado no algoritmo de busca local *threshold accepting* (TA) [Dueck & Scheuer, 1990] para problemas definidos conforme o primeiro modelo. O procedimento parte de uma solução inicial e busca por soluções candidatas sobre vizinhanças definidas através de operações simples (inserção, remoção e troca). Para evitar convergências prematuras, é permitido ao TA aceitar soluções piores se a diferença de qualidade não exceder um valor limitante, que é reduzido à medida que o procedimento evolui. Soluções iniciais são geradas a partir da fixação das variáveis de preparação. Quando essas variáveis são fixadas, o problema resultante é um problema de fluxo de custo mínimo, cuja resolução é simples. Três algoritmos heurísticos são gerados com base no TA: SIM, MOD e CAP. Os dois primeiros aceitam soluções não aprimorantes e superam resultados obtidas por Haase [1996]. Já o método CAP apresentou convergência rápida para mínimos locais, uma vez que apenas soluções aprimorantes são aceitas, e não obteve boas soluções.

Meyr [2000] estendeu o problema PGDS para lidar com tempos de preparação dependentes da sequência e definiu o Problema Geral de Dimensionamento e Sequenciamento com Preparação Dependente da Sequência (PGDSPDS). Para obter o modelo PGDSPDS, considerou-se tempos de preparação dependentes da sequência que reduzem a já escassa capacidade do período. Para resolver esse problema, também são propostos dois algoritmos heurísticos. O primeiro é semelhante ao TA mencionado acima. O segundo procedimento baseia-se na metaheurítica *Simulated Annealing* (SA) [Aarts & Van Laarhoven, 1987]. Para o PGDSPDS, o subproblema de fluxo em redes (PFR) gerado quando as variáveis de preparação são fixadas é reotimizado com base no seu problema dual através de um método proposto por Ali et al. [1989]. Além disso, utiliza-se os preços duais para rejeitar soluções candidatas (geradas através de pequenas pertubações no PFR) antes que o PFR modificado seja reotimizado. A aliança entre reotimização por métodos duais e as metaheurísticas TA e SA resultou em algoritmos eficientes com respeito à qualidade das soluções e ao tempo computacional.

Meyr [2002] modelou o Problema Simultâneo de Dimensionamento e Sequenciamento de Lotes em Máquinas Paralelas (PSDSLMP) como extensão do PGDSPDS ao permitir que os produtos sejam sequenciados em alguma das máquinas paralelas não relacionadas disponíveis. Neste trabalho, o autor avaliou a aplicação dos procedimentos adotados para resolver o PGDSPDS na resolução do problema PSDSLMP. Concluiu-se que o procedimento foi capaz de gerar soluções satisfatórias para as instâncias testadas, porém ineficiente no que se refere ao tempo computacional.

Sabbag Neto [1993] desenvolveu um procedimento heurístico para o Problema de Dimensionamento de Lotes Capacitado em Máquinas Paralelas (PDLCMP), que difere do PDLCAMP, essencialmente, por não permitir atrasos de produção. A proibição de atrasos acrescenta uma complexidade ao problema, pois agora apenas decidir se há soluções viáveis para o PDLCMP é um problema *NP-Completo*. Em um trabalho posterior, Toledo & Armentano [2006] desenvolveram uma heurística Lagrangeana para encontrar soluções viáveis para o PDLCMP e geraração de limitantes inferiores. A ideia adotada em ambos os procedimentos baseou-se na estratégia de dividir a busca por soluções viáveis e de boa qualidade em três etapas: geração de uma solução inicial, viabilização e melhoria. Esses trabalhos são de grande relevância para esta dissertação, pois os procedimentos adotados serviram de inspiração para as heurísticas aqui desenvolvidas.

Nas seções a seguir, são discutidos com mais detalhes os problemas de dimensionamento de lotes que apareces como subproblemas no método heurístico proposto nesta dissertação.

No capítulo 1 foi definido o PDLCA que será integrado ao PCMP. No entanto, pode ser definida uma versão não capacitada, que caracteriza o *Problema de Dimensionamento de Lotes com Atraso - PDLA*.

Trata-se de um problema de dimensionamento de lotes de produção com único item, capacidade irrestrita e permissão de atrasar a produção. O problema PDLA surge como subproblema na heurística proposta (capítulo 4).

O problema de dimensionamento de lotes não capacitado com apenas um item, uma máquina e um nível (WW), cujo objetivo é minimizar a soma dos custos de produção, estoque e preparação, foi largamente estudado e existem algoritmos polinomiais capazes de resolvê-lo de maneira exata. O primeiro método ótimo para o WW foi proposto por Wagner & Whitin [1958], que apresentou um algoritmo de programação dinâmica que necessita $O(T^2)$ operações. Posteriormente, Evans [1985] propôs uma implementação eficiente para o algoritmo de Wagner-Whitin que mostrou-se mais efetiva em termos computacionais, embora não tenha melhorado a sua complexidade teórica. Mais recentemente, os trabalhos independentes de [Aggarwal & Park, 1993; Federgruen & Tzur, 1991; Wagelmans et al., 1992] apresentaram algoritmos exatos cuja resolução é da ordem $O(T \log T)$. A complexidade pode ser ainda reduzida para $O(T)$ quando os custos de produção são não decrescentes.

Zangwill [1969] estendeu o modelo de Wagner-Whitin para permitir atrasos de produção (ou estoques negativos) que caracteriza o PDLA. O problema foi representado como um problema de fluxo de custo mínimo em redes e o algoritmo proposto necessita $O(T^2)$ operações para encontrar a solução ótima. Aggarwal & Park [1993]; Federgruen & Tzur [1991] também apresentaram algoritmos da ordem $O(T \log T)$ capazes de resolver o problema PDLA com atraso.

Da mesma forma que o WW, o PDLA pode ser representado por um problema de fluxo de custo mínimo em redes com apenas um sorvedouro. Assim, existe uma solução ótima que segue a propriedade de que cada nó da rede pode ter no máximo uma entrada positiva, ou seja, apenas uma das variáveis produção, estoque ou quantidade atrasada pode apresentar valor não nulo. Esse problema pode ser resolvido por um algoritmo de programação dinâmica que usa as equações recursivas propostas por Zangwill [1969], descritas na seção 4.3. Por sua vez, uma extensão do PDLA pode ser considerada para um conjunto de máquinas paralelas, em que custos de preparação e produção são dependentes das máquinas. Essa extensão, denominada *Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas - PDLAMP*, é explorado na definições dos limites duais no capítulo 5.

Capítulo 3

Formulações Matemáticas

Este capítulo apresenta formulações matemáticas para os problemas, já descritos na seção 1.2, abordados nesta dissertação: *Problema Integrado de Dimensionamento e Sequenciamento (PIDLS)* (seção 3.3), e o *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas (PDLCAMP)* (seção 3.5). Como o PIDLS representa a agregação de um problema de dimensionamento de lotes com outro de sequenciamento, as formulações para o *Problema de Dimensionamento de Lotes Capacitado com Atraso (PDLCA)* (seção 3.1) e o *Problema de Sequenciamento em Máquinas Paralelas (PSMP)* (seção 3.2) são também mostradas para contextualizar o modelo do PIDLS. Por fim, são formulados os demais problemas que surgem ao longo da dissertação como parte da resolução da heurística proposta ou procedimento para geração de limites duais, respectivamente: o *Problema de Dimensionamento de Lotes com Atraso (PDLA)* (seção 3.4) e o *Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas (PDLAMP)* (seção 3.6).

3.1 Problema de Dimensionamento de Lotes Capacitado com Atraso (PDLCA)

O problema de dimensionamento de lotes capacitado atende às necessidades do nível tático de planejamento ao definir os lotes de produção para um conjunto de períodos. A formulação clássica para o PDLC com custos de preparação é estendida com o acréscimo da permissão de atraso de fabricação de lotes quando a capacidade dos períodos não é suficiente para atender as demandas (modelo PDLCA).

Antes da formulação do PDLCA, seguem algumas definições:

- $p = 1 \ldots P$ são os produtos que podem ser produzidos em apenas uma máquina;

- $t = 1 \dots T$ correspondem a um conjunto de períodos que dividem o horizonte de planejamento;
- w_t representa a capacidade de tempo dos recursos no período t;
- e_p é capacidade em unidade de tempo consumida quando uma unidade do produto p é fabricada;
- q_p é o tempo necessário para preparar a máquina para produzir o produto p;
- d_{pt} é a demanda do produto p no período t;
- s_{pt} é a variável que representa a quantidade em estoque do produto p no final do período t;
- x_{pt} é a variável que representa a quantidade produzida do produto p no período t.
- z_{pt} é a variável que representa a quantidade de demanda atrasada do produto p no período t.
- f_{pt} é a variável binária que representa se houve preparação para o produto p no período t ($f_{pt} = 1$).

A seguir, a formulação matemática do PDLCA, que tem como objetivo minimizar custos de produção, estoque, atraso e preparação:

$$\min \sum_{t=1}^{T} \sum_{p=1}^{P} c_{pt}x_{pt} + h_{pt}s_{pt} + v_{pt}z_{pt} + u_{pt}f_{pt} \tag{3.1}$$

$$s_{p(t-1)} - s_{pt} + x_{pt} + z_{pt} - z_{p(t-1)} = d_{pt}, \quad \forall p, t \tag{3.2}$$

$$\sum_{p=1}^{P} e_p x_{pt} + q_p f_{pt} \leq w_t, \quad \forall t \tag{3.3}$$

$$x_{pt} - w_t f_{pt} \leq 0, \quad \forall p, t \tag{3.4}$$

$$f_{pt} \in \{0, 1\}, \quad \forall p, t \tag{3.5}$$

$$s_{pt}, x_{pt}, z_{pt} \geq 0, \quad \forall p, t \tag{3.6}$$

A função objetivo 3.1 minimiza custos de produção (c_{pt}), estoque (h_{pt}), atraso (v_{pt}) e preparação (u_{pt}). As restrições 3.2 garantem o balanço de fluxo entre produção, estoque e atraso. As restrições 3.3 asseguram que a capacidade disponível no período seja respeitada. O acoplamento entre produção e preparação da máquina é garantido pelas restrições 3.4, ou seja, $x_{pt} > 0$ apenas se $f_{pt} = 1$. As variáveis de preparação são definidas binárias nas equações 3.5. As inequações 3.6 são condições de não negatividade.

3.2 Problema de Sequenciamento em Máquinas Paralelas (PSMP)

O estudo sobre problemas de sequenciamento tem atraído uma quantidade considerável de pesquisas nos últimos tempos. Além das diversas aplicações práticas, obter soluções de qualidade para tais problemas, principalmente quando são adicionadas restrições que os aproximam de situações reais encontradas em sistemas de produção, é uma tarefa bastante complexa.

Como mencionando no capítulo 1, o sequenciamento dos produtos é aplicado no nível operacional de planejamento para definir a alocação das máquinas e a ordem em que os produtos são fabricados. Considerando sistemas hierárquicos, os lotes, cujas dimensões são definidas no nível tático, são vistos como um conjunto de tarefas que devem ser processados na última etapa do planejamento.

Brucker [2004] classificou os problemas de sequenciamento levando em conta o número, disposição e tipos das máquinas, características das tarefas e o critério para avaliar o desempenho do sequenciamento definido. Alguns levantamentos sobre o estado da arte dos problemas de sequenciamento e algoritmos de solução podem ser encontrados em [Ravetti, 2007; Cheng & Sin, 1990; Lawler et al., 1993; Mokotoff, 2001].

Considera-se um sistema de produção composto por um conjunto de máquinas paralelas não relacionadas, datas de entrega e tempos de preparação dependentes da sequência. Isto é, existe um parque industrial composto por máquinas com tecnologias diferentes (não relacionadas) e que estão sempre disponíveis para processar qualquer produto (paralelas). O processamento dessas tarefas deve ser finalizado (cancelamentos não são permitidos) dentro do prazo definido pela data de entrega. Eventuais atrasos são tolerados, desde que penalizados. Cada tarefa deve ser alocada a uma máquina e deve-se definir uma sequência de processamento que minimize um objetivo.

Rocha et al. [2008] modelaram o *Problema de Sequenciamento em Máquinas Paralelas*. O modelo, definido a seguir, atende às características do sistema de produção considerado nesta dissertação.

Levando-se em conta nesta dissertação as definições a seguir. Cabe observar que para algumas variáveis foram mantidas as mesmas letras, mas deixam de representar unidades de produtos para unidades de tempo.

- $p = 1 \dots P$ são os produtos que podem processados em qualquer máquina;
- $m = 1 \dots M$ são as máquinas disponíveis para processamento;
- e_{pm} o tempo necessário para processar a tarefa p na máquina m;
- d_p é a data de entrega para o produto P
- q_{ip}^m é tempo de preparação gasto ao mudar o estado de preparação do produto i para o produto p na máquina m;
- v_p é uma penalidade unitária por unidade de atraso no tempo do produto p;
- K é um número arbitrariamente grande;
- l_p é a variável que representa o tempo de início de processamento do produto p;
- f_p^m é a variável binária que representa que o produto p foi alocado à máquina m ($f_p^m = 1$);
- r_{ip}^m é a variável binária que representa se o produto i foi processado em um momento anterior ao produto p na máquina m ($r_{ip}^m = 1$);
- z_p é a variável que representa o tempo de atraso, em relação à data de entrega, do produto p;
- A finalização de processamento de um produto p na máquina m é dada por $l_p + e_{pm}$;

Segue a formulação para o PSMP, com objetivo de minimizar o tempo para finalizar todas as tarefas, ou *makespan*, e o atraso.

$$\min \quad Z + \sum_{p=1}^{P} (v_p z_p) \tag{3.7}$$

$$\sum_{m=1}^{M} f_p^m = 1, \quad \forall p \tag{3.8}$$

$$Z \geq l_p + e_{pm} - (1 - f_p^m)K, \quad \forall p, m \tag{3.9}$$

$$d_p + z_p \geq l_p + e_{pm} - (1 - f_p^m)K, \quad \forall p, m \tag{3.10}$$

$$(1 - f_i^m)K + (1 - f_p^m)K + (1 - r_{ip}^m)K + l_p \geq l_i + e_{im} + q_{ip}^m, \quad i < p, \forall p, i, m \tag{3.11}$$

$$(1 - f_i^m)K + (1 - f_p^m)K + r_{ip}^m + l_i \geq l_p + e_{pm} + q_{pi}^m, \quad p < i, \forall p, i, m \tag{3.12}$$

$$f_p^m, r_{ip}^m \in \{0, 1\} \tag{3.13}$$

$$l_p, z_p \geq 0, \forall p \tag{3.14}$$

A função objetivo (3.7) minimiza o tempo necessário para finalizar todas as tarefas (*makespan*) acrescido das penalizações relativas aos términos de processamento que excedem as datas de entrega. Cada produto deve ser processado em exatamente uma máquina (3.8). O *makespan* deve ser maior ou igual ao tempo de finalização de todos os produtos (3.9). Os tempos de finalização dos produtos devem ser inferiores as suas datas de entregas somadas aos possíveis atrasos (3.10). As restrições 3.11 asseguram que a tarefa p será processada na máquina m (apenas se $f_i^m = 1$ e $f_p^m = 1$), no mínimo, após a finalização da tarefa i, que antecede p, somada ao tempo de preparação (apenas se $r_{ip}^m = 1$). As restrições 3.12 são recíprocas às restrições 3.11. As variáveis de preparação e sequenciamento são definidas binárias nas equações 3.13. As inequações 3.14 são condições de não negatividade.

