

**UNIVERSIDADE FEDERAL DE MINAS GERAIS**  
**ESCOLA DE ENGENHARIA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA**

**Gabriela Rodrigues Niquini**

**RESOLUÇÃO DO PROBLEMA DO SEQUENCIAMENTO DE LOTES**  
**APLICADO À PRODUÇÃO DE PAPEL-CARTÃO**

**BELO HORIZONTE - MG**  
**2021**

**Gabriela Rodrigues Niquini**

**RESOLUÇÃO DO PROBLEMA DO SEQUENCIAMENTO DE LOTES  
APLICADO À PRODUÇÃO DE PAPEL-CARTÃO**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Química da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito parcial para obtenção do Grau de Mestre em Engenharia Química.

Linha de Pesquisa: Engenharia de Sistemas em Processos

Orientador: Prof. D. Sc. Esly Ferreira da Costa Junior

Coorientador: Prof. Dr. Mauricio Cardoso de Souza

BELO HORIZONTE - MG

2021

N721r	<p>Niquini, Gabriela Rodrigues.  Resolução do problema de sequenciamento de lotes aplicado à produção de papel-cartão [recurso eletrônico] / Gabriela Rodrigues Niquini. - 2021.  1 recurso online (99 f. : il., color.) : pdf.</p> <p>Orientador: Esly Ferreira da Costa Junior.  Coorientador: Mauricio Cardoso de Souza.</p> <p>Dissertação (mestrado) - Universidade Federal de Minas Gerais, Escola de Engenharia.</p> <p>Apêndices: f. 85-99.</p> <p>Bibliografia: f. 81-84.  Exigências do sistema: Adobe Acrobat Reader.</p> <p>1. Engenharia química - Teses. 2. Papel - Indústria - Teses.  3. Embalagens - Indústria - Teses. 4. Planejamento - Teses. I. Costa Junior, Esly Ferreira da. II. Souza, Maurício Cardoso de. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.</p> <p style="text-align: right;">CDU: 66.0(043)</p>
-------	--

Ficha catalográfica elaborada pela bibliotecária Roseli Alves de Oliveira CRB/6 2121  
Biblioteca Prof. Mário Werneck, Escola de Engenharia da UFMG



UNIVERSIDADE FEDERAL DE MINAS GERAIS  
ESCOLA DE ENGENHARIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA

### FOLHA DE APROVAÇÃO

## "RESOLUÇÃO DO PROBLEMA DE SEQUENCIAMENTO DE LOTES APLICADO À PRODUÇÃO DE PAPEL-CARTÃO"

**Gabriela Rodrigues Niquini**

Dissertação submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Química da Escola de Engenharia da Universidade Federal de Minas Gerais, como parte dos requisitos à obtenção do título de **MESTRE EM ENGENHARIA QUÍMICA**.

#### 283ª DISSERTAÇÃO APROVADA EM 20 DE ABRIL DE 2021 POR:



Documento assinado eletronicamente por **Esly Ferreira da Costa Junior, Professor do Magistério Superior**, em 20/04/2021, às 16:59, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Silvio Alexandre de Araujo, Usuário Externo**, em 20/04/2021, às 17:00, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Oswaldo Vieira, Usuário Externo**, em 20/04/2021, às 17:03, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Maurício Cardoso de Souza, Coordenador(a) de curso de pós-graduação**, em 20/04/2021, às 17:05, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site [https://sei.ufmg.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **0649955** e o código CRC **C23C10EC**.

## **AGRADECIMENTOS**

Aos meus pais e ao meu irmão, pelo apoio incondicional e por terem sido sempre meu suporte, não apenas nos dois anos de mestrado, mas durante toda a minha vida.

Ao meu orientador Esly pela solicitude e por ter compartilhado seu conhecimento comigo ao longo desses anos de parceria. Ao coorientador Mauricio pela contribuição decisiva neste trabalho. Agradeço também aos professores do Departamento de Engenharia Química, que contribuíram para minha formação e para que eu chegasse até aqui e acreditaram no meu trabalho. À secretária Fernanda, por ser sempre gentil e eficiente.

Aos membros da banca, pelo esforço para que este texto e o trabalho sejam aprimorados. Ao Osvaldo e ao Claudeci, pela boa vontade em ajudar e por ter contribuído para estabelecer a parceria entre academia e indústria.

A todos os amigos. Ao Rafael e à Marina, por terem sido sempre companhia e por terem tornado meus dias na UFMG mais alegres. Agradeço também aos companheiros do MOP e dos outros grupos de pesquisa do PPGEQ pelos cafés, pelas horas agradáveis de trabalho e pelos momentos de lazer.

Ao CNPq pelo apoio financeiro.

## RESUMO

O papel-cartão é um material de ampla utilização no mercado de embalagens, o qual se encontra em ascensão sólida neste século, devido ao aumento global no consumo de produtos industrializados e de compras realizadas pela internet. Na pandemia do novo coronavírus, que atinge o planeta desde o final do ano de 2019, observou-se crescimento na demanda de papel para embalagens, em função da essencialidade dos setores alimentícios e de saúde e da maior utilização de serviços de entrega de alimentos a domicílio. Tendo em vista que o preço do papel é definido pelo mercado e a diferenciação de preços é pequena, o planejamento otimizado da produção é decisivo para redução das despesas e aumento da margem de lucro. Para o desenvolvimento deste trabalho, utilizaram-se dados fornecidos por uma indústria de papel para embalagens localizada no Brasil, a qual produz papéis-cartão de variadas gramaturas e composições. O objetivo deste trabalho consiste em resolver o problema de sequenciamento de lotes em uma máquina de papel-cartão, de modo a minimizar custos relativos às perdas acarretadas pelas trocas entre produtos na máquina. Alguns dos produtos sofrem retiradas periódicas por parte de seu cliente, havendo a necessidade de manutenção de um determinado nível de estoque para os mesmos. Para a resolução do problema, foram propostos dois métodos. Um deles consiste em um problema de programação inteira mista, baseado no problema do caixeiro viajante com coleta de prêmios, e o outro trata-se de uma heurística de construção gulosa, aleatorizada e adaptativa. Ambos os métodos propiciaram a obtenção de sequências com tempo total de *setup* consideravelmente inferior ao das sequências fornecidas pela indústria, respeitando as restrições operacionais e atendendo às demandas estabelecidas.