3.3 Problema Integrado de Dimensionamento de Lotes e Sequenciamento (PIDLS)

Mateus et al. [2010] propuseram um modelo matemático para o *Problema Integrado de Dimensionamento de Lotes e Sequenciamento (PIDLS)*. As decisões de sequenciamento da formulação para o problema PSMP (seção 3.2) são embutidas na formulação para o PDLCA (seção 3.1) com o objetivo de modelar as limitações reais de capacidade, uma vez que tempos de preparação dependentes da sequência são introduzidos.

Antes da formulação do PIDLS, seguem algumas definições:

- $p = 1 \dots P$ são os produtos que podem processados em qualquer máquina;
- $m = 1 \dots M$ são as máquinas disponíveis para processamento;
- $t = 1 \dots T$ correspondem a um conjunto de períodos que dividem o horizonte de planejamento;
- d_{pt}, s_{pt}, x_{pt} e z_{pt} têm a mesma interpretação do modelo apresentado na seção 3.1;
- e_{pm}, q_{ip}^m, K, l_{pt}, f_{pt}^m e r_{ipt}^m têm a mesma interpretação do modelo apresentado na seção 3.2, apenas estendendo os significados para os períodos (quando o índice t ocorre);
- Cada par (produto p - período t) gera uma tarefa que precisa ser processada num tempo compreendido entre $\sum_{\bar{t}=1}^{t-1} w_{\bar{t}}$ e $\sum_{\bar{t}=1}^{t} w_{\bar{t}}$, onde $w_{\bar{t}}$ é a capacidade do período $\bar{t}$ medida em unidades de tempo;

- $\beta_t = \sum_{\bar{t}=1}^{t} w_{\bar{t}}$.

A seguir, a formulação matemática do PIDLS, com objetivo de minimizar custos de produção, estoque, atraso e preparação em máquinas paralelas:

$$\min \sum_{t=1}^{T}\sum_{p=1}^{P} c_{pt}x_{pt} + h_{pt}s_{pt} + v_{pt}z_{pt} + \sum_{m=1}^{M} u_{pt}^{m} f_{pt}^{m} \tag{3.15}$$

$$s_{p(t-1)} - s_{pt} + x_{pt} + z_{pt} - z_{p(t-1)} = d_{pt}, \quad \forall p,t \tag{3.16}$$

$$x_{pt} - w_t \sum_{m=1}^{M} f_{pt}^{m} \leq 0, \quad \forall p,t \tag{3.17}$$

$$\sum_{m=1}^{M} f_{pt}^{m} \leq 1, \quad \forall p,t \tag{3.18}$$

$$l_{pt} + \left(1 - \sum_{m=1}^{M} f_{pt}^{m}\right)K - \sum_{m=1}^{M} a_{p}^{m} f_{pt}^{m} \geq \beta_{t-1}, \quad \forall p,t \tag{3.19}$$

$$l_{pt} + e_{pm}x_{pt} - (1 - f_{pt}^{m})K \leq \beta_t, \quad \forall p,t,m \tag{3.20}$$

$$(1 - f_{pt}^{m})K + (1 - f_{it}^{m})K + (1 - r_{pit}^{m})K + l_{it} - l_{pt} - e_{pm}x_{pt} \geq q_{pi}^{m}, \quad p < i, \forall p,i,t,m \tag{3.21}$$

$$(1 - f_{pt}^{m})K + (1 - f_{it}^{m})K + r_{pit}^{m}K + l_{pt} - l_{it} - e_{im}x_{it} \geq q_{ip}^{m}, \quad i < p, \forall p,i,t,m \tag{3.22}$$

$$s_{pt}, x_{pt}, z_{pt}, l_{pt} \geq 0, \forall p,t \tag{3.23}$$

$$f_{pt}^{m}, r_{ipt}^{m} \in \{0,1\}, \forall p,t,m,i \tag{3.24}$$

A função objetivo 3.15 minimiza os custos de produção (c_{pt}), estoque (h_{pt}) e atraso (v_{pt}), além de computar o custo preciso de preparação (u_{pt}^{m}) por máquina. As restrições 3.16 expressam o balanço de fluxo. As restrições 3.17 acoplam o dimensionamento de lotes às variáveis de preparação. As restrições 3.18 garantem que no mesmo período um produto pode ser processado no máximo por uma máquina. As restrições 3.19 e 3.20 asseguram que o início e término de processamento de um produto acontecem dentro dos limites do período e o sequenciamento considera os tempos de preparação dependentes da sequência e o tempo processamento dependente da máquina nas restrições 3.21 e 3.22. As variáveis de preparação e sequenciamento são definidas binárias nas equações 3.23. As inequações 3.24 são condições de não negatividade.

Os tempos de preparação $q_{pt}^{m} = q_{ip}^{m'} + a_{p}^{m}$ são definidos pela soma das parcelas q_{ip}^{m} e a_{p}^{m}, que possuem os seguintes significados:

- $q_{ip}^{m'}$ são tempos de preparação necessários para eliminar qualquer alteração no estado da máquina decorrente da passagem do produto i, que ocorre antes da fabricação produto p. Esses tempos acontecem em situações, por exemplo, onde há contaminação se houver produção de dois produtos distintos na mesma máquina sem que haja uma operação de limpeza.

- a_p^m são os tempos de preparação que incorrem antes do produto ser fabricado considerando que a máquina está pronta para recebê-lo. Tais tempos são gastos, por exemplo, para acertar parâmetros de configuração das máquinas;

No início de qualquer período as máquinas encontram-se prontas para receber qualquer produto, ou seja, não há memorização do estado de preparação entre dois períodos consecutivos. Assim, não há tempo de preparação dependente da sequência para o primeiro produto fabricado. O somatório $\sum a_p^m f_{pt}^m$ foi acrescentado na equação 3.19 da formulação original proposta por Mateus et al. [2010] para modelar essa situação. A soma do tempo a_p^m à marca de início do período β_{t-1}, caso o produto p seja fabricado na máquina m, indica que a fabricação de um produto consome ao menos um tempo de preparação independente da sequência. Não é difícil perceber que os tempos a_p^m só têm relevância para os produtos fabricados no início dos períodos, já que os tempos q_{ip}^m os dominam de cordo com as equações 3.21 e 3.22.

3.4 Problema de Dimensionamento de Lotes com Atraso (PDLA)

Nesta seção, é apresentada uma formulação matemática para o PDLS. Este modelo estende as formulação clássica para o problema WW (seção 2.2), principalmente, devido a incorporação da possibilidade de atraso.

Antes da formulação do problema, seguem algumas definições:

- d_t demanda no período t;
- $x_t \geq 0$ é a variável que representa a quantidade produzida no período t;
- f_t é a variável binária que representa a preparação da máquina no período t;
- s_t é a variável que representa a quantidade estocada no período t;
- z_t é a variável que representa a quantidade de produção em atraso no período t;
- K é um número arbitrariamente grande.

A seguir, uma formulação para PDLA, com objetivo de minimizar custos de produção, estoque, atraso e preparação:

$$\min \sum_{t=1}^{T} c_t x_t + h_t s_t + v_t z_t + u_t f_t \tag{3.25}$$

$$s_{(t-1)} - s_t + x_t + z_t - z_{(t-1)} \;=\; d_t, \quad \forall t \tag{3.26}$$

$$x_t \;\leq\; K f_t, \quad \forall t \tag{3.27}$$

$$x_t, s_t, z_t 0 \quad \forall t \tag{3.28}$$

A função objetivo 3.25 minimiza os custos de produção (c_t^m), preparação (u_t), estoque (h_t) e produção atrasada (v_t). As restrições 3.26 garantem o balanço de fluxo entre produção, estoque e atraso. O acoplamento entre produção e preparação da máquina é garantido pelas restrições 3.27. As inequações 3.28 são condições de não negatividade.

3.5 Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas (PDLCAMP)

O PDLCAMP refere-se ao planejamento da produção de produtos em diversas máquinas paralelas. O sistema considerado é mono-estágio e a capacidade é limitada em cada período do horizonte de planejamento. O atraso de produção é permitido, desde que haja penalização na função objetivo. Toledo & Armentano [2006] apresentam uma formulação para o PDLCAMP, aqui ligeiramente modificada ao considerar a capacidade como sendo particular do período, em vez de considerar como característica da máquina.

Os dados e as variáveis do problema são definidos a seguir:

- $p = 1 \dots P$ são os produtos que podem processados em qualquer máquina;

- $m = 1 \dots M$ são as máquinas disponíveis para processamento;

- $t = 1 \dots T$ correspondem a um conjunto de períodos que dividem o horizonte de planejamento;

- d_{pt}, s_{pt}, x_{pt}^m, z_{pt}, f_{pt}^m, e_{pm}, w_t e K têm a mesma interpretação do modelo apresentado seção 3.3, apenas estendendo os significados das variáveis de produção x_{pt} para as máquinas;

- a_p^m é o tempo necessário para preparar a máquina m antes de fabricar o produto p.

Segue uma formulação para o PDLCAMP, que tem como objetivo minimizar custos de produção, estoque, atraso e preparação em máquinas paralelas:

$$\min \sum_{p=1}^{P}\sum_{t=1}^{T}\sum_{m=1}^{M} c_{pt}x_{pt}^m + u_{pt}^m f_{pt}^m + \sum_{p=1}^{P}\sum_{t=1}^{T} h_{pt}s_{pt} + v_{pt}z_{pt} \tag{3.29}$$

$$\sum_{m=1}^{M} x_{pt}^m + s_{p(t-1)} + z_{pt} = d_{pt} + s_{pt} + z_{p(t-1)}, \quad \forall p,t \tag{3.30}$$

$$\sum_{p=1}^{P} e_{pm}x_{pt}^m + a_p^m f_{pt}^m \leq w_t, \quad \forall m,t \tag{3.31}$$

$$x_{pt}^m \leq K f_{pt}^m, \quad \forall p,t,m \tag{3.32}$$

$$\sum_{m=1}^{M} f_{pt}^m \leq 1, \quad \forall p,t \tag{3.33}$$

$$f_{pt}^m \in \{0,1\} \tag{3.34}$$

$$x_{pt}^m, s_{pt}, z_{pt} \geq 0 \quad \forall p,t,m \tag{3.35}$$

A função objetivo minimiza os custos de produção (c_{pt}), estoque (h_{pt}), atraso (v_{pt}) e preparação (u_{pt}^m). De acordo com as restrições 3.30, as variáveis x_{pt}^m representam a produção do período e possuem a mesma interpretação das variáveis x_{pt} do problema PIDLS, já que os custos de produção não dependem das máquinas. As restrições 3.30, 3.32 e 3.33 possuem equivalentes no modelo para o PIDLS com o mesmo significado. As restrições 3.31 consideram máquinas paralelas, mas com tempos de preparação a_p^m que não dependem da sequência. As variáveis de preparação são definidas binárias nas equações 3.34. As inequações 3.35 são condições de não negatividade.

O PDLCAMP é uma relaxação combinatória para o problema PIDLS, uma vez que apenas os tempos de preparação independentes da sequência a_p^m são considerados. Dessa maneira, as equações que modelam a limitação de tempo 3.31 são sempre mais folgadas que a combinação entre as equações 3.19, 3.20, 3.21 e 3.22 (ver formulação da seção 3.3) quando são retirados os tempos de preparação dependentes da sequência.

3.6 Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas (PDLAMP)

Quando as restrições de capacidade do PDLCAMP são relaxadas é definido um problema PDLAMP para cada produto. Como não há limitações de recursos de tempo, no problema resultante são descartados os tempos de produção e preparação.

Os dados e as variáveis do problema são definidos a seguir:

- $m = 1 \dots M$ são as máquinas disponíveis para processamento;
- $t = 1 \dots T$ correspondem a um conjunto de períodos que dividem o horizonte de planejamento;
- d_t, s_t, x_t^m, z_t, f_t^m e K têm a mesma interpretação do modelo apresentado seção 3.5;
- a^m é o tempo necessário para preparar a máquina m antes de fabricar o produto.

Segue uma formulação para o PDLAMP, que tem como objetivo minimizar custos de produção, estoque, atraso e preparação em máquinas paralelas:

$$\min \sum_{t=1}^{T} \sum_{m=1}^{M} c_t x_t^m + u_t^m f_t^m + \sum_{t=1}^{T} h_t s_t + v_t z_t \tag{3.36}$$

$$\sum_{m=1}^{M} x_t^m + s_{(t-1)} + z_t = d_t + s_t + z_{(t-1)}, \qquad \forall p, t \tag{3.37}$$

$$x_t^m \leq K f_t^m, \qquad \forall p, t, m \tag{3.38}$$

$$\sum_{m=1}^{M} f_t^m \leq 1, \qquad \forall p, t \tag{3.39}$$

$$f_t^m \in \{0, 1\} \tag{3.40}$$

$$x_t^m, s_t, z_t \geq 0 \qquad \forall p, t, m \tag{3.41}$$

A função objetivo minimiza os custos de produção (c_t), estoque (h_t), atraso (v_t) e preparação (u_t^m). As restrições 3.37 garantem o balanço de fluxo entre produção, estoque e atraso. O acoplamento entre produção e preparação da máquinas é garantido pelas restrições 3.38. As restrições 3.39 garante que a produção do período ocorre em no máximo uma máquina. As variáveis de preparação são definidas binárias nas equações 3.40. As inequações 3.41 são condições de não negatividade.

Capítulo 4

Abordagem Heurística para o Problema

Neste capítulo é apresentado um procedimento heurístico para resolver o *Problema Integrado de Dimensionamento de Lotes e Sequenciamento - PIDLS*, já descrito na seção 3.3. Tal estratégia de solução é composta por etapas que envolvem a geração de uma solução inicial (seção 4.4), viabilização do dessa solução (seção 4.5) e busca local (seção 4.6). O algoritmo de melhoria foi elaborado tendo como base a metaheurística Busca Tabu, que é introduzida na seção 4.6.3.

4.1 Considerações Iniciais

O método, inspirado no trabalho de Sabbag Neto [1993], é composto por três etapas:

1. obtenção de uma solução inicial;
2. reconstrução para uma solução do problema PIDLS viável;
3. melhoria iterativa.