**Palavras-chave:** Indústria de papel; planejamento operacional; *setup* dependente da sequência.

## ABSTRACT

Paperboard is a broadly utilized material in packaging industry. Such industry is on solid rise in this century, due to the global increase in purchase of industrialized products and e-commerce. As a consequence of coronavirus pandemic, which started on late 2019, packaging paper demand has increased, due to the essentiality of food and health supplies and greater utilization of delivery services. Since paper price is defined by market and the differentiation is small among industries, the optimization of production planning plays a decisive role in reduction of expenses and increase of profit. For the development of this work, data were provided by a packaging paper industry in Brazil, which produces paperboard of various grammages and compositions. The objective of this work consists in solving the lot scheduling problem in a paperboard machine, aiming to minimize costs related to losses caused by changeovers between products. Some of the products are constantly withdrawn by the client, therefore a minimum stock level must be held for those. Two methods were proposed to solve the scheduling problem. One is a mixed integer programming formulation, based on the prize collecting traveling salesman problem, and the other one is a greedy, randomized and adaptive construction heuristic. Both methods led to sequences which respected the operational constraints and met the established demands, and a considerable decrease in the total setup time was observed.

**Keywords:** Paper industry; operational planning; sequence-dependent setup.

## LISTA DE FIGURAS

Figura 1 - Produção brasileira por tipo de papel .....	15
Figura 2 - Destino da produção nacional de papel por categoria .....	19
Figura 3 - Processo de produção de papel .....	21
Figura 4 - Principais decisões envolvidas no sequenciamento de atividades .....	22
Figura 5 - Balanço de custos no dimensionamento de lotes .....	24
Figura 6 - Modelagem do estoque para carregamento e consumo a taxas constantes .....	25
Figura 7 - Exemplo de problema do caixeiro viajante com 5 vértices .....	33
Figura 8 - Pseudocódigo para a meta-heurística GRASP .....	37
Figura 9 - Pseudocódigo para a etapa de construção do método GRASP .....	38
Figura 10 - Pseudocódigo para o procedimento de busca local no método GRASP .....	39
Figura 11 - Representação gráfica das sequências obtidas pela PIM .....	62
Figura 12 - Estoque previsto e estoque de segurança dos produtos K205 pela PIM .....	64
Figura 13 - Estoque previsto e estoque de segurança dos produtos K274 pela PIM .....	65
Figura 14 - Menores valores obtidos para o tempo de <i>setup</i> para cada tamanho máximo da LRC .....	68
Figura 15 - Estoque previsto e estoque de segurança dos produtos K205 pela heurística .....	72
Figura 16 - Estoque previsto e estoque de segurança dos produtos K274 pela heurística .....	73



## LISTA DE TABELAS

Tabela 1 - Definições aplicadas aos ambientes de máquinas .....	26
Tabela 2 - Modelos para o problema integrado de dimensionamento e sequenciamento de lotes .....	30
Tabela 3 - Produtos considerados nas instâncias.....	41
Tabela 4 - Demanda mensal dos produtos, em toneladas.....	56
Tabela 5 - Estoques iniciais e de segurança e previsão de demanda para os produtos K205 e K274 .....	57
Tabela 6 - Taxas de produção, em toneladas por hora, de cada espécie de papel-cartão .....	57
Tabela 7 - Tempo de <i>setup</i> , em minutos, para os pares de papel-cartão .....	58
Tabela 8 - Sequências fornecidas pela indústria.....	59
Tabela 9 - Formulação das especificidades das instâncias .....	61
Tabela 10 - Sequências obtidas para cada mês pela PIM .....	62
Tabela 11 - Comparação das sequências fornecidas pela indústria com os resultados obtidos pela PIM .....	63
Tabela 12 - Sequências de menor tempo de <i>setup</i> obtidas no teste para dimensão da LRC ....	69
Tabela 13 - Comparação dos resultados da heurística com as soluções exatas.....	69
Tabela 14 - Tempo computacional para os testes realizados.....	70
Tabela 15 - Sequências obtidas no <i>pool</i> para o mês 1 .....	74
Tabela 16 - Sequências obtidas no <i>pool</i> para o mês 2 .....	75
Tabela 17 - Sequências obtidas no <i>pool</i> para o mês 3 .....	76
Tabela 18 - Sequências obtidas no <i>pool</i> para o mês 4 .....	77
Tabela 19 - Sequências obtidas para o <i>pool</i> no mês 4, fixando-se o primeiro produto .....	78