A estratégia baseia-se na relaxação das restrições de capacidade e sequenciamento do PIDLS, de tal que forma que o problema é decomposto em P subproblemas, sendo um para cada produto. Cada um dos subproblemas consiste resolver o *Problema de Dimensionamento de Lotes com Atraso* (PDLA), cuja solução determina o planejamento de produção para todos os produtos/períodos, considerando recursos infinitos e apenas uma máquina (etapa 1). Com os níveis de produção definidos para cada produto/período, na etapa 2, define-se o sequenciamento dos produtos em cada máquina.

O sequenciamento definido é geralmente inviável quanto às restrições de capacidade, sendo necessário a aplicação de procedimentos que envolvem transferência de lotes entre os períodos para estabelecer a viabilidade. Na etapa 3, uma solução viável é submetida a um processo de busca que visa atingir um ótimo local de boa qualidade.

4.2 Descrição do Procedimento

Na fase inicial do método de solução, o problema é relaxado e decomposto em subproblemas sem restrições de capacidade de tempo. Dessa maneira, os tempos de produção e preparação são descartados. Os custos de produção são estimados como sendo o custo mínimo para preparar uma das máquinas para o produto. São gerados P subproblemas (um para cada produto). A solução ótima para os subproblemas é encontrada através da resolução do *Problema de Dimensionamento de Lotes com Atraso*. Esse procedimento compõe a primeira fase do método de solução e responde pela atribuição das quantidades iniciais de produção para cada produto e em todos os períodos.

A figura 4.1 representa o planejamento final para o problema relaxado após a resolução dos subproblemas. No exemplo, há 4 produtos e 3 períodos.

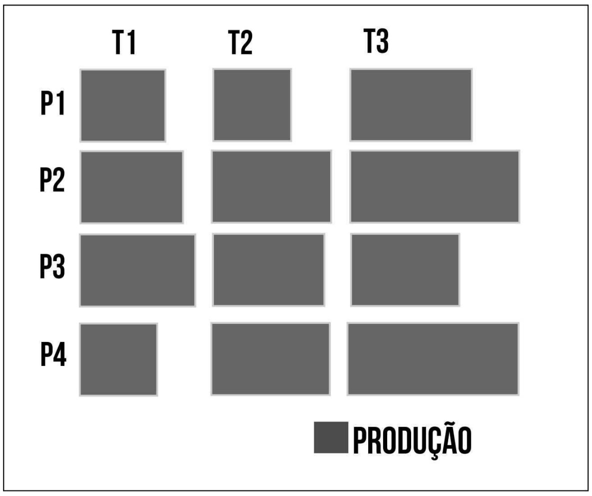

Figura 4.1. Junção de Subproblemas

A fase seguinte do procedimento consiste em retomar as restrições de capacidade. Nesse caso, os tempos de produção e preparação são reconsiderados, sendo necessário definir a sequência dos produtos nas máquinas, uma vez que a estimativa de custo de configuração força a escolha da alocação inicial das máquinas. Para definir o sequenciamento dos produtos em cada uma das máquinas é adotado um procedimento guloso

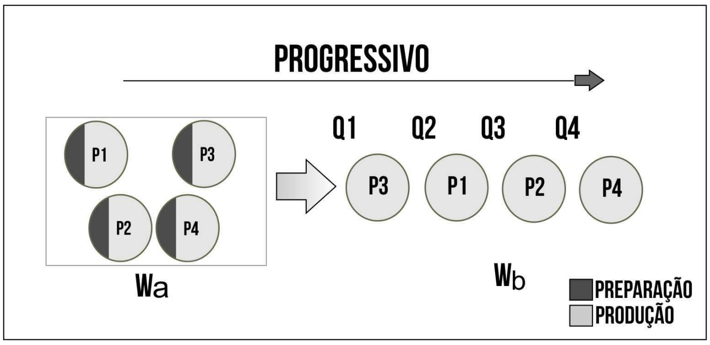

Figura 4.2. Redefinição

que ordena os produtos já alocados a elas. Esse procedimento segue progressivamente do período inicial até o final e determina a ordem dos produtos em todas as máquinas.

A figura 4.2 mostra a ação do procedimentos sobre os lotes gerados na primeira fase da heurística. No exemplo, são considerados 4 produtos. Os tempos de preparação são levados em conta para a definição da sequência.

Após o sequenciamento, as máquinas podem apresentar excessos de produção. Então, aplica-se um procedimento de viabilização das capacidades. Esse procedimento transfere regressivamente ou progressivamente parcelas dos lotes de produção entre períodos. Os procedimentos de transferência, assim como em [Sabbag Neto, 1993], foram baseados nas estratégias desenvolvidas por Trigeiro et al. [1989] para o Problema de Dimensionamento de Lotes Capacitado.

A figura 4.3 mostra o planejamento de produção de um dos produtos de um plano composto por 5 períodos. As figuras 4.4 e 4.5 exemplificam as transferências regressiva e progressiva, respectivamente.

Período	1	2	3	4	5
Demanda	20	40	35	50	60
Produção	25	32	35	58	55
Estoque	5	0	0	5	0
Atraso	0	3	3	0	0

Figura 4.3. Níveis de Produção

A figura 4.4 mostra a transferência de 6 unidades do período 4 para o período 2. Esse tipo de transferência incrementa a produção do período que recebe o lote e aumenta os estoques dos períodos compreendidos entre o 2 (inclusive) e o 4 ou reduz o atraso dos mesmos, alterando o valor da função objetivo.

Período	1	2	3	4	5
Demanda	20	40	35	50	60
Produção	25	38	35	52	55
Estoque	5	3	3	5	0
Atraso	0	0	0	0	0

Figura 4.4. Transferência Progressiva

Período	1	2	3	4	5
Demanda	20	40	35	50	60
Produção	25	26	35	64	55
Estoque	5	0	0	5	0
Atraso	0	9	9	0	0

Figura 4.5. Transferência Regressiva

A figura 4.5 mostra a transferência de 6 unidades do período 4 para o período 2. Esse tipo de transferência incrementa a produção do período que recebe o lote e aumenta o atraso dos períodos compreendidos entre o 2 (inclusive) e o 4 ou reduz o estoque dos mesmos, alterando o valor da função objetivo.

São adotadas políticas para determinar a máquina que receberá um lote transferido regressivamente ou progressivamente. Tais políticas procuram balancear as variações na função objetivo e na ocupação das máquinas. A quantidade transferida é sempre a máxima que a máquina destino é capaz de receber, desde que essa quantidade não seja superior ao tamanho do lote. Por fim, tem-se um planejamento de produção conforme as restrições do problema. Apesar da viabilidade da solução, é possível ainda melhorá-la. Para tal, é efetuado um procedimento heurístico que alterna entre as transferências regressivas e progressivas, além de modificar o sequenciamento das máquinas recebem lotes transferidos, com o objetivo de reduzir o custo total. Esse procedimento compõe a última etapa do procedimento.

4.3 Algoritmo Exato para o Problema de Dimensionamento de Lotes com Atraso - PDLA

Esta seção descreve um algoritmo de programação dinâmica para resolver o problema de dimensionamento de lotes com atraso (PDLA) (modelo da seção 3.4).

Seja Ω_t o custo mínimo de produção do período 1 ao t. Sendo assim, Ω_t pode ser determinado recursivamente por:

$$\Omega_t = \begin{cases} \min\limits_{0 \leq k < t} \Omega_k + \Psi_{kt}, & \text{para} \quad t = 1 \dots T \\ 0, & \text{para} \quad t = 0 \end{cases} \tag{4.1}$$

Tal que Ψ_{kt} define o custo mínimo para satisfazer toda a demanda entre os períodos $k+1$ e t em algum período j contido neste intervalo $(k+1 \leq j \leq t)$ e pode ser calculado como:

$$\Psi_{kt} = \begin{cases} f_t + c_t d_t & \text{se} \quad t = k+1 \\ \min\limits_{k+1 \leq j \leq t} \left[f_j + c_j \left(\sum\limits_{i=k+1}^{t} d_i \right) + \sum\limits_{i=k+1}^{j-1} v_i z_i + \sum\limits_{i=j}^{t-1} h_i s_i \right], & \text{se} \quad t > k+1 \end{cases} \tag{4.2}$$

Um algoritmo que adapta as melhorias sugeridas por Evans [1985] foi desenvolvido por Toledo & Shiguemoto [2005] para resolver de maneira eficiente as equações descritas acima.

```
     Entrada: T, c_t, h_t, v_t e f_t
     Saída: Solução ótima para o problema
1.1  D_0 ← 0;
1.2  para t = 1 até T faça
1.3      M_t ← h_t + c_t d_t;
1.4      D_t ← D_{t-1} + d_t;
1.5      Acum_t ← 0;
1.6  Ω_0 ← 0;
1.7  Ω_1 ← M_1;
1.8  para t = 2 até T faça
1.9      F_t = ∞;
1.10     para k = 1 até t − 1 faça
1.11         Acum_k ← Acum_k + h_{t-1};
1.12         M_k ← M_k + d_t(c_k + Acum_k);
1.13     para k = t − 1 até 0 faça
1.14         se t = k + 1 então
1.15             F ← M_t;
1.16         senão
1.17             F ← ∞;
1.18             para j = k + 1 até t faça
1.19                 Prov ← M_j + c_j(D_{j-1} − D_k);
1.20                 para i = j − 1 até k + 1 faça
1.21                     Prov ← Prov + v_i(D_i − D_k);
1.22                     se Prov < F então
1.23                         F ← Prov;
1.24
1.25     se F + Ω_k < Ω_t então
1.26         Ω_t ← Ω_k + F;
1.27
```

Algoritmo 1: Algoritmo Exato para o PDLA

O algoritmo 1 calcula de maneira eficiente as equações 4.1 e 4.2. Entre as linhas 1.2 e 1.5, as estruturas de dados são inicializadas. A equação 4.1 é calculada, progressivamente, entre as linhas 1.8 e 1.27. As melhorias sugeridas por [Evans, 1985] são introduzidas entre as linhas 1.10 e 1.12, nas quais a matriz M atualiza os custos de estoque acumulado aproveitando os valores previamente calculados. Em cada etapa da recursão, a equação 4.2 é calculada entre as linhas 1.13 e 1.24. Apesar da complexidade teórica desse algoritmo ser $O(T^2)$ e existirem procedimentos de solução para o problema com complexidade $O(T \log T)$, o mesmo foi escolhido pela facilidade de implementação e por apresentar eficiência em *benchmarks* disponíveis em Toledo & Shiguemoto [2005].

4.4 Solução Inicial

Quando as restrições de capacidade e sequenciamento são retiradas e o PIDLS é decomposto, para cada produto $p = 1 \dots P$ tem-se o seguinte problema:

$$Q_p = \min \sum_{t=1}^{T} c_{pt}x_{pt} + h_{pt}s_{pt} + v_{pt}z_{pt} + u'_{pt}f_{pt} \tag{4.3}$$

Sujeito às restrições 3.26 e 3.27.

A remoção das restrições de capacidade do PIDLS possibilita que os subproblemas sejam transformados em problemas de dimensionamento de lotes com apenas uma máquina. Ora, como os custos de produção no PIDLS independem da máquina que fabrica os produtos e a capacidade dos subproblemas é irrestrita, os custos de preparação, para cada período, podem ser definidos como:

$$u'_{pt} = u^m_{pt}, m \in M \quad \forall t \tag{4.4}$$

Conforme a equação 4.4, deve-se adotar alguma política para definir de qual máquina será considerado o custo de preparação. Assim, os problemas Q_p podem ser resolvidos através do algoritmo 1.

O algoritmo para procedimento *obterSolucaoInicial()* que retorna uma solução inicial, possivelmente inviável, para o PIDLS é descrito a seguir:

Entrada: Instância $Inst$ de um problema PIDLS
Saída: Níveis de produção e alocação das máquinas
2.1 $Sol \leftarrow \emptyset$;
2.2 **para** $p = 1$ **até** P **faça**
2.3 $Q_p \leftarrow definirInstanciaPDLA(Inst,\ p,\ politica)$;
2.4 $Sol_p \leftarrow resolverPDLA(Q_p)$;
2.5 $Sol \leftarrow Sol \cup Sol_p$;
2.6 **retorna** Sol;

Algoritmo 2: Obtenção de Solução Inicial para o PIDLS

Na linha 2.3 os custos para o produto p são extraídos para compor a instância do subproblema Q_p. São definidos problemas para cada produto e os costos de preparação são definidos de acordo com alguma política. O problema PDLA é resolvido para cada instância Q_p (linha 2.4) e o resultado compõe a solução final (linha 2.5).

4.5 Reconstrução

Para reincorporar as restrições de sequenciamento e capacidade retiradas na relaxação do modelo PIDLS, são executados dois procedimentos:

1. as sequências em que os produtos são processados nas máquinas são definidas;

2. um algoritmo que revê as sequências definidas no passo 1 ou transfere lotes entre máquinas diferentes de qualquer período é aplicado, assumindo-se que as restrições de capacidade são violadas em ao menos um período;

4.5.1 Definição das Sequências

Uma solução Sol definida pelo algoritmo construtivo apresentado na seção 4.4 é uma composição de P soluções Sol_p para subproblemas PDLA. Durante a construção de cada Sol_p, apesar do PDLA considerar apenas uma máquina, é feita uma escolha gulosa para determinar, em cada período, qual das máquinas paralelas vai fabricar o produto. Portanto, para restabelecer as restrições de sequenciamento basta determinar a ordem que os produtos são apresentados às máquinas.

Como pode ser observado na formulação apresentada na seção 3.3 os custos de preparação no PIDLS são independentes da sequência. Assim, a minimização do consumo de capacidade devido aos tempos de preparação dependentes da sequência é utilizada como objetivo no algoritmo apresentado a seguir:

Entrada: Uma solução Sol de um problema PIDLS infactível
Saída: Restabelecimento das restrições de sequenciamento

3.1 $Seq_t^m \leftarrow \emptyset, \quad \forall p, m;$
3.2 $Q_t^m \leftarrow 0, \quad \forall p, m;$
3.3 **para** $t = 1$ **até** T **faça**
3.4   **para** $m = 1$ **até** M **faça**
3.5   **para** $p = 1$ **até** P **faça**
3.6   **se** $f_{pt}^m = 1$ **então**
3.7 $Seq_{t,m} \leftarrow Seq_{t,m} \cup \{p\};$
3.8
3.9 **para** $t = 1$ **até** T **faça**
3.10   **para** $m = 1$ **até** M **faça**
3.11   **enquanto** Seq_{tm} *não estabelecida* **faça**
3.12 $SubS_a \leftarrow selecionaSubSequencia(Seq_{tm});$
3.13 $SubS_b \leftarrow selecionaSubSequencia(Seq_{tm});$
3.14 $Q_t^m \leftarrow Q_t^m + conectaSubSequencias(Seq_{tm}, SubS_a, SubS_b);$
3.15 **retorna** $Sol;$

Algoritmo 3: Sequenciamento

O laço das linhas 3.3 até 3.8 preenche a estrutura Seq_{tm} com subsequências compostas por apenas um produto. Nas linhas 3.12 e 3.13 são selecionadas duas subsequências $SubS_a$ e $SubS_b$ aleatoriamente, que são conectadas na linha 3.14. Uma conexão substitui as duas sequências originais por outra que liga o último produto p de uma delas ao produto inicial i da outra, de tal maneira que o custo q_{pi}^m seja mínimo. O procedimento é realizado até que uma das sequências retornadas seja vazia. A capacidade q_{pi}^m é adicionada ao consumo de capacidade devido ao sequenciamento Q_t^m.