## LISTA DE SIGLAS

<i>CLSD</i>	Problema capacitado de dimensionamento de lotes com <i>setup</i> dependente da sequência <i>Capacitated lot sizing problem with sequence-dependent setups</i>
<i>CLSP</i>	Problema capacitado de dimensionamento de lotes <i>Capacitated lot sizing problem</i>
<i>CSLP</i>	Problema de dimensionamento de lotes com <i>setup</i> contínuo <i>Continuous setup lot sizing problem</i>
<i>DLSP</i>	Problema discreto de dimensionamento e sequenciamento de lotes <i>Discrete lot sizing and scheduling problem</i>
<i>ELSP</i>	Problema de sequenciamento de lotes econômicos <i>Economic lot scheduling problem</i>
<i>EOGLSP</i>	Problema geral de dimensionamento e sequenciamento de lotes orientado à energia <i>Energy-oriented general lot sizing and scheduling problem</i>
<i>EWO</i>	Otimização global da empresa <i>Enterprise-wide optimization</i>
<i>GLSP</i>	Problema geral de dimensionamento e sequenciamento de lotes <i>General lot sizing and scheduling problem</i>
<i>GRASP</i>	Procedimento de busca guloso, aleatorizado e adaptativo <i>Greedy randomized adaptive search procedure</i>
<i>IBÁ</i>	Indústria Brasileira de Árvores
<i>ISO</i>	Organização internacional para padronização <i>International Organization for Standardization</i>
<i>LRC</i>	Lista restrita de candidatos
<i>MIP</i>	<i>Mixed-integer programming</i> (equivalente a PIM)
<i>PCP</i>	Planejamento e controle da produção
<i>PCV</i>	Problema do caixeiro viajante
<i>PCTSP</i>	<i>Prize collecting traveling salesman problem</i> (equivalente a PCVCP)
<i>PCVCP</i>	Problema do caixeiro viajante com coleta de prêmios
<i>PEBD</i>	Polietileno de baixa densidade
<i>PIM</i>	Programação inteira mista
<i>PLSP</i>	Problema proporcional de dimensionamento e sequenciamento de lotes <i>Proportional lot sizing and scheduling problem</i>
<i>RCL</i>	<i>Restricted candidate list</i> (equivalente a LRC)
<i>TSP</i>	<i>Traveling salesman problem</i> (equivalente a PCV)

## LISTA DE SÍMBOLOS

### Parâmetros globais

$a_j$	taxa de produção de um produto $j$	[ton/h]
$b_j$	taxa de demanda de um produto $j$ com retiradas contínuas	[ton/h]
$d_j$	demanda total de um produto $j$	[ton]
$h_j$	estoque inicial de um produto $j$ com retiradas contínuas	[ton]
$l_j$	lote mínimo de um produto $j$	[ton]
$\bar{l}_j$	lote máximo de um produto $j$	[ton]
$m_j$	estoque de segurança de um produto $j$ com retiradas contínuas	[ton]
$s_{ij}$	tempo de <i>setup</i> de um produto $i$ para um produto $j$	[h]

### Símbolos da programação inteira mista

$f_{ij}^q$	variável de fluxo para a rota do problema do caixeiro viajante	[-]
$Q$	número máximo de ciclos	[-]
$t_{jq}$	tempo de produção do produto $j$ no ciclo $q$	[h]
$x_{ij}^q$	se o produto $j$ é ou não imediato sucessor do produto $i$ no ciclo $q$	[-]
$y_{jq}$	se o produto $j$ é ou não fabricado no ciclo $q$	[-]
$z_{ij}^q$	se o produto $i$ é ou não o último do ciclo $q$ , e o $j$ é o primeiro do ciclo $q + 1$	[-]

### Símbolos da heurística

$dk_j$	demanda destinada a retiradas constantes de um produto $j$	[ton]
$dt_j$	demanda temporal de um produto $j$	[h]
$f_{ij}$	se a troca do produto $i$ para o produto $j$ é ou não permitida	[-]
$\bar{lt}_j$	maior lote disponível do produto $j$	[h]
$td_j$	tempo para que o produto $j$ atinja o estoque de segurança	[h]
$tl_j$	tempo para alocar um produto $j$ entre o último alocado e o próximo a atingir o estoque de segurança	[h]
$tt_j$	tempo para alocar um produto $j$ entre dois de demanda contínua	[h]

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>14</b>
1.1. OBJETIVOS .....	16
1.2. ORGANIZAÇÃO DO TRABALHO.....	16
<b>2. REVISÃO DA LITERATURA.....</b>	<b>18</b>
2.1. PRODUÇÃO E UTILIZAÇÃO DO PAPEL.....	18
2.1.1. Panorama da indústria de papel e celulose no Brasil.....	18
2.1.2. Processo produtivo do papel.....	18
2.1.3. Papel-cartão .....	20
2.2. PLANEJAMENTO E CONTROLE DA PRODUÇÃO .....	22
2.2.1. Dimensionamento de lotes.....	23
2.2.2. Sequenciamento .....	25
2.2.3. Problema integrado de dimensionamento e sequenciamento de lotes .....	28
<b>3. REFERENCIAL TEÓRICO .....</b>	<b>32</b>
3.1. PROBLEMA DO CAIXEIRO VIAJANTE.....	32
3.2. PROBLEMA DO CAIXEIRO VIAJANTE COM COLETA DE PRÊMIOS .....	34
3.3. HEURÍSTICAS DE CONSTRUÇÃO .....	36
<b>4. PROBLEMA ABORDADO .....</b>	<b>40</b>
4.1. DEFINIÇÃO DO PROBLEMA .....	40
4.2. ABORDAGEM DE PROGRAMAÇÃO INTEIRA MISTA .....	42
4.3. ABORDAGEM HEURÍSTICA .....	47
<b>5. ESTUDO COMPUTACIONAL .....</b>	<b>55</b>
5.1. AMBIENTES DE TESTE .....	55
5.2. PARÂMETROS UTILIZADOS NAS INSTÂNCIAS .....	55
5.3. IMPLEMENTAÇÃO DO MÉTODO DE PROGRAMAÇÃO INTEIRA MISTA ...	60
5.3.1. Implementação das variáveis e restrições .....	60
5.3.2. Verificação da manutenção de estoque.....	63
5.4. IMPLEMENTAÇÃO DA HEURÍSTICA DE CONSTRUÇÃO.....	65
5.4.1. Implementação das variáveis e restrições .....	65
5.4.2. Avaliação da dimensão da lista restrita de candidatos .....	67
5.4.3. Sequências obtidas e comparação com a PIM .....	68
5.4.4. Verificação da manutenção de estoque.....	71