4.5.2 Viabilização das Capacidades

Esse procedimento tem como objetivo restabelecer a viabilidade de uma solução PIDLS que não obedece os limites de capacidade. Antes de apresentar o algoritmo, algumas definições são apresentadas a seguir:

4.5.2.1 Sobrecarga de um Período

Diz-se que um período está sobrecarregado se a utilização de alguma das máquinas supera a capacidade dos recursos disponíveis.

Para cada máquina m, calcula-se o excesso de consumo de recursos ΔCap_t^m de acordo com a equação 4.5.

$$\Delta Cap_t^m = (E_t^m + Q_t^m) - w_t \tag{4.5}$$

Onde $E_t^m = \sum_{p=1}^{P} e_{pm} f_{pt}^m$ e Q_t^m (ver algoritmo 3) definem o consumo de tempo devido à produção e à preparação, respectivamente. Se $\Delta Cap_t^m < 0$, significa que a máquina m apresenta ociosidade em algum momento do período.

Para $t = 1 \ldots T$, se existe algum $\min(\Delta Cap_t^m, 0) > 0,\ m = 1 \ldots M$, então o período t está sobrecarregado e precisa ser viabilizado.

4.5.2.2 Redefinição das Sequências

O problema PIDLS não considera custos de preparação dependentes da sequência, conforme pode ser observado na formulação apresentada na seção 3.3. Porém, a redução da capacidade de utilização das máquinas decorrente dos tempos de preparação dependentes da sequência influencia o custo final do planejamento da produção ao interferir nas dimensões dos lotes.

Encontrar a sequência ótima de um conjunto de produtos associados a uma máquina consiste em resolver o *Problema do Caixeiro Viajante (PCV)* [Applegate et al., 2006], que pertence à classe de problemas NP-Difícil. Assim, métodos heurísticos são comumente empregados para resolver o PCV [Christofides, 1976; Marinakis et al., 2005b,a].

Um heurística de melhoramento *2-Opt* [Croes, 1958] é empregada para reduzir o impacto dos tempos de preparação dependentes da sequência na presença de excesso numa máquina.

Quando um produto p é inserido na posição i da máquina m a variação da capacidade ΔQ_{ip}^+ é dada por:

$$\Delta Q_{ip}^+ = (q_{p_{i-1},p}^m + q_{p,p_{i+1}}^m) - q_{p_{i-1},p_{i+1}}^m \tag{4.6}$$

onde p_{i-1} e p_{i+1} denotam os produtos que precedem e sucedem p, respectivamente, após a inserção.

Quando um produto p que está na posição i da máquina m é removido a variação da capacidade ΔQ_{ip}^- é dada por:

$$\Delta Q^-_{ip} = q^m_{p_{i-1},p_{i+1}} - (q^m_{p_{i-1},p} + q^m_{p,p_{i+1}}) \tag{4.7}$$

onde p_{i-1} e p_{i+1} denotam os produtos que precedem e sucedem p, respectivamente, antes da remoção.

Assim, quando os produtos p_1 e p_2, que estão nas posições i e j, respectivamente, são invertidos, em cada passo da heurística *2-Opt*, a variação da capacidade da máquina ΔQ é dada por:

$$\Delta Q = \Delta Q^-_{i,p_1} + \Delta Q^+_{j,p_1} + \Delta Q^-_{j,p_2} + \Delta Q^+_{i,p_2} \tag{4.8}$$

4.5.2.3 Transferência de Lotes

A existência de períodos sobrecarregados significa que os recursos disponíveis não são suficientes para realizar a produção de todos os lotes na máquina com excesso. Para tornar o período viável, necessita-se transferir alguns desses lotes (ou parte deles) para uma máquina que apresente ociosidade.

Seja um produto p cuja produção do lote foi alocada a uma máquina m que apresenta excesso de utilização. Para contribuir com a diminuição da sobrecarga do período, a parcela Δx do lote de p que pode ser transferida é dada por:

$$\Delta x = \min(\frac{\Gamma^m_t}{e_{pm}}, x_{pt}) \tag{4.9}$$

Por outro lado, a ociosidade da máquina m' que recebe o lote no período t' talvez seja insuficiente para fabricar a quantidade Δx. Se não for permitido que a transferência viole a capacidade da máquina destino, então esse valor precisa ser recalculado como a seguir:

$$\Delta x' = \min(\Delta x, \frac{-\Delta Cap^{m'}_{t'} - Q^+_{ip}}{e_{p,m'}}) \tag{4.10}$$

O lote em recalculado em $\Delta x'$ apenas quando não for possível violar a capacidade de tempo da máquina destino. Nesse caso, são consideradas apenas máquina com ociosidade, por isso a correção de sinal na variável $\Delta Cap^{m'}_{t'}$. A variável Q^+_{ip} denota a variação de capacidade de tempo quando o produto é inserido na posição i da máquina destino.

Se $\frac{\Gamma^m_t}{e_{pm}} \leq x_{pt}$, então a transferência que envolve produto p é suficiente para eliminar o excesso da máquina. A capacidade ΔW liberada após a transferência é dada por:

$$\Delta W = \begin{cases} e_{pm}x_{pt} + q_{ip}^m + q_{jp}^m - q_{ij}^m & \text{se} \quad \Delta x = x_{pt} \\ e_{pm}\Delta x & \text{se} \quad \Delta x < x_{pt} \end{cases} \tag{4.11}$$

onde i e j são os produtos que precedem e sucedem p, respectivamente.

4.5.2.4 Variação da Função Objetivo

A variação da função objetivo ΔObj quando a quantidade Δx é transferida do período t e máquina m para o período t' e máquina m' é dado por:

$$\Delta Obj = EOA(\Delta x, p, t, t') + (1 - f_{pt'}^{m'})u_{pt'}^{m'} - Tot.u_{pt}^m + \Delta x(c_{pt'} - c_{pt}) \tag{4.12}$$

onde Obj denota o custo da função objetivo antes da troca,

$$EOA(\Delta x, p, t, t') = \begin{cases} \Delta x \sum_{i=t'}^{t-1} \max(0, \Delta x - z_{pi})h_{pi} - \min(z_{pi}, \Delta x)v_{pi}, & \text{se} \quad t' < t \\ \sum_{i=t}^{t'-1} \max(0, \Delta x - s_{pi})v_{pi} - \min(\Delta x, s_{pi})h_{pi}, & \text{se} \quad t' > t \\ 0 & \text{se} \quad t' = t \end{cases} \tag{4.13}$$

e

$$Tot = \begin{cases} 1 & \text{se} \quad \Delta x = x_{pt} \\ 0 & \text{se} \quad \Delta x < x_{pt} \end{cases} \tag{4.14}$$

A equação 4.13 calcula a variação precisa dos custos para cada par produto/período. Na primeira linha da equação, os lotes antecipados visam suprir, preferencialmente, a produção atrasada ($\min(z_{pi}, \Delta x)$) e em seguida, caso haja sobra, compõem o estoque ($\max(0, \Delta x - z_{pi})$). De maneira equivalente, os lotes atrasados são compensados pelo estoque disponível ($\min(\Delta x, s_{pi})$) e apenas a quantidade restante é atrasada ($\max(0, \Delta x - s_{pi})$). Isso se deve aos custos de estoque h_{pt} serem inferiores aos custos de atraso v_{pt}.

4.5.2.5 Escolha da Máquina

Supondo que há várias máquinas ociosas distribuídas ao longo do horizonte de planejamento, necessita-se de uma estratégia de escolha para definir a melhor máquina, dentre as diversas opções disponíveis, para receber o lote transferido. Considere a equação 4.15:

$$G_{t'}^{m'} = \frac{\Delta Obj}{\Delta W} + \alpha \frac{Pen_{t',m'}}{\Delta W} \tag{4.15}$$

Onde ΔW denota a variação do excesso na máquina de origem (equação 4.11) após a transferência e $Pen_{t',m'}$ penaliza a escolha da máquina de destino em função da perda de ociosidade (equação 4.17).

Seja MT_{pt}^{m} o conjunto dos pares (período - t', máquina - m') que podem receber lotes do produto p, produzidos no período t e oriundos da máquina m. Há dois critérios que permitem o par (m', t') pertencer a MT_{pt}^{m}:

- se $f_{pt}^{m''} = 1$, para uma máquina $m'' \neq m'$, do período t, então a máquiona m' deve receber, além do lote transferido, o lote já produzido no período t. Isso se deve às restrições 3.18 ($\sum_{m=1}^{M} f_{pt}^{m} \leq 1, \ \forall p, t$);

- se $t = t'$, então o lote só pode ser transferido em sua totalidade. O motivo é o mesmo do item anterior.

Assim, o par $(m', t') \in MT_{pt}^{m}$ dever ser tal que:

$$G_{t'}^{m'} = \min_{(m'',t'') \in MT_{pt}^{m}} G_{t''}^{m''} \tag{4.16}$$

Se a máquina escolhida m' não fabrica o produto p ($f_{pt'}^{m'} = 0$), então este deve ser inserido entre os produtos i e j, tal que $\Delta Q = q_{ip}^{m'} + q_{pj}^{m'} - q_{ij}^{m'}$ seja minimizado.

4.5.2.6 Penalidade

A variação de capacidade da máquina que recebe o lote transferido precisa ser levada em conta no processo de escolha para o melhor destino. O critério $\frac{\Delta Obj}{\Delta W}$ tenta minimizar a razão entre a variação de custo geral e a redução do excesso, mas não traz nenhuma informação sobre a capacidade que fica indisponível na máquina destino.

A fração $Pen_{t',m'}$ calcula o percentual do quanto se perde em ociosidade após a máquina m' do par (t', m') receber um lote do produto p:

$$Pen_{t'}^{m'} = \Delta xe_{pm'} + \Delta Seq \tag{4.17}$$

Onde:

$$\Delta Seq = \begin{cases} q_{ip}^{m'} + q_{pj}^{m'} - q_{ij}^{m'} & \text{se} \quad f_{pt'}^{m'} = 0 \\ 0 & \text{se} \quad f_{pt'}^{m'} = 1 \end{cases} \tag{4.18}$$

denota o consumo de tempo devido às preparações, sendo $f^{m'}_{pt'}$ o estado de preparação de m' antes da transferência.

4.5.2.7 Procedimento de Viabilização

A ideia deste procedimento é investigar a sobrecarga dos períodos em busca de máquinas com excesso de produção. Quando uma máquina viola as restrições de capacidade a configuração dos produtos, seja referente à ordenação ou ao tamanho dos lotes, precisa ser revista.

Três movimentos compõem o processo de viabilização:

1. **Redefinição de Sequências:** a sequência de uma máquina é redefinida iterativamente até que o excesso seja removido ou não haja mais redução. Esse procedimento é realizado antes que haja transferência de lotes nos passos 2 e 3;

2. **Transferência Regressiva:** este movimento transfere lotes dos períodos $t = T, \ldots, 2$, examinados nesse ordem, para períodos t' mais recentes. Esse tipo de transferência não viabiliza máquinas com excesso no primeiro período.

3. **Transferência Progressiva:** este procedimento procura excessos de modo inverso ao anterior, ou seja, transfere lotes dos períodos $t = 1, \ldots, T-1$, examinados nesse ordem, para períodos t' posteriores ou para o mesmo período.

Os algoritmos para esses movimentos são apresentados a seguir.

4.5.2.8 Algoritmo para Redefinição de Sequências

Entrada: Máquina m e Período t
Saída: Redefinição da sequência dos produtos $p \in Seq_{tm}$
4.1 **para cada** $p \in Sec_{tm}$ **faça**
4.2   **para cada** $i \in Sec_{tm},\ i \neq p$ **faça**
4.3 $Sec_{tm'} \leftarrow troca(Sec_{tm}, p, i)$;
4.4   **se** $\Delta Cap^{m'}_t < \Delta Cap^m_t$ **então**
4.5 $Seq_{tm} \leftarrow Sec'_{tm}$;
4.6

Algoritmo 4: Heurística 2-Opt

Nesse procedimento, são escolhidos dois produtos p e i (linhas 4.1 e 4.2) da sequência Sec_{tm} de alguma máquina para efetuar a troca. O procedimento *troca* (linha 4.3) inverte a posição dos produtos p e i e gera a sequência $Seq_{tm'}$, que se refere a uma máquina coringa m'. Se a capacidade de tempo da sequência gerada é inferior a capacidade da sequência inicial (linha 4.4), então a Seq_{tm} é salva em $Seq_{tm'}$(4.5).

4.5.2.9 Algoritmo para Transferência Regressiva

Esse algoritmo procura satisfazer as demandas através dos estoques, uma vez que as transferências são realizadas para períodos anteriores ao atual. Caso a solução viável não seja encontrada através desse procedimento, a investigação progressiva é realizada.

Entrada: Uma solução Sol para o problema PIDLS
Saída: Uma solução Sol' modificada
5.1 **para** $t = T$ **até** 2 **faça**
5.2   **para cada** $m \in Fila_t$ **faça**
5.3 $efetuaRedefinicao(t, m)$;
5.4   **enquanto** $\Gamma_t^m > 0$ **faça**
5.5 $melhorCusto \leftarrow \infty$;
5.6   **para** $p \in Seq_{tm}$ **faça**
5.7   **para** $tDes = t - 1$ **até** 1 **faça**
5.8   **para** $mDes = 1$ **até** M **faça**
5.9 $custo \leftarrow custoTransferir(\Delta x, p, t, m, tDes, mDes)$;
5.10   **se** $custo < melhorCusto$ **então**
5.11   Armazene a tupla mínima $(\Delta x, p, t, m, tDes, mDes)$;
5.12
5.13 $efetuaTransferenciaLote(\Delta x, p, t, m, tDes, mDes)$;

Algoritmo 5: Transferência Regressiva

No algoritmo 5, os períodos são investigados deste o último até o anterior ao primeiro para verificar se há máquinas com excesso (laço 5.1). As máquinas são mantidas numa fila de prioridades que considera como chave os excessos em ordem decrescente (linha 6.2). Os elementos são retirados um a um da fila de prioridades na linha 6.2. Antes de verificar se há excesso na máquina, o algoritmo de redefinição de sequências (algoritmo 4) é executado. Se a máquina possui excesso (linha 5.4), então a tupla que minimiza a equação 4.16 é escolhida para efetuar a transferência. Para determinar a tupla mínima, é realizada uma busca que considera todos os produtos fabricados na máquina (linha 5.6), os períodos compreendidos entre o anterior ao corrente até o primeiro (linha 5.7) e todas as possibilidades de máquinas (linha 5.8). Por fim, a tupla com menor custo, calculado conforme a equação 4.15, é transferida na linha 5.13. Vale salientar que as transferências podem iniciar a configuração ou alterar o a alocação do produto na máquina destino, dessa forma as restrições impostas ao tamanho do lote estabelecidas na seção 4.5.2.5 devem ser respeitadas. As transferências durante o processo de viabilização podem causar ou aumentar o excesso na máquina destino, portanto não é necessário recalcular o tamanho do lote como sugere a equação 4.10. Ao fim de cada iteração do laço da linha 5.1, todas as máquinas do período não possuem excesso. No entanto, o procedimento pode terminar com máquinas inviáveis no primeiro período. Caso esse algoritmo não retorne um plano de produção viável, o procedimento da seção 4.5.2.10 é executado.