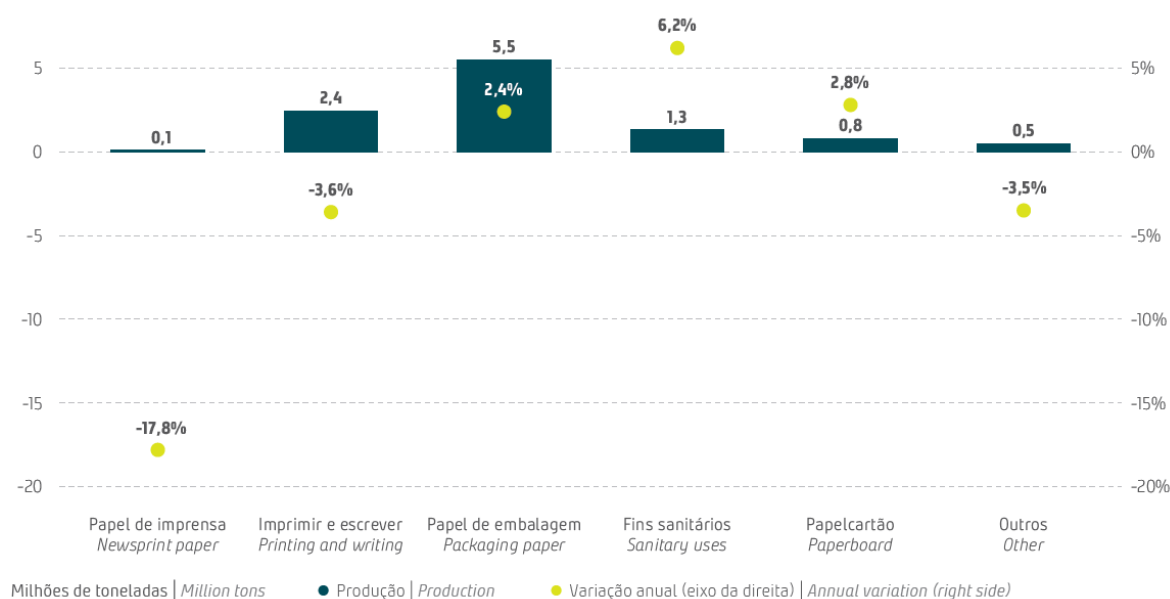
5.4.5. Obtenção de sequências alternativas .....	73
5.4.6. Teste adicional para a instância do mês 4 .....	77
<b>6. CONSIDERAÇÕES FINAIS.....</b>	<b>79</b>
6.1. CONTRIBUIÇÕES DO TRABALHO .....	80
6.2. SUGESTÕES PARA TRABALHOS FUTUROS .....	80
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>81</b>
<b>APÊNDICES .....</b>	<b>85</b>

## 1. INTRODUÇÃO

O preço de venda do papel – seja papel para escrita, papelão, papel-cartão, dentre outros – é determinado pelo mercado, de modo que, para a obtenção de maiores margens de lucro, as empresas devem reduzir tanto quanto possível os custos de produção, ao mesmo tempo em que a qualidade dos produtos é mantida. Uma fonte de redução de custos é o planejamento otimizado da produção, em que se define a melhor forma de execução dos processos, nos mais diversos níveis, tendo em mente um objetivo predeterminado. Na prática, o planejamento global da produção é realizado de forma manual pelo corpo técnico da indústria, por meio da análise de planilhas e com base em sua experiência, embora haja uma série de algoritmos que podem ser empregados para a otimização da produção (HARJUNKOSKI *et al.*, 2014; FIGUEIRA *et al.*, 2015).

O consumo global de papel para embalagens – principalmente papel ondulado (papelão) e papel-cartão – e de papel para uso sanitário apresentou crescimento sólido neste século. Isso se deve, segundo CAPO (2018), ao desenvolvimento da Ásia. A autora ainda projeta um crescimento para os próximos anos, em função da expansão das compras pela internet (*e-commerce*) e à urbanização dos mercados em desenvolvimento. Em oposição a outros materiais utilizados em embalagens, como plástico e vidro, o papel é obtido a partir de matéria-prima renovável e possui um menor tempo de decomposição no meio-ambiente, características interessantes do ponto de vista da sustentabilidade. No Brasil, houve crescimento na produção de papel para embalagens nos últimos anos, acompanhando a tendência global. De acordo com o relatório da Indústria Brasileira de Árvores (IBÁ), houve, em 2019, aumento de 2,4% na produção de papel para embalagem (papel ondulado) e 2,8% na produção de papel-cartão em relação ao ano anterior (IBÁ, 2020). A Figura 1 mostra a produção nacional em toneladas, no ano de 2019, e a variação em relação ao ano anterior para as diferentes finalidades de papel.

Desde o final do ano de 2019 até o momento da redação deste trabalho, o planeta atravessa a maior crise sanitária dos últimos cem anos, devido à pandemia do novo coronavírus. A pandemia tem causado uma série de impactos negativos na economia mundial, afetando inúmeros setores. No entanto, a indústria de papel e celulose manteve sua atividade, devido à essencialidade dos setores de alimentação, higiene e saúde, destacando-se a produção de papéis para embalagens e de papéis para fins sanitários (*tissue*) (CAPO, 2021).



**Figura 1** - Produção brasileira por tipo de papel. Fonte: IBÁ (2020)

O presente trabalho de mestrado foi desenvolvido com base em dados reais de uma indústria de papel para embalagem, em particular da produção de papéis-cartão de variadas composições e gramaturas distintas. Esta indústria fornece papel-cartão para produtoras de embalagens, nas quais são utilizados na fabricação de caixas longa-vida para bebidas, como leite e suco, caixas de alimentos e conjuntos de latas ou garrafas de cerveja.

A máquina de papel-cartão considerada neste trabalho opera em regime contínuo, e as trocas entre variedades de papel são processadas com a máquina em funcionamento. Para cada troca, há um período em que o papel-cartão produzido possui qualidade inferior – sendo tratado como refugo –, devido aos ajustes nos insumos utilizados e na gramatura. A duração desse período, chamado tempo de *setup*, depende da variação dos insumos e da diferença de gramatura. É de interesse da indústria reduzir o tempo total de *setup*, pois o custo de troca é fortemente dependente do tempo para estabilização do produto seguinte. Além disso, um menor tempo de *setup* se traduz em redução na geração de rejeitos e otimização da produção por meio do melhor aproveitamento do tempo.

Este trabalho se propõe a realizar o sequenciamento da produção de papel-cartão, a partir de uma demanda predeterminada, com o objetivo de minimizar o tempo total de *setup*. Certas restrições devem ser observadas. A principal delas diz respeito à demanda de alguns dos produtos, os quais sofrem retiradas frequentes por parte do cliente. Para esses produtos, há a necessidade de manutenção de um estoque mínimo, e o planejamento deve ser realizado de maneira a respeitar o nível de estoque exigido. Em adição, o dimensionamento de lotes deve

obedecer a limites inferiores e superiores. Lotes de poucas toneladas são desvantajosos devido à complexidade do processo em máquina de grande porte; por outro lado, lotes extensos podem comprometer a disponibilidade de determinados insumos e acarretar sobrecarga nas rebobinadeiras, etapa subsequente à máquina de papel.

No desenvolvimento do trabalho, foram utilizados algoritmos de otimização, proporcionando a aplicação do conhecimento acadêmico a uma situação pertencente à realidade industrial. Em comparação ao planejamento realizado manualmente pelos operadores, o presente estudo apresenta como vantagens a proposição e a avaliação automatizadas de um amplo número de cenários de planejamento, visando a obtenção de resultados otimizados. O planejamento otimizado possui potencial para a redução de custos e de geração de rejeitos na indústria, representando ganhos econômicos e ambientais.

## 1.1. OBJETIVOS

O objetivo geral deste trabalho consiste na resolução do problema de sequenciamento de lotes, visando a redução do tempo total de *setup*.

Os objetivos específicos são:

- i. compreensão da realidade industrial, com o intuito de elaborar um problema de otimização compatível com a mesma;
- ii. desenvolvimento de um algoritmo de programação inteira mista (PIM) baseado no problema do caixeiro viajante com coleta de prêmios;
- iii. desenvolvimento de uma heurística de construção gulosa e aleatorizada, para a obtenção de sequências viáveis em tempo computacional reduzido;
- iv. aplicação dos algoritmos desenvolvidos às instâncias fornecidas pela indústria.

## 1.2. ORGANIZAÇÃO DO TRABALHO

Esta dissertação conta com seis capítulos. No Capítulo 2, é apresentada uma revisão bibliográfica, a qual trata da produção de papel e do planejamento e controle da produção. No Capítulo 3, é fornecida a base teórica para os dois métodos de otimização utilizados no trabalho. No Capítulo 4, o problema industrial é descrito, e os procedimentos adotados para a otimização



são detalhados. O Capítulo 5 contém detalhes das instâncias utilizadas e os resultados obtidos para a otimização de cada uma delas. Por fim, no Capítulo 6, há as considerações finais.