4.5.2.10 Algoritmo para Transferência Progressiva

A ideia principal do algoritmo de transferência progressiva consiste em adiar a produção para períodos subsequentes ao atual. Esse procedimento garante que a solução encontrada seja viável, pois na pior das hipóteses o excesso de produção é adiado para o último período e não será produzido. O algoritmo é semelhante ao anterior, diferindo na ordem em que as máquinas são analisadas. A possibilidade de haver transferências entre máquinas do período que está sendo investigado.

Entrada: Uma solução Sol para o problema PIDLS

Saída: Uma solução Sol' modificada

6.1 **para** $t = 1$ **até** $T - 1$ **faça**

6.2 **para cada** $m \in Fila_t$ **faça**

6.3 $efetuaRedefinicao(t, m)$;

6.4 **enquanto** $\Gamma_t^m > 0$ **faça**

6.5 $melhorCusto \leftarrow \infty$;

6.6 **para** $p \in Seq_{tm}$ **faça**

6.7 **para** $tDes = t$ **até** T **faça**

6.8 **para** $mDes = 1$ **até** M **faça**

6.9 $custo \leftarrow custoTransferir(\Delta x, p, t, m, tDes, mDes)$;

6.10 **se** $custo < melhorCusto$ **então**

6.11 Armazene a tupla mínima $(\Delta x, p, t, m, tDes, mDes)$;

6.12

6.13 $efetuaTransferenciaLote(\Delta x, p, t, m, tDes, mDes)$;

Algoritmo 6: Transferência Progressiva

As diferença entre o algoritmo 6 e o algoritmo apresentado na seção 4.5.2.9 (algoritmo 5) ocorrem nas linhas 6.1 e 6.7, nas quais os períodos são investigados de maneira progressiva, ou seja, partindo do primeiro até o último. Além disso, como pode ser observado na linha 6.7, o período cujas máquinas estão sendo investigadas também é incluído na busca da melhor tupla. No caso de transferências no mesmo período, são permitidos apenas lotes que englobam toda a produção do produto escolhido (de acordo com as premissas estabelecidas na seção 4.5.2.5). Deve-se tomar um cuidado adicional quando se permite transferência no mesmo período para evitar a ciclagem do algoritmo, como sugere Sabbag Neto [1993]. No final do procedimento, apenas máquinas do último período podem apresentar excesso de produção. Nessa situação, os produtos com produção excedentes têm parcelas de suas produções canceladas.

4.6 Melhoria

Os procedimentos de busca local são realizados sobre um vizinhança definida previamente. Dessa maneira, sugere-se a aplicação de duas vizinhanças que seguem as ideias de transferência de lotes descritas no procedimento de viabilização (seção 4.5.2).

O algoritmo 7 calcula a quantidade a ser transferida que produz o melhor custo (maior decremento ou menor incremento na função objetivo). Tal algoritmo é utilizado pelos movimentos descritos nas seções 4.6.1 e 4.6.2 para determinar o tamanho do lote que irá compor a tupla de transferência. Os procedimentos de melhoria não produzem inviabilidades nas máquinas que recebem lotes, portanto o tamanho do lote transferido deve sempre ser recalculado de acordo com a equação 4.10.

Entrada: Uma tupla $(p, t, m, tDes, mDes)$ para transferência
Saída: Tamanho Δx do lote

7.1 $\Delta x_1 \leftarrow x_{pt}$;
7.2 $\Delta x_1' \leftarrow recalculaLote(\Delta x_1, p, tDes, mDes)$;
7.3 $custo_1 \leftarrow custoTransferir(\Delta x_1', p, t, m, tDes, mDes)$;
7.4 $\Delta x \leftarrow \Delta x_1'$;
7.5 $\Delta x_2 \leftarrow x_{pt} - d_{pt}$;
7.6 **se** $\Delta x_2 > 0$**então**
7.7 $\Delta x_2' \leftarrow recalculaLote(\Delta x_2, p, tDes, mDes)$;
7.8 $custo_2 \leftarrow custoTransferir(\Delta x_2', p, t, m, tDes, mDes)$;
7.9   **se** $custo_2 < custo_1$ **então**
7.10 $custo_1 \leftarrow custo_2$;
7.11 $\Delta x \leftarrow \Delta x_2'$;
7.12 **se** $t = T$ **então**
7.13   **se** $\Gamma_t^m > 0$ **então**
7.14 $\Delta x_3 \leftarrow \min\{x_{pt}, \Gamma_t^m / e_{pt}\}$;
7.15 $\Delta x_3' \leftarrow recalculaLote(\Delta x_3, p, tDes, mDes)$;
7.16 $custo_{atrasar} \leftarrow \Delta x_3'(v_{pt} - x_{pt}) - u_{pt}^m * f_{pt}^m$;
7.17 $custo_3 \leftarrow custo_{atrasar} - custoTransferir(\Delta x_3', p, t, m, tDes, mDes)$;
7.18   **se** $custo_3 < custo_1$ **então**
7.19 $\Delta x \leftarrow \Delta x_3'$;

Algoritmo 7: Determina Tamanho do Lote para o Procedimento de Melhoria

O algoritmo 7 define três estratégias para determinar, para um produto candidato a transferência, qual o melhor quantidade poderá ser transferida. Na primeira ideia, o lote total produzido no período (linha 7.1) é escolhido como sendo a melhor dimensão para a transferência. O procedimento *recalculaLote* realiza os ajustes ao lote, de acordo com a equação 4.10, para impedir que a máquina destino fique inviável (linhas 7.2, 7.7 e 7.15). Na segunda ideia, tenta-se preservar a produção que supre a demanda, enquanto a produção para satisfazer estoques ou atrasos é transferida (linha 7.5). Por último, verifica-se o custo de transferir os lotes que causam excesso 7.14, situação possível apenas em transferências cuja origem é o último período (linha 7.12). Os custos são calculados para cada um dos lotes determinados (linhas 7.3, 7.8 e 7.17) e o lote que gera o menor custo é retornado.

4.6.1 Transferência Regressiva - N_1

Este movimento verifica todos os produtos com produção positiva ($x_{pt} > 0$)) do período t considerado e transfere parte do seu lote para ser fabricado em algum período t' anterior. O par máquina/período que irá receber o produto é escolhido, dentre todos os possíveis, como sendo aquele que produz o menor custo e a quantidade a ser transferida é calculada de acordo com o algoritmo 7.

A busca local utilizada neste procedimento é baseada na heurística de *melhor descida*. Na heurística *melhor descida*, são avaliados todos os movimentos possíveis da vizinhança e aquele que produz o melhor ganho é escolhido. Sendo assim, a tupla escolhida para transferência é a melhor de todas as tuplas possíveis. Máquinas que não possuem ociosidade não são candidatas a receber qualquer quantidade de lote, portanto essas máquinas são excluídas da busca. As transferências deste movimento sempre podem ser desfeitas por um movimento descrito na seção 4.6.2.

4.6.2 Transferência Progressiva - N_2

Este movimento opera de maneira inversa ao anterior. Ou seja, em vez de antecipar a produção, o procedimento atrasa a produção do produto sorteado, desde que seu lote tenha dimensão positiva ($x_{pt} > 0$)), para algum período posterior. Os tamanhos dos lotes transferidos são também calculados pelo algoritmo 7. Esse tipo de movimento visa diminuir os custos devido aos estoques. As transferências deste movimento sempre podem ser desfeitas por um movimento descrito na seção 4.6.1.

4.6.3 Busca Tabu

A metaheurística Busca Tabu (BT) [Glover et al., 1997] é um procedimento adaptativo que guia o processo de exploração de um espaço de busca definido por estruturas de vizinhança. A adoção de uma memória flexível, conhecida como *Lista Tabu*, permite armazenar conhecimento sobre o espaço de busca e assim evitar que se retorne a ótimos locais recém visitados.

Uma heurística baseada em BT tem os seguintes componentes:

- Um procedimento capaz de gerar soluções viáveis para o problema considerado;
- Uma Lista Tabu para armazenar as soluções proibidas (ditas *tabu*);
- Um contador que define o prazo que uma solução permanece na lista tabu;
- Condições de aspiração que permitem a visitação de soluções tabu.

O algoritmo 8 apresenta um arcabouço para uma metaheurística Busca Tabu.

Entrada: Solução inicial S_0
Saída: Um ótimo local S

$S \leftarrow S_0$;
$S^* \leftarrow S_0$;
$ListaTabu \leftarrow \emptyset$;
enquanto *critério de parada não satisfeito* **faça**
    $S' \leftarrow obterMelhorVizinho(s, N, ListaTabu)$;
    se $custo(S') < custo(S^*)$ **então**
        $S^* \leftarrow S'$;
    senão
        se $custo(S') > custo(S)$ **então**
            $ListaTabu \leftarrow ListaTabu \cup S$;
    $S \leftarrow S'$;

Algoritmo 8: Metaheurística Busca Tabu

4.6.3.1 Considerações sobre a metaheurística

Para melhorar a eficiência do processo de exploração por melhores soluções, é necessário que não apenas informações locais, como o valor da função objetivo, sejam avaliadas, mas também alguma informações sobre o processo de exploração. O uso de memória é uma característica essencial da Busca Tabu (BT). Ou seja, enquanto a maioria dos métodos de exploração armazenam apenas o melhor valor de todas as soluções visitadas até então, a Busca Tabu guarda informações sobre o caminho que levou até a última solução visitada. Essas informações são usadas para guiar o movimento de uma solução S_1 para a próxima solução S_2 pertencente à vizinhança $N(S_1)$. O uso da memória proíbe a visitação de um subconjunto de soluções $TABU_{iter} \subset N(S_1)$. O conjunto $TABU_{iter}$ é variável ao longo das iterações em função do caminho percorrido para se chegar a S_1. O procedimento, em contraste ao métodos exploratórios de descida, também aceita soluções não aprimorantes nas transições entre S_1 e S_2 ($f(S_1) < f(S2)$ em problemas de minimização). A aceitação de soluções não aprimorantes visa fugir de mínimos locais.

4.6.4 Algoritmo de Melhoria

O procedimento de busca local é baseado na metaheurística Busca Tabu (seção 4.6.3). Antes do algoritmo ser apresentado, são feitas algumas considerações a seguir:

- os movimentos que definem as vizinhanças N_1 e N_2 são aplicados de maneira alternada durante o processo de busca;
- as tuplas (p, m, t, m', t') são armazenadas na Lista Tabu para evitar que transferências progressivas desfaçam as transferências realizadas por movimentos regressivos, ou vice-versa;
- as tuplas permanecem na Lista Tabu durante k iterações;
- O critério de aspiração permite que movimentos tabus sejam executados quando não há mais máquinas destino disponíveis ou se o custo da solução tabu apresentar o melhor valor de todas as soluções visitadas até então.
- A vizinhança é alternada sempre que um mínimo local é alcançado.
- se a solução não melhora durante as últimas k' iterações, é gerada uma pertubação na solução atual. Essa pertubação altera a configurações de um produto p escolhido aleatoriamente em cada uma das máquina para uma máquina m também aleatória. Note que a pertubação pode gerar planos inviáveis, então o procedimento de viabilização deve ser executado em seguida.

O algoritmo 9 sumariza o procedimento de melhoria. Uma solução inicial viável obtida por um procedimento qualquer (nesta dissertação, adota-se a solução inicial gerada conforme descrito na seção 4.4) é salva como solução atual (linha 9.1) e a melhor solução (linha 9.1) antes do procedimento iterativo começar. A vizinhança inicial é definida como sendo a N_1 (linha 9.4). A lista tabu é inicializada na linha 9.14. Durante as iterações definidas pela variável *NumIteracoes* (linha 9.5), escolhe-se a tupla para transferência que gera o menor custo pelo procedimento *obtenhaMelhorTuplaTransferencia* (linha 9.6). Esse procedimento leva em consideração a busca local N_i adotada na iteração corrente e a lista tabu. Se a busca local melhorou a melhor solução encontrada até então, essa solução é salva na linha 9.9. Se a solução obtida piorou a solução

Entrada: Solução inicial S_0
Saída: Um ótimo local S^*

9.1 $S \leftarrow S_0$;
9.2 $S^* \leftarrow S_0$;
9.3 $ListaTabu \leftarrow \emptyset$;
9.4 $i \leftarrow 1$;
9.5 **para** $iter = 1 \leftarrow NumIteracoes$ **faça**
9.6 $\quad (t', m') \leftarrow obtenhaMelhorTuplaTransferencia(p, N_i, ListaTabu)$;
9.7 $\quad S' \leftarrow efetuarTransferenciaLote(S, t1, m')$;
9.8 $\quad$ **se** $custo(S') < custo(S^*)$ **então**
9.9 $\quad\quad S^* \leftarrow S'$;
9.10 $\quad$ **senão**
9.11 $\quad\quad$ **se** $custo(S') > custo(S)$ **então**
9.12 $\quad\quad\quad$ **se** $i = 0$ **então**
9.13 $\quad\quad\quad\quad i \leftarrow 1$;
9.14 $\quad\quad\quad$ **senão**
9.15 $\quad\quad\quad\quad i \leftarrow 0$;
9.16 $\quad atualizarListaTabu(ListaTabu, p, t, m, t', m')$;
9.17 $\quad S \leftarrow S'$;
9.18 $\quad$ **se** *não melhorou nas últimas* k' *iterações* **então**
9.19 $\quad\quad S \leftarrow pertubar(S)$;
9.20 $\quad\quad S \leftarrow viabilizar(S)$;

Algoritmo 9: Heurística de Melhoria

corrente, então a vizinhança é trocada (linha 9.11 até 9.15). O movimento é salvo como tabu na linha 9.16 e a solução encontrada é adotada como solução atual na linha 9.17.

Os movimentos de busca local são incapazes de alterar as configurações das máquinas. Dessa forma, um mínimo local é atingido rapidamente. Para fugir de mínimos locais, a solução atual, no caso de não ter sido melhorada nas últimas k' iterações (linha 9.18), é pertubada e uma nova solução é gerada pelo procedimento *pertubar* (linha 9.19). Esse movimento não faz nada mais que selecionar aleatoriamente um produto p, deste que $x_{pt} > 0$, e alterar sua configuração para uma máquina m também aleatória. Como pertubações podem gerar inviabilidades, o procedimento de viabilização é executado na linha 9.20.

A metaheurística Busca Tabu foi escolhida como base para o procedimento de melhoria por ter apresentado bons resultados quando os movimentos da vizinhança N_1 (N_2) foram proibidos de serem desfeitos pelos movimentos vizinhança N_2 (N_1) durante algumas iterações. A alternância entre as vizinhanças foi baseada na metaheurística *Variable Neighborhood Descent - (VND)* [Hansen & Mladenovi, 2001]. Testes computacionais mostraram que a alternância de vizinhança gerou soluções melhores que a escolha aleatória entre as vizinhanças disponíveis ou a não distinção entre os movimentos progressivo e regressivo.

Capítulo 5

Limites Duais para o PIDLS

Este capítulo apresenta um procedimento para obtenção de limites duais para o *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas - PDLCAMP* (seção 5.4). Mostra-se, na seção 5.1, que tais limites também se aplicam ao *Problema Integrado de Dimensionamento de Lotes e Sequenciamento* quando as aproximações dos tempos de preparação do PDLCAMP são escolhidas como sendo o tempo de preparação independente da sequência do PIDLS.

5.1 Considerações Iniciais

Quando os custos de preparação dependentes da sequência são removidos do problema original, a questão de onde produzir se limita em determinar apenas as máquinas que vão fabricar os produtos, sem considerar a ordenação de produção. Essa relaxação do PIDLS, denominada de *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas - PDLCAMP*, é capaz de determinar esse plano. Como os tempos de preparação são inexistentes, a produção dentro dos períodos se torna mais flexível, uma vez que haverá disponibilidade de tempo maior. Portanto, o PDLCAMP é realmente uma relaxação para o problema original. Dessa maneira, a solução ótima ou relaxações para o PDLCAMP são também relaxações para o PIDLS. A figura 5.1 exemplifica uma solução para o problema PDLCAMP. Como se pode observar, os tempos de preparação independem da sequência.

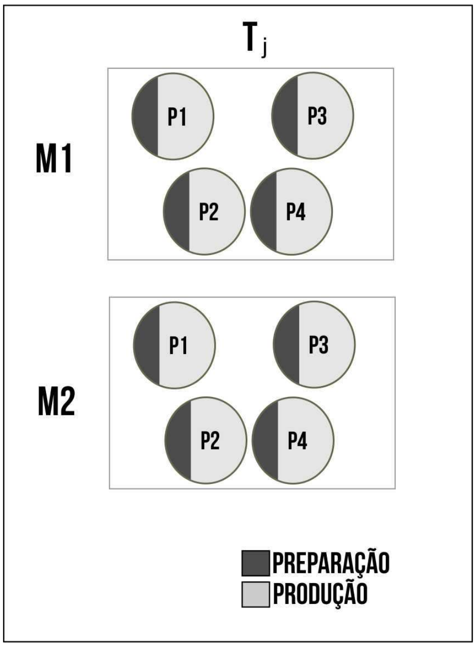

Figura 5.1. Problema PDLCAMP

5.2 Limites Primais

Encontrar soluções viáveis para problemas de otimização combinatória pode ser uma tarefa simples ou complexa, dependendo da natureza do problema considerado. Soluções viáveis representam limitantes superiores para problemas de minimização (inferiores para problemas de maximização). No caso dos problemas de dimensionamento de lotes e sequenciamento, determinar a existência de limites duais é um problema *NP-Completo* quando há restrições de capacidade de produção e atrasos não são permitidos. Muitos dos trabalhos encontrados na literatura [Toledo & Armentano, 2006; Trigeiro et al., 1989; Sabbag Neto, 1993; França et al., 1997] lidam com essa situação e apresentam procedimentos de solução que dão atenção especial à etapa de viabilização. Para o PIDLS, encontrar uma solução viável não é difícil, mas a questão fundamental é como encontrar boas soluções. O procedimento apresentado no capítulo 4 tenta obter bons limites primais para o PIDLS e, neste capítulo, o objetivo é gerar bons limites duais (limites inferiores).

5.3 Limites Duais

Obter limites inferiores para problemas de minimização (limites superiores para problemas de maximização) é uma atividade desafiadora. Alguns métodos para encontrar esses limites relaxam o problema original, ou seja, convertem o problema cuja solução está sendo investigada em outro mais simples, cuja obtenção da solução exige menos esforço computacional. Entre métodos existentes, os mais tradicionais são baseados em Relaxação Linear, Relaxação combinatória e Relaxação Lagrangeana.

5.3.1 Relaxação Linear

Em tal relaxação as variáveis inteiras da formulação problema são convertidas em variáveis reais. Nesse tipo de método, procura-se desenvolver formulações tidas como "boas", ou seja, que forneçam limites duais fortes.

5.3.2 Relaxação Combinatória

Quando a relaxação é um problema de otimização combinatória dizemos que ela é uma *Relaxação Combinatória*. No caso dos problemas de dimensionamento, a remoção das restrições de capacidade induz a problemas de dimensionamento de lotes não capacitados que são uma relaxação do problema que está sendo resolvido.

5.3.3 Relaxação Lagrangeana

Muitos problemas de otimização combinatória considerados difíceis podem ser vistos como problemas fáceis, porém complicados por um subconjunto de restrições. Considere o problema de programação inteira a seguir:

$$PI \quad z = \min \quad cx \tag{5.1}$$
$$Ax \geq b \tag{5.2}$$
$$Dx \geq d \tag{5.3}$$
$$x \in Z_+^n \tag{5.4}$$

Se as restrições 5.2 podem ser identificadas como "fáceis", no sentido de que um problema inteiro com apenas essas restrições é tido como "simples"(pseudo-polinomial ou mesmo polinomial), então podemos eliminar as restrições complicadoras 5.3 e gerar uma relaxação que é mais fácil de resolver que o problema original. Em vez de simplesmente remover restrições, o que poderia gerar limites inferiores fracos, a Relaxação Lagrangeana as incorpora à função objetivo, usando multiplicadores de Lagrange como penalidades, conforme abaixo:

$$PI(u) \quad z(u) = \min \quad cx + u(d - Dx) \tag{5.5}$$

$$Ax \geq b \tag{5.6}$$

$$x \in Z^n_+ \tag{5.7}$$

para qualquer valor de $u = \{u_1, \ldots, u_m\} \geq 0$.

Proposição 5.3.1 *O problema* PI(u) *é uma relaxação do problema PI qualquer que seja $u \geq 0$.*

Como $PI(u)$ é uma relaxação do problema PI, o valor $z(u)$ representa um limite inferior ($z(u) \leq z$) para o problema PI. Para encontrar o melhor limite inferior (ou seja, aquele com maior valor) para todos os valores de u, deve-se resolver o *Problema Dual Lagrangeano* (equação 5.8).

$$z_{DL} = \max\{z(u) : u \geq 0\} \tag{5.8}$$

A resolução da relaxação Lagrangeana $PI(u)$ pode, eventualmente, conduzir a uma solução do problema inteiro original.

Proposição 5.3.2 *Se $u \geq 0$:*

1. *$x(u)$ é solução ótima para $PI(u)$, e*

2. *$Dx(u) \geq d$, e*

3. *$(Dx(u))_i = d_i$, sempre que $u_i > 0$ (complementariedade de folga),*

então $x(u)$ é solução ótima para o problema PI.

As demonstrações para as proposições 5.3.1 e 5.3.2 podem ser encontradas em Wolsey [1999].

5.3.3.1 Resolução do Problema Dual Lagrangeano

Fisher [1981] demonstrou que o *Problema Dual Lagrangeano* pode ser visto como o problema de maximizar uma função linear por partes côncava (figura 5.2), mas não diferenciável. Na maximização de funções côncavas, o ótimo local é também ótimo

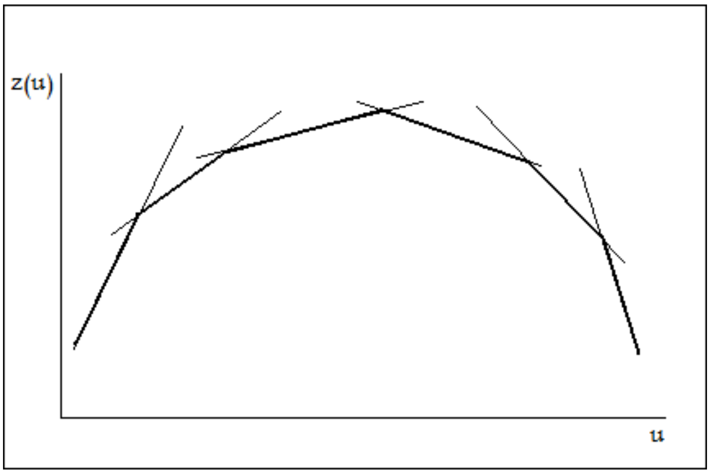

Figura 5.2. Função do problema dual Lagrangeano

global. Essa característica faz com que a Relaxação Lagrangeana seja uma estratégia atrativa para obter limites duais.

O Método dos Subgradientes foi projetado para resolver o problema de maximizar uma função linear por partes côncava. Um subgradiente é uma generalização direta de um gradiente.

Definição 5.3.1 *Um subgradiente em u de uma função convexa $f : R^m \rightarrow R^1$ é um vetor $\gamma(u) \in R^m$, tal que $f(v) \geq f(u) + \gamma(u)^T(v-u)$, qualquer que seja $v \in R^m$. Para uma função convexa continuamente diferenciável f, $\gamma(u) = \nabla f(u) = (\frac{\partial f}{\partial u_i} \ldots \frac{\partial f}{\partial u_m})$ é um gradiente de f em u.*

Os passos abaixo são seguidos para resolver o Problema Lagrangeano:

1. **Inicialização:** os parâmetros do algoritmo são inicializados;
2. **Resolver o problema relaxado:** deve-se desenvolver um algoritmo para resolver o problema $PI(u)$. Tal procedimento deve ter resolução fácil (pseudo-polinomial ou polinomial);
3. **Salvar a solução atual:** se o limite é melhorado, este valor deve ser armazenado;
4. **Condição de parada:** define-se uma condição de parada para o algoritmo (número de iterações ou subgradiente nulo, por exemplo);
5. **Atualiza o tamanho do salto:** O salto deve ser atualizado conforme a equação 5.9, onde ls é um limite superior para o problema e ν_k é um multiplicador para o salto.

$$\epsilon_k = \frac{\nu_k(ls - z(u^k))}{||d - Dx(u^k)||^2}, \quad 0 < \nu_k < 2 \tag{5.9}$$

A demonstração de convergência para o método dos subgradientes pode ser encontrada em Wolsey [1999].

5.4 Relaxação Lagrangeana para o PDLCAMP

A Relaxação Lagrangeana (LR-PDLCAMP) é obtida a partir da dualização das restrições de capacidade 3.31 do problema PDLCAMP. Essa relaxação é similar àquela proposta por Armentano & Toledo [1997] para o Problema de Dimensionamento de Lotes Capacitado em Máquinas Paralelas. Considere um conjunto de multiplicadores lagrangeanos $\lambda_{mt} \geq 0$, $m = 1 \ldots M$ e $t = 1 \ldots T$. O problema Lagrangeano resultante é dado por:

$$\begin{aligned} Q(\lambda_{mt}) = \min \sum_{p=1}^{P} \sum_{t=1}^{T} \sum_{m=1}^{M} (c_{pt} + \lambda_{mt} e_{pm}) x_{pt}^m + (u_{pt}^m + \lambda_{mt} a_{pm}) f_{pt}^m \\ + \sum_{p=1}^{P} \sum_{t=1}^{T} h_{pt} s_{pt} + v_{pt} z_{pt} - \sum_{m=1}^{M} \sum_{t=1}^{T} \lambda_{mt} w_t \end{aligned} \tag{5.10}$$

sujeito às restrições 3.30, 3.32 e 3.33

O problema $Q(\lambda_{mt})$ pode ser decomposto na soma da constante $-\sum_{m=1}^{M} \sum_{t=1}^{T} \lambda_{mt} w_t$ a um conjunto de P subproblemas $Q_p(\lambda_{mt})$ com apenas um produto e sem restrições de capacidade.

$$Q_p(\lambda_{mt}) = \min \sum_{t=1}^{T} \sum_{m=1}^{M} (c_{pt} + \lambda_{mt} e_{pm}) x_{pt}^m + (u_{pt}^m + \lambda_{mt} a_{pm}) f_{pt}^m + \sum_{t=1}^{T} h_{pt} s_{pt} + v_{pt} z_{pt} \tag{5.11}$$

$$\sum_{m=1}^{M} x_{pt}^m + s_{p(t-1)} + z_{pt} = d_{pt} + s_{pt} + z_{p(t-1)}, \quad \forall t \tag{5.12}$$

$$x_{pt}^m \leq K f_{pt}^m, \quad \forall t, m \tag{5.13}$$

$$\sum_{m=1}^{M} f_{pt}^m \leq 1, \quad \forall t \tag{5.14}$$

$$f_{pt}^m \in \{0, 1\}, x_{pt}^m \in N, \quad \forall p, m \tag{5.15}$$

$$s_{pt}, z_{pt} \geq 0, \quad \forall t \tag{5.16}$$

O modelo acima refere-se ao *Problema de Dimensionamento de Lotes com Atraso em Máquinas Paralelas – PDLAMP*, que é uma extensão do problema PDLA 3.4 que considera custos de produção e preparação dependentes da máquina.

Toledo & Shiguemoto [2005] apresentaram um algoritmo de programação dinâmica que encontra soluções ótimas para o PDLAMP. Esse algoritmo é uma variação do procedimento exato descrito na seção 4.3, ao computar os custos de produção para cada uma das máquinas, modificando a complexidade do algoritmo para $O(MT^2)$.

Outras implementações podem ser encontradas em Sung [1986], que também apresentou um algoritmo de programação dinâmica para o problema de dimensionamento de lotes que considera múltiplas máquinas com complexidade computacional da ordem $O(MT^2)$ e Armentano & Toledo [1997], que propuseram uma implementação eficiente, baseada em Evans [1985], para tal algoritmo.

Os passos abaixo detalham a Relaxação Lagrangeana implementada, tomando como base o algoritmo apresentado na seção 5.3.3.1.

1. **Inicialização**

   a) Inicialize todos os multiplicadores para $\lambda = 0$;

   b) Inicialize o contador de iterações $iter = 0$;

   c) Inicialize o tamanho do salto $\epsilon_0 = 2$;

   d) Inicialize o multiplicador $\nu_0 = 1$;

   e) Inicialize o limite inferior $LI = -\infty$;

   f) Obtenha um limite superior $LS = obterLimiteSuperior()$;

2. **Resolver o problema relaxado:** resolva cada um dos subproblemas Q_p utilizando o algoritmo exato para o PDLAMP e calcule o limite inferior atual LI_{iter};

3. **Salva a solução atual:** Se $LI_{iter} > LI$, então $LI = LI_{iter}$;

4. **Condição de parada:** Se $LI = LS$, então pare. Se $tempoExecucao > 10min$, então pare. Calcule os gradientes. Se os subgradientes forem nulos, pare.