## **2. REVISÃO DA LITERATURA**

### **2.1. PRODUÇÃO E UTILIZAÇÃO DO PAPEL**

#### **2.1.1. Panorama da indústria de papel e celulose no Brasil**

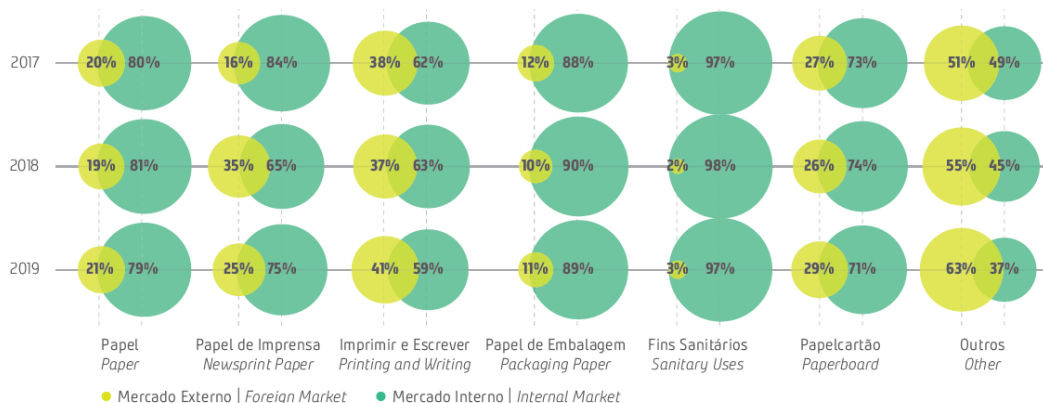
A indústria de árvores plantadas gerou em 2019 uma receita bruta de R\$ 97,4 bilhões, representando 1,2% do Produto Interno Bruto (PIB) brasileiro, tendo sido a atividade industrial de maior contribuição no PIB. Em relação ao ano anterior, foi observado crescimento de 12,6% na arrecadação bruta (IBÁ, 2020). Os produtos derivados da madeira são utilizados na construção civil, na fabricação de móveis, na produção de papel, na geração de energia, entre outros. Além disso, esta indústria fornece produtos não madeireiros, como sementes, óleos e corantes naturais. As variedades de árvore mais plantadas no Brasil são o eucalipto e pinus, correspondendo respectivamente a 6,97 (77%) e 1,64 (18%) dos 9 milhões de hectares de árvores plantadas no Brasil em 2019 (IBÁ, 2020).

O PIB gerado em 2019 pelas indústrias de celulose e papel corresponde a 46,5% da arrecadação do setor de árvores plantadas e é a atividade que mais contribuiu neste setor. Globalmente, o Brasil ocupa lugar de destaque na produção de celulose, tendo sido o segundo maior produtor em 2019, atrás apenas dos Estados Unidos. Na produção de papel, o Brasil ocupou a décima posição nesse mesmo ano (IBÁ, 2020). Enquanto a maior parte da celulose produzida no Brasil destina-se à exportação, a produção de papel é mais voltada para o mercado interno. No ano de 2019, 71% da produção de papel-cartão nacional foi destinada ao mercado interno, e 29%, à exportação. A Figura 2 mostra a distribuição dos papéis de diferentes finalidades entre mercado interno e externo para os anos de 2017, 2018 e 2019. Observa-se que, a não ser pela categoria “Outros”, a maior parte do papel produzido no Brasil é consumida internamente.

#### **2.1.2. Processo produtivo do papel**

A primeira grande etapa do processo de produção de papel é a polpação (ou *pulping*), a qual consiste na liberação das fibras celulósicas. A celulose é encontrada na parede celular da

Elaboração: FGV. Fonte: Ibmá, SECEX. | Prepared by FGV. Source: Ibmá, SECEX.



**Figura 2** - Destino da produção nacional de papel por categoria. Fonte: IBÁ (2020).

madeira, fixada por moléculas de lignina. Na polpação, a lignina é rompida ou dissolvida, e as fibras de celulose são liberadas para que sejam realinhadas na produção do papel.

A polpação pode ocorrer por diversos processos, sendo dois deles o mecânico e o químico. No processo mecânico, toras de madeira com cerca de 1,20 metro de comprimento são friccionadas contra um desfibrador de pedra em rotação, e o calor gerado pela fricção é responsável pelo desprendimento das fibras de celulose. O material resultante consiste em uma pasta de lignina e celulose. Este processo apresenta rendimento elevado, e o papel obtido por essa via possui boas propriedades de impressão e elevada opacidade, porém sua resistência mecânica é reduzida. Ele é utilizado para produtos de uso rápido, como jornais, catálogos e alguns papéis de uso sanitário, bem como para polpas moldadas.

O processo químico, por sua vez, baseia-se na dissolução química da lignina por meio da adição de compostos químicos. O processo mais utilizado desta natureza é o chamado Kraft. Nele, a madeira utilizada encontra-se em forma de cavacos, os quais são levados ao digestor para o processo de cozimento, em conjunto com um reagente conhecido como licor branco, que consiste em uma solução aquosa de hidróxido de sódio (NaOH) e sulfeto de sódio (Na<sub>2</sub>S). O licor branco dissolve a lignina, desprendendo a celulose. Neste caso, o rendimento do processo é inferior, devido à dissolução da lignina no licor branco, porém o papel possui resistência mecânica superior. A polpa obtida é utilizada na produção de papel para escrita e na confecção de papéis para embalagem.

Após a polpação, o processo de branqueamento é realizado em uma fração da polpa, a depender da finalidade do papel a ser obtido. Para a polpa química, o branqueamento é obtido por meio da retirada da lignina residual. Na polpa mecânica, a estrutura das moléculas de lignina é modificada para que o produto torne-se mais claro. Em ambos os casos, o branqueamento é

realizado por oxidação, sendo que na polpa mecânica são utilizados principalmente peróxidos, e na polpa química podem ser utilizados dióxido de cloro, oxigênio, ozônio ou peróxidos (ROBERTSON, 2013). Cabe ressaltar que o processo de branqueamento não é realizado para toda a polpa. No papel-cartão, por exemplo, a camada base e as camadas internas são confeccionadas a partir de polpa não branqueada.

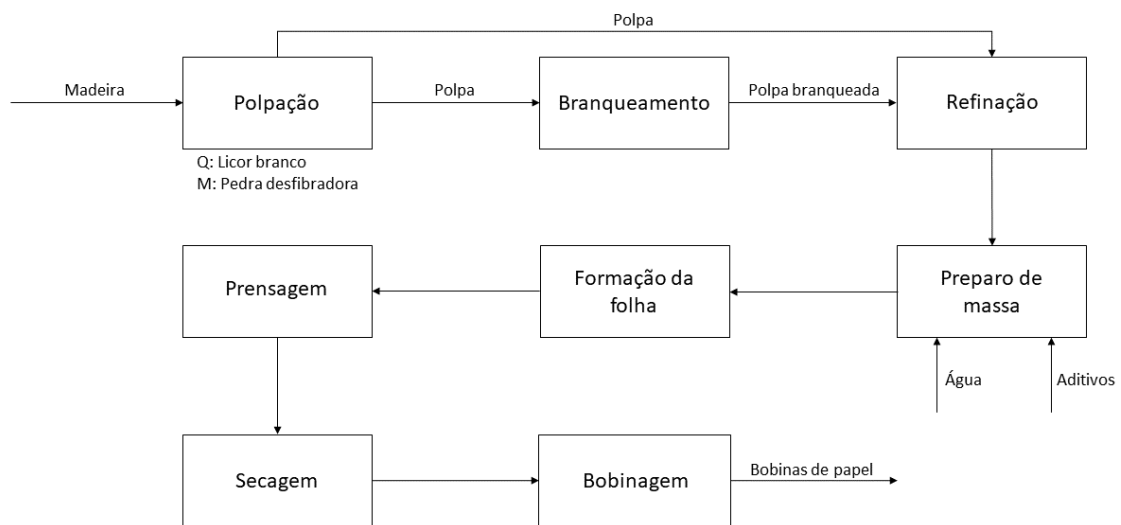
Antes de seguir para a máquina de papel, a celulose passa pelo processo de refinação, em que a área superficial das fibras é elevada. Este processo possui o objetivo de tornar as fibras mais flexíveis, facilitando sua utilização na máquina de papel. Também antes da passagem pela máquina de papel, no chamado preparo de massa, a celulose sofre adição de água e agentes como carbonato de cálcio, materiais adesivos e outros aditivos químicos (SILVA *et al.*, 2015), a depender das propriedades desejadas para o produto final. Além disso, pode haver também a inserção de polpa reciclada de celulose.

Na máquina de papel, ocorre a formação da folha, e há um primeiro processo de desaguamento, em que a água contida no material é eliminada por sucção. Subsequentemente, a folha passa pelos processos de prensagem e secagem, para a remoção de maior volume de água. Na sequência, pode ainda haver a incorporação de outros aditivos. É comum, por exemplo, a adição de amido a papel para escrita, com a finalidade de tornar a superfície mais lisa e reduzir a absorção de água, melhorando a aplicação das tintas.

O papel obtido é levado às rebobinadeiras, onde são alocados em bobinas. A depender da finalidade, há ainda o corte do papel para que ele seja entregue ao cliente no tamanho desejado. Uma representação esquemática simplificada da obtenção de papel é dada na Figura 3. O papel produzido é classificado de acordo com sua composição – a qualidade da celulose utilizada e os aditivos químicos adicionados – e sua gramatura, uma medida da massa de papel por área, usualmente grama por metro quadrado.

### **2.1.3. Papel-cartão**

Segundo a Organização Internacional para Padronização (*ISO – International Organization for Standardization*), o papel passa a ser chamado de papel-cartão para gramaturas superiores a 250 g/m<sup>2</sup> (ATTWOOD, 2009). Entretanto, na prática, esta definição é mais flexível, e papel-cartão é entendido como o papel mais rígido. Ele pode ser feito em uma



**Figura 3** - Processo de produção de papel (Q: Polpação química, M: Polpação mecânica). Fonte: Elaborada pela autora.

única camada, como no caso de algumas cartolinas, porém melhores resultados são obtidos quando a confecção é realizada em múltiplas camadas, devido à combinação das propriedades de cada camada. ROBERTSON (2013) divide os papéis-cartão em três principais categorias, de acordo com sua composição e finalidade: papel para caixas dobráveis, para caixas de bebidas e polpa moldada.

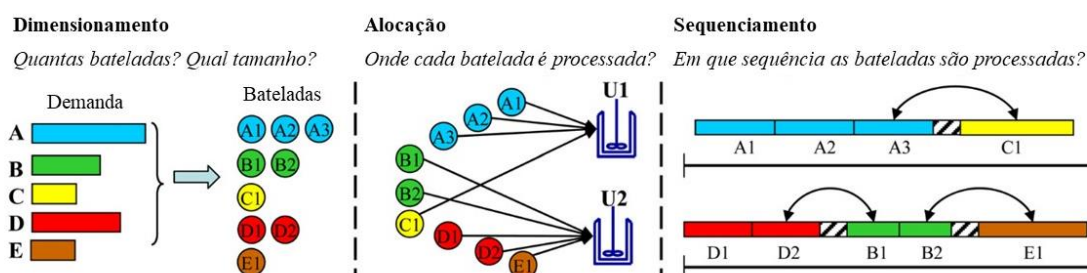
O papel para caixas dobráveis possui estrutura em camadas, podendo conter fibras de diversas qualidades. Em alguns casos, a camada externa contém polpa de qualidade superior, para facilitar a impressão. Um exemplo de papel-cartão utilizado para a confecção dessas caixas são as chamadas folhas *folding box boards*, compostas de camadas internas de polpa mecânica e camadas externas de polpa química, além de um revestimento para facilitar a impressão. As *folding box boards* são utilizadas na confecção de embalagens de chás, biscoitos, medicamentos, alimentos congelados, entre outros.