5. **Atualiza o tamanho do salto:** $\epsilon_{iter+1} = \epsilon_{iter}/2$ se o limite inferior não é melhorado durante as últimas 30 iterações;

6. **Atualiza multiplicadores:** atualiza os multiplicadores Lagrangeanos;

7. $iter = iter + 1$. Retorne ao passo 2;

O procedimento $obterLimiteSuperior()$ obtém uma solução primal usando a heurística descrita no capítulo 4, descartando os passos que tratam de sequenciamento. Os parâmetros do procedimento foram definidos experimentalmente após alguns testes computacionais.

Capítulo 6

Resultados computacionais

Este capítulo apresenta os resultados computacionais obtidos através de alguns cenários de testes. São considerados, a título de comparação, o *Problema Integrado de Dimensionamento de Lotes e Sequenciamento - PIDLS* e o *Problema de Dimensionamento de Lotes Capacitado com Atraso em Máquinas Paralelas - PDLCAMP*, bem como relaxações para esses problemas.

Inicialmente, foram testadas as instâncias geradas em [Belisário et al., 2009], considerando-se apenas os resultados da heurística para o problema PIDLS. Em seguida, foram construídos cenários de testes a partir de instâncias geradas artificialmente como descrito na seção 6.1.

As heurístcias foram codificadas na linguagem java (versão do compilador $1.6.0\_15 - b03$) e executadas na máquina virtual *OpenJDK Runtime Environment* versão 1.6.0_18 64bit. Os problemas foram modelados e as formulações resolvidas por esses aplicativos desenvolvidos em C++ (compilados com a versão 4.3.3 do gcc) através da *API Concert Technology* e o *solver* CPLEX versão 10.2. Os testes foram realizados em um *Core 2 Quad* $2.5GHz$ com $4GB$ de memória e sistema operacional Linux Ubuntu Hardy Heron (8.04). Para todas as instâncias, foi definido um tempo máximo de processamento de 10 minutos.

Com relação ao testes realizados, foram anotados os resultados obtidos para os parâmetros a seguir:

- **C. PIDLS** Capacidade média do planejamento de produção obtido de acordo com as restrições do modelo PIDLS;

- **T. PIDLS** Tempo médio em segundos para resolver as instâncias problema PIDLS;

- **C. PDLCAMP** Capacidade média do planejamento de produção obtido de acordo com as restrições do modelo PDLCAMP;
- **T. PDLCAMP** Tempo médio em segundos para resolver as instâncias o problema PDLCAMP;
- **Min. PIDLS (%)** Menor desvio em relação ao limite inferior para o problema PIDLS;
- **Max. PIDLS (%)** Maior desvio em relação ao limite inferior para o problema PIDLS;
- **Min. PDLCAMP (%)** Menor desvio em relação ao limite inferior para o problema PDLCAMP;
- **Max. PDLCAMP (%)** Maior desvio em relação ao limite inferior para o problema PDLCAMP;

A fórmula $desvio = \frac{100 \times (limite\_superior - limite\_inferior}{limite\_superior)}$ foi utilizada para calcular o desvio. Já a capacidade média foi calculada conforme a fórmula: $\sum_{t=0}^{T}\sum_{m=0}^{M} 100 \times ocupacao_{mt}/capacidade_t$, onde $ocupacao_{mt} \leq capacidade_t$ denota a ocupação da máquina m no período t e $capacidade_t$ se refere a capacidade de tempo do período. No caso de máquinas do último período com excesso, a ocupação foi considerada como sendo a capacidade do período.

6.1 Geração de Instâncias

O algoritmo para geração de instâncias artificiais estende o procedimento descrito em Paula [2008]. Seja W a soma dos tempos de produção e preparação após a aplicação do algoritmo EDD-Estendido (algoritmo 10). Para calcular os tempo de processamento considera-se o tamanho da tarefa como sendo a demanda do período. Os custos de produção, estoque, atraso e configuração, assim como os tempos de preparação e produção, são gerados por uma distribuição uniforme com limites descritos nas tabela 6.1

Os tempos de preparação também satisfazem a desigualdade triangular, tal que $q_{ip} < q_{ik} + q_{kp}$, $\forall i \neq p \neq k \in |P|$. Uma regra válida para o cálculo dos limites de tempo do período é que quanto maior o valor de θ mais difícil de resolver é a instância, pois a capacidade do período fica mais apertada.

As instâncias foram geradas para 4 cenários distintos quanto ao número de produtos, períodos e máquinas. No primeiro cenário (tipo 1), foram considerados 6 produtos,

Dado	Min	Max
Custo de Produção	3	5
Custo de Estoque	1	8
Custo de Atraso	20	50
Custo de Preparação	5	80
Tempo de Produção	1	5
Demanda	0	20
Limite do Tempo	$\max_{p \in Lista} Lista[p]$	$2 * W/\theta$

Tabela 6.1. Limites das Instâncias

6 períodos e 2 máquinas; já no segundo (tipo 2), 8 produtos, 6 períodos e 2 máquinas; no terceiro (tipo 3), 12 produtos, 12 períodos, e 4 máquinas; finalmente, no último (tipo 4), 16 produtos, 12 períodos e 4 máquinas. O parâmetro θ recebeu os valores 1, 3 e 5. Assim, foram geradas instâncias com disponibilidade de tempo ALTA, MÉDIA e BAIXA.

Para avaliar o impacto dos tempos de preparação dependentes da sequência, foram considerados os intervalos $(30, 30)$, $(30, 40)$ e $(30, 50)$ para gerar os valores na distribuição uniforme. Assim, foram geradas instância com dispersão ALTA, MÉDIA E BAIXA dos valores dos tempos de preparação dependentes da sequência. A variação desses valores é importante, uma vez que o tempo de preparação no PDLCAMP é considerado como o menor tempo de preparação de um produto levando em conta os tempos de todos os produtos que podem antecedê-lo.

A tabela 6.2 sumariza os cenários de testes considerados neste trabalho.

Cenário	Nº Prod.	Nº Per.	Nº Maq.	T. Cap.	T. Prep
A1	6	6	2	BAIXA	BAIXA
B1					MÉDIA
C1					ALTA
D1				MÉDIA	BAIXA
E1					MÉDIA
F1					ALTA
G1				ALTA	BAIXA
H1					MÉDIA
I1					ALTA
A2	8	6	2	BAIXA	BAIXA
B2					MÉDIA
C2					ALTA
D2				MÉDIA	BAIXA
E2					MÉDIA
F2					ALTA
G2				ALTA	BAIXA
H2					MÉDIA
I2					ALTA
A3	12	12	4	BAIXA	BAIXA
B3					MÉDIA
C3					ALTA
D3				MÉDIA	BAIXA
E3					MÉDIA
F3					ALTA
G3				ALTA	BAIXA
H3					MÉDIA
I3					ALTA
A4	16	12	4	BAIXA	BAIXA
B4					MÉDIA
C4					ALTA
D4				MÉDIA	BAIXA
E4					MÉDIA
F4					ALTA
G4				ALTA	BAIXA
H4					MÉDIA
I4					ALTA

Tabela 6.2. Análise dos Resultados

Entrada: Uma lista ordenada de tarefas $List$

Saída: Uma solução viável

para $t = 1$ **até** T **faça**
- **para** $p \in List$ **faça**
  - $MelhorMaquina \leftarrow 0$;
  - $MelhorObjetivo \leftarrow \infty$;
  - **para cada** $m \in |M|$ **faça**
    - $Objetivo \leftarrow$ função objetivo parcial assumindo que p é processado no final da lista de processamento da máquina m ;
    - **se** $Objetivo < MelhorObjetivo$ **então**
      - $MelhorMaquina \leftarrow m$;
      - $MelhorObjetivo \leftarrow Objetivo$;
  - Processe $List[p]$ no final da lista de processamento da $MelhorMaquina$;

Algoritmo 10: EDD-Estendido

6.2 Cenários de Teste

Nas seções que se seguem, os resultados computacionais obtidos nesta dissertação são apresentados e avaliados. Na seção 6.2.1, as instâncias geradas artificialmente neste trabalho são submetidas aos procedimentos desenvolvidos. Para esses resultados, são apresentados os valores para o PIDLS e PDLCAMP comparados ao limite dual obtido através da relaxação Lagrangeana (seção 5). Na seção 6.2.2, são mostrados os resultados dos experimentos realizados por Belisário et al. [2009] comparados com os resultados obtidos pela heurística desenvolvida na seção 4.

6.2.1 Avaliação das Instâncias Geradas Artificialmente

Para cada combinação de produtos, períodos, máquinas, variação de capacidade e dispersão de tempo de preparação, foram geradas 10 instâncias conforme descrito na seção 6.1.

A tabela 6.3 exibe os resultados computacionais para o problema PIDLS. Como pode ser observado, mesmo sendo permitido o tempo computacional máximo de 10 minutos, nenhuma instância demorou mais que 2 minutos para obter a melhor solução.

Comparando os resultados obtidos pela heurística para resolver o PIDLS com os resultados da relaxação Lagrangeana, percebe-se que alguns dos desvios gerados são superiores a 20%. Para esses valores, foi observado que a capacidade de tempo disponível para todo o horizonte de planejamento não foi suficiente para produzir todas as demandas. Nessas situações, os desvios entre as soluções das heurísticas e os valores obtidos pela relaxação Lagrangeana elevaram-se por conta dos altos custos de atrasos que penalizam a função objetivo. Como pode ser observado, essas instâncias apresentaram ocupação média superior a 90%.

Por outro lado, mesmo com a ocupação média das máquinas nunca ter atingido valores inferiores a 70%, alguns desvios apresentaram valores aceitáveis. Mesmo para as instâncias do tipo 4, que possuem dimensões que tornam inviável a obtenção de soluções exatas através de *solvers* comerciais, 50% dos desvios não alcançaram valores maiores que 10%. Vale ressaltar que os limites inferiores gerados pela relaxação Lagrangeana são limites duais para uma relaxação combinatória do problema PIDLS. Portanto, só em casos em que os tempos de preparação possuem os mesmos valores pode ocorrer a igualdade entre os limites duais e primais.

A variação das capacidades de tempo interferiu na qualidade dos desvios. Observa-se que as soluções que apresentaram os piores desvios também apresentam altas taxas de ocupação (categorias BAIXA ou MÉDIA). Capacidades de tempo mais apertadas resultam em atrasos de produção ao longo do planejamento, fazendo com que o custo final seja penalizado.

Já os tempos de preparação dependentes da sequência mais dispersos (categorias MÉDIA e ALTA) também produzem desvios piores. Quando são considerados tempos de preparação no problema PDLCAMP muito baixos, a capacidade de tempo disponível pode ser melhor aproveitada. Como são considerados sempre os menores tempos de preparação do PIDLS no PDLCAMP, pode acontecer do PDLCAMP ser resolvido com tempos de preparação independente da sequência muito distantes dos tempos dependentes da sequência considerados no PIDLS. Sendo assim, as soluções exatas do problema PDLCAMP tendem a ser mais distantes que as soluções exatas para o PIDLS. Dessa maneira, os desvios também tendem a ser maiores.

Na tabela 6.4, são exibidos os resultados computacionais da heurística para o PDLCAMP. Para resolver o problema PDLCAMP, a heurística proposta no capítulo 4 foi adaptada para desconsiderar os tempos de preparação dependentes da sequência. Dessa maneira, foi considerado como tempo de preparação no PDLCAMP o tempo mínimo no problema PIDLS para preparar qualquer produto. Assim, a solução exata obtida pelo PDLCAMP é uma relaxação combinatória para o PIDLS. Por conseguinte, um limite dual para o PDLCAMP é também um limite inferior para o PIDLS. A tabela 6.3 mostra os resultados computacionais do PIDLS em relação limites duais gerados pela relaxação Lagrangeana.

Cenário	C. PIDLS (%)	T. PIDLS	Min. PIDLS	Max. PIDLS
A1	91,4	0	22,52	26,8
B1	92,1	0	18,22	19,45
C1	89,9	0	15,89	25,13
D1	83,9	0	5,22	18,78
E1	87,6	0	5,16	22,57
F1	82	0	4,81	19,3
G1	81,47	0	1,62	14,3
H1	80,06	0	1,59	8,66
I1	79,99	0	1,86	9,25
A2	98,3	0	14,88	21,61
B2	96,5	0	9,12	25,21
C2	93,1	0	12,72	29,33
D2	84,5	0	4,3	11,47
E2	83,3	0	8,47	9,55
F2	83,2	0	6,22	6,91
G2	78,26	0	1,21	4,82
H2	83,41	0	1,66	4,68
I2	81,03	0	1,99	4,42
A3	92,26	1	6,38	10,4
B3	90,82	12	7,6	10,75
C3	90,4	3	6,54	12,01
D3	81,11	4	3,37	5,99
E3	80	4	3,51	6,15
F3	83,4	7	3,3	7,12
G3	75,64	6	1,88	4,06
H3	75,85	7	2,22	3,67
I3	75,07	4	2,26	4,06
A4	92,8	49	13,1	6,86
B4	92,4	12	8,33	10,66
C4	94,7	95	6,15	14,29
D4	88,5	15	3,38	4,31
E4	85,3	1	3,34	3,57
F4	87,9	3	3,1	4,91
G4	79,64	7	1,59	2,32
H4	78,82	6	1,32	1,86
I4	75,72	8	1,51	1,72

Tabela 6.3. Resultados para o Problema PIDLS

Como pode ser percebido na tabela 6.4, os desvios superiores a 20% não são influenciados diretamente pelos tempos de preparação dependentes da sequência, já que a baixa qualidade dos desvios é mantida quando os mesmos são retirados. Por sua vez, a existência desses tempos potencializa o desvio entre os limites dual e primal, uma vez que a existência desses tempos reduz a já escassa capacidade de tempo. A influência dos tempos de preparação dependentes da sequência pode ser observada na comparação entre a ocupação média das máquinas quando os problemas PIDLS e PDLCAMP são resolvidos.

6.2.2 Comparação com a Literatura

Na tabela 6.5, são comparados os resultados de Belisário et al. [2009] com os resultados alcançados pela procedimento heurístico implementado. As soluções marcadas com asterisco são ótimas. Essas instâncias não consideram custos de produção e nem tempos de preparação para os primeiros produtos fabricados. A ausência dos tempos de preparação faz com o que os limites duais gerados sejam fracos, pois o problema PDLCAMP que mantém a característica de ser um limite inferior para o PIDLS não deve apresentar tempos de preparação. Por esse motivo, por não representarem informação relevante, os limites duais foram omitidos.

No que se refere aos resultados numéricos, quando se compara as melhores soluções encontradas por alguma das heurísticas implementadas em Belisário et al. [2009], cujos valores são anotados na coluna *D. Liter. (%)*, observa-se que o procedimento de solução proposto nesta dissertação conseguiu superar quase todos os resultados. A coluna *Cplex* exibe os melhores resultados que o *solver* Cplex conseguiu achar até alcançar o tempo máximo de 1 hora. Como se pode notar, foram encontradas soluções competitivas tanto em tempo quando em qualidade. Também se observa que as piores soluções encontradas são para instâncias cujas ocupações médias chegam a quase 100%. Instâncias que apresentam ocupação elevada muitas vezes produzem soluções que apresentam produção atrasada no último período. A escolha do melhor produto dentre aqueles que geram excessos não foi priorizada no procerdiemtno de busca, assim, soluções que apresentam excesso tendem a não serem tão boas.

Cenário	C. PDLCAMP (%)	T. PDLCAMP	Min. PDLCAMP	Max. PDCLMP
A1	91,4	0	22,52	26,8
B1	88,99	0	9,6	14,6
C1	87,3	0	9,9	20,34
D1	83,9	0	5,22	18,78
E1	83,6	0	4,63	21,52
F1	79,8	0	2,51	16,65
G1	76,24	0	1,62	14,3
H1	74,87	0	1,59	12,92
I1	72,18	0	1,86	6,08
A2	98,3	0	14,88	21,61
B2	88,8	18	8,3	29,67
C2	89,5	0	7,83	26,22
D2	84,5	0	4,3	11,47
E2	83,9	0	6,17	7,41
F2	80,7	0	4,6	6,35
G2	78,26	0	1,21	4,82
H2	77,61	0	1,56	5,56
I2	76,46	0	1,44	5,34
A3	89,1	3	5,48	10,95
B3	87,6	15	6,51	9,45
C3	88,8	18	4,83	8,84
D3	81,9	8	3,45	5,73
E3	82,1	3	3,17	5,36
F3	81,1	3	2,81	5,81
G3	78,26	0	1,96	3,98
H3	73,66	5	1,9	3,39
I3	71,42	6	1,95	3,42
A4	96,4	29	6,93	12,2
B4	92,51	45	6,12	8,75
C4	93,1	6	4,59	8,35
D4	88,3	6	3,36	4,59
E4	84,7	8	2,68	3,4
F4	85,5	9	2	3,55
G4	78,73	3	1,71	2,31
H4	78	2	1,37	1,95
I4	74,68	5	1,31	1,7

Tabela 6.4. Resultados para o Problema PDLCAMP

Instância	Cplex	Desvio (%)	D. Liter. (%)	Tempo	Ocupação
inst2.5i.dat	18300	-6,91	-9,77	22	94,18
inst2.4i.dat	6633	-2,5	-0,2	0	84,16
inst1.0i.dat	1373,54*	0,61	0,33	0	23,35
inst2.6i.dat	68629	-11,05	-10,03	23	94,37
inst2.0i.dat	2394	1,16	15,37	30	17,89
inst2.1i.dat	3555	0,95	15,81	4	34,67
inst2.3i.dat	5688	-0,37	2,51	10	66,47
inst1.1i.dat	1682,99*	0,3	1,02	0	37,44
inst1.5i.dat	59693*	0,2	0,02	0	99,85
inst1.4i.dat	40980	-0,01	0	0	98,13
inst1.6i.dat	105047	0	0	0	99,79
inst1.3i.dat	25299,06*	1,12	0	0	87,44
inst2.2i.dat	4598	0,76	5,36	10	46,38

Tabela 6.5. Comporação com os resultados de Belisário et al. [2009]

Capítulo 7

Conclusões e Trabalhos Futuros

Este trabalho abordou o Problemas Integrado de Dimensionamento de Lotes e Sequenciamento com tempos de preparação dependentes da sequência, considerando-se máquinas paralelas não relacionadas. O problema fundamental em discussão consiste em analisar a influência dos tempos de preparação nos custos finais do planejamento de produção. Para efetuar tal análise, os dois problemas de *Dimensionamento de Lotes* e *Sequenciamento em Máquinas Paralelas*, que são resolvidos, em geral, separadamente, são agrupados numa formulação integrada. Essa formulação agrega as restrições de sequenciamento ao modelo clássico de dimensionamento.

O modelo proposto por Mateus et al. [2010] para o problema integrado foi modificado para considerar tempos de preparação para os primeiros produtos fabricados no período. Como o estado de preparação das máquinas não é conservado entre períodos consecutivos, esse tempo independe da sequência.

Para resolver o problema integrado, uma estratégia heurística é proposta. O procedimento parte de uma solução inicial inviável (na maioria dos casos), restaura a viabilidade e, finalmente, efetua uma busca local. Para auxiliar na geração da solução inicial, foi implementado um algoritmo de programação dinâmica para um problema de dimensionamento não capacitado. A viabilidade é restabelecida através de movimentos básicos de transferência de lotes e inversão de posições de tarefas nas máquinas. A busca local foi desenvolvida tomando como inspiração a metaheurística Busca Tabu, que realiza os movimentos básicos mencionados na exploração das vizinhanças.

Na segunda parte do trabalho, propomos limites duais para o Problema Integrado de Dimensionamento de Lotes e Sequenciamento. Uma relaxação combinatória é desenvolvida para o modelo integrado com a remoção dos tempos de preparação dependentes da sequência. Restando apenas os tempos de preparação independentes, o problema resultante trata-se do *Problema de Dimensionamento de Lotes Capacitado com Atraso* *em Máquinas Paralelas*. O limites são obtidos por meio das relaxações lineares e Lagrangeana desse problema, uma vez que o mesmo é *NP-difícil*. Dessa maneira, também foi implementado um procedimento capaz resolver essa relaxação Lagrangeana. O algoritmo gerado baseia-se no Método dos Subgradientes e foi desenvolvido unicamente para obtenção dos limites.

Não se tem conhecimento de outros problemas da literatura que levam em conta as mesmas premissas aqui consideradas. Portanto, os desempenhos das heurísticas e limites duais foram avaliados sobre um conjunto de instâncias de teste geradas artificialmente. Para verificar o efeito dos tempos de preparação dependentes da sequência na qualidade das soluções e limites, essas instâncias possuem capacidade de tempo nos níveis alto, médio e baixo, assim como os tempos de preparação independentes foram construídos a uma distância alta, média e baixa dos tempos dependentes da sequência.

Como trabalhos futuros, pretende-se utilizar as soluções geradas no algoritmo Lagrangeano para gerar soluções de partida para a heurística. Partindo de pontos iniciais mais diversificados, espera-se melhorar a eficiência do procedimento. Em termos de tempos de preparação, o problema pode ser modificado para conservar o estado de preparação da máquina em períodos consecutivos. Esse tipo de problema, com transição de preparação (*setup carryover*), é mais difícil de resolver, porém pode ser encontrado em diversas situações práticas em que não se dispõe de tempo hábil para reiniciar as máquinas. O método de obtenção de soluções primais proposto também pode ser avaliado, com algumas modificações, em problemas que consideram custos de preparação dependentes da sequência.

Referências Bibliográficas

Aarts, E. & Van Laarhoven, P. (1987). Simulated annealing: Theory and applications. *Reidel, Dordrecht, The Netherlands.*

Aggarwal, A. & Park, J. (1993). Improved algorithms for economic lot size problems. *Operations Research*, 41(3):549--571.

Ali, A.; Padman, R. & Thiagarajan, H. (1989). Dual algorithms for pure network problems. *Operations Research*, 37(1):159--171.

Allahverdi, A.; Ng, C.; Cheng, T. & Kovalyov, M. (2008). A survey of scheduling problems with setup times or costs. *European Journal of Operational Research*, 187(3):985--1032.

Applegate, D.; Bixby, R.; Chvátal, V. & Cook, W. (2006). *The traveling salesman problem: a computational study*. Princeton University Press.

Armentano, V. & Toledo, F. (1997). Dynamic Programming Algorithms for the Parallel Machine Lot Sizing Problem. *Pesquisa Operacional*, 17:137--149.

Belisário, L.; Guedes, L.; Mateus, G. & Souza, M. (2009). Heurísticas para o problema de dimensionamento de lotes e sequenciamento de máquinas. *Anais do Simpósio Brasileiro de Pesquisa Operacional*, 41(3):1--5.

Bitran, G. & Matsuo, H. (1986). Approximation formulations for the single-product capacitated lot size problem. *Operations Research*, 34(1):63--74.

Brüggemann, W. & Jahnke, H. (2000). The discrete lot-sizing and scheduling problem: Complexity and modification for batch availability. *European Journal of Operational Research*, 124(3):511--528.

Brucker, P. (2004). *Scheduling Algorithms*. Springer.

Cheng, T. & Sin, C. (1990). A state-of-the-art review of parallel-machine scheduling research. *European Journal of Operational Research*, 47(3):271--292.

Christofides, N. (1976). *Worst-Case Analysis of a New Heuristic for the Travelling Salesman Problem.* Management Sciences Research Group, Graduate School of Industrial Administration, Carnegie-Mellon University.

Croes, G. (1958). A method for solving traveling-salesman problems. *Operations Research*, pp. 791--812.

Dobson, G. (1987). The economic lot-scheduling problem: achieving feasibility using time-varying lot sizes. *Operations Research*, 35(5):764--771.

Drexl, A. & Haase, K. (1995). Proportional lotsizing and scheduling. *International Journal of Production Economics*, 40(1):73--87.

Drexl, A. & Kimms, A. (1997). Lot sizing and scheduling–survey and extensions. *European Journal of Operational Research*, 99(2):221--235.

Dueck, G. & Scheuer, T. (1990). Threshold accepting: A general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, 90(1):161 – 175.

Elmaghraby, S. (1978). The economic lot scheduling problem (ELSP): review and extensions. *Management Science*, 24(6):587--598.

Erlenkotter, D. (1990). Ford Whitman Harris and the economic order quantity model. *Operations Research*, 38(6):937--946.

Evans, J. (1985). An efficient implementation of the Wagner-Whitin algorithm for dynamic lot-sizing. *Journal of Operations Management*, 5(2):229--235.

Federgruen, A. & Tzur, M. (1991). A Simple Forward Algorithm to Solve General Dynamic Lot Sizing Models with n Periods in 0 (n \log n) or 0 (n) Time. *Management Science*, pp. 909--925.

Fisher, M. L. (1981). The lagrangian relaxation method for solving integer programming problems. *Management Science*, 27:1–18.

Fleischmann, B. & Meyr, H. (1997). The general lotsizing and scheduling problem. *OR Spectrum*, 19:11–21.

França, P. M.; Armentano, V. A.; Berretta, R. E. & Clark, A. R. (1997). A heuristic method for lot-sizing in multi-stage systems. *Computers and Operations Research*, 24(9):861 – 874.

Glover, F.; Laguna, M. et al. (1997). *Tabu search*. Springer.

Haase, K. (1996). Capacitated lot-sizing with sequence dependent setup costs. *OR Spectrum*, 18(1):51--59.

Hansen, P. & Mladenovi, N. (2001). Variable neighborhood search: Principles and applications. *European journal of operational research*, 130(3):449--467.

Harris, F. (1990). How many parts to make at once. *Operations Research*, 38(6):947--950.

Hsu, W. (1983). On the general feasibility test of scheduling lot sizes for several products on one machine. *Management Science*, 29:93–105.

Karimi, B.; Fatemi Ghomi, S. & Wilson, J. (2003). The capacitated lot sizing problem: a review of models and algorithms. *Omega*, 31(5):365--378.

Karmarkar, U.; Kekre, S. & Kekre, S. (1987). The dynamic lot-sizing problem with startup and reservation costs. *Operations Research*, 35(3):389--398.

Khouja, M.; Michalewicz, Z. & Wilmot, M. (1998). The use of genetic algorithms to solve the economic lot size scheduling problem. *European Journal of Operational Research*, 110(3):509--524.

Lawler, E.; Lenstra, J.; Kan, A. & Shmoys, D. (1993). Sequencing and scheduling: Algorithms and complexity. *Logistics of production and inventory*, 4:445--522.

Maes, J.; McClain, J. & Van Wassenhove, L. (1991). Multilevel capacitated lotsizing complexity and LP-based heuristics. *European Journal of Operational Research*, 53:131--148.

Marinakis, Y.; Migdalas, A. & Pardalos, P. (2005a). A Hybrid GeneticGRASP Algorithm Using Lagrangean Relaxation for the Traveling Salesman Problem. *Journal of Combinatorial Optimization*, 10(4):311--326.

Marinakis, Y.; Migdalas, A. & Pardalos, P. (2005b). Expanding neighborhood GRASP for the traveling salesman problem. *Computational Optimization and Applications*, 32(3):231--257.

Mateus, G.; Ravetti, M.; de Souza, M. & Valeriano, T. (2010). Capacitated lot sizing and sequence dependent setup scheduling: an iterative approach for integration. *Journal of Scheduling*, 13(3):245--259.

Meyr, H. (2000). Simultaneous lotsizing and scheduling by combining local search with dual reoptimization. *European Journal of Operational Research*, 120(2):311--326.

Meyr, H. (2002). Simultaneous lotsizing and scheduling on parallel machines. *European Journal of Operational Research*, 139(2):277--292.

Mokotoff, E. (2001). Parallel machine scheduling problems: a survey. *Asia Pacific Journal of Operational Research*, 18(2):193--242.

Paula, M. R. (2008). Heurísticas para a minimização dos atrasos em sequenciamento de máquinas paralelas com tempos de preparação dependentes da sequência. Master's thesis, Universidade Federal de Minas Gerais.

Ravetti, M. G. (2007). *Algoritmos para o Problema de Sequenciamento com Máquinas Paralelas e Tempos de Preparação Dependentes da Sequência*. PhD thesis, Universidade Federal de Minas Gerais.

Rocha, P.; Ravetti, M.; Mateus, G. & Pardalos, P. (2008). Exact algorithms for a scheduling problem with unrelated parallel machines and sequence and machine-dependent setup times. *Computers and Operations Research*, 35(4):1250--1264.

Sabbag Neto, Z. (1993). Planejamento da produção em máquinas paralelas sob restrições de capacidade. Master's thesis, Universidade Estadual de Campinas.

Sung, C. (1986). A single-product parallel-facilities production-planning model. *International Journal of Systems Science*, 17:983--989.

Toledo, F. & Armentano, V. (2006). A Lagrangian-based heuristic for the capacitated lot-sizing problem in parallel machines. *European Journal of Operational Research*, 175(2):1070--1083.

Toledo, F. & Shiguemoto, A. (2005). Lot-sizing problem with several production centers. *Pesquisa Operacional*, 25:479--492.

Trigeiro, W.; Thomas, L. & McClain, J. (1989). Capacitated lot sizing with setup times. *Management Science*, 35(3):353--366.

Wagelmans, A.; Van Hoesel, S. & Kolen, A. (1992). Economic lot sizing: an O (n log n) algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40:145--156.

Wagner, H. M. & Whitin, T. M. (1958). Dynamic version of the economic lot size model. *Management Science*, 5:89–96.

Wolsey, L. A. (1999). *Integer programming*. John Wiley & Sons.

Zangwill, W. (1969). A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach. *Management Science*, 15(9):506--527.