As folhas utilizadas em caixas longa vida para bebidas são usualmente confeccionadas com múltiplas camadas de polpa química, e as camadas externas são revestidas por polietileno de baixa densidade (PEBD), para garantir a impermeabilidade. O material conta também com uma folha de alumínio, que funciona como uma barreira para a luminosidade e os gases. Por fim, as polpas moldadas são fabricadas com polpas químicas ou mecânicas, além de fibras recicladas. Em vez de folhas, elas são fabricadas em moldes, e são utilizadas em produtos como pentes de ovos e bandejas de frutas.

## 2.2. PLANEJAMENTO E CONTROLE DA PRODUÇÃO

A otimização global da empresa (*EWO – enterprise-wide optimization*) é uma área do conhecimento que se situa na interface entre a Engenharia Química (Engenharia de Sistemas em Processos) e Pesquisa Operacional (GROSSMANN, 2005). Esta área tem por objetivo otimizar os processos industriais, desde decisões relacionadas à estratégia da empresa a longo prazo, até o controle em tempo real de cada unidade da planta.

O planejamento em um empreendimento deve ser realizado desde o nível estratégico – em que são definidos os produtos a serem fabricados, a localização das unidades fabris e a aquisição de equipamentos, dentre outros – até o nível operacional, em que são tratadas as decisões cotidianas relativas aos processos. O foco deste trabalho encontra-se no planejamento operacional. Segundo HARJUNKOSKI *et al.* (2014), as principais decisões nessa esfera são o dimensionamento de lotes (duração, se batelada; taxa de produção, se contínuo), a atribuição de tarefas a equipamentos e o sequenciamento das tarefas em cada equipamento, conforme representação esquemática dada na Figura 4.



**Figura 4** - Principais decisões envolvidas no sequenciamento de atividades. Fonte: Adaptada de HARJUNKOSKI *et al.* (2014).

Concomitantemente ao planejamento, é fundamental a realização do controle da produção, e, por esse motivo, essa área do conhecimento é frequentemente chamada de planejamento e controle da produção (PCP). Segundo SLACK, BRANDON-JONES e JOHNSTON (2016), as atividades de controle se encarregam dos ajustes os quais permitem que os objetivos definidos no planejamento sejam atingidos, mesmo que as considerações feitas no planejamento tenham sido invalidadas. Por sua vez, o planejamento fornece as diretrizes para as atividades de controle. Enquanto o planejamento trata de ações a serem executadas no futuro, seja em anos ou minutos, o controle lida com o momento corrente, realizando o monitoramento dos processos e correção de desvios.

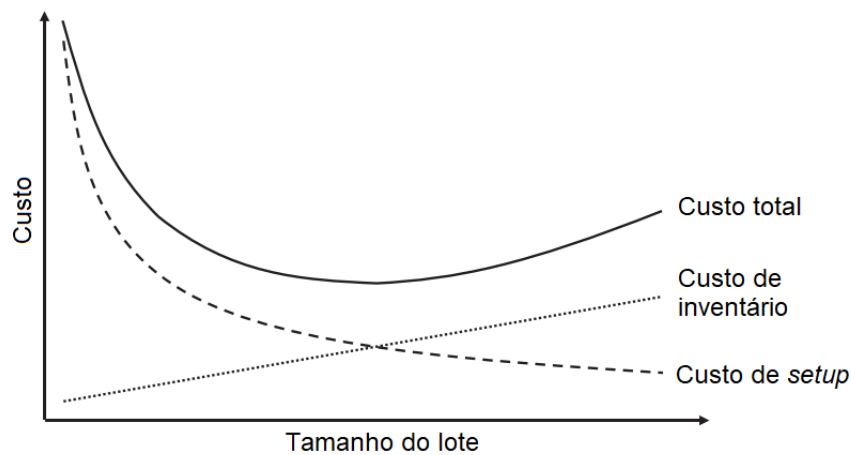
Há na literatura diversas publicações a respeito do planejamento e controle da produção na indústria de papel e celulose. No trabalho de SANTIAGO *et al.* (2019), os autores apresentam os resultados da implementação de um sistema de controle no processo de picagem de cavacos em uma indústria de celulose. O resultado obtido foi o aumento na produção de cavacos, possibilitando maior fabricação de celulose. Em FERREIRA *et al.* (2020), os autores tratam da produção de papel corrugado, e a decisão estudada é a alocação de bobinas no corrugador, de maneira a atender à demanda. O objetivo do trabalho consiste em minimizar o número de bobinas utilizadas, consumo parcial de bobinas, resíduos não utilizáveis e paradas do corrugador.

Este trabalho consiste em um problema de sequenciamento com alguns aspectos de dimensionamento de lotes. Na sequência, serão discutidas essas duas atividades, pertencentes ao planejamento e controle da produção. Primeiro, esses tópicos serão apresentados de maneira separada, destacando aspectos intrínsecos a cada um deles e exemplos de formulações. Posteriormente, será discutida a integração entre essas duas atividades.

### **2.2.1. Dimensionamento de lotes**

De acordo com GLOCK, GROSSE e RIES (2014), o dimensionamento de lotes possui como objetivo determinar o tamanho dos lotes em um ambiente produtivo, levando em consideração os custos de inventário e de *setup*. Uma representação simplificada é dada na Figura 5. Caso a produção ocorra em numerosos lotes pequenos, os custos de inventário são baixos, pois os estoques são reduzidos, mas os custos de *setup* são mais elevados, pois é necessário configurar as máquinas diversas vezes. Se a produção ocorrer em lotes maiores, os custos de *setup* decrescem, mas o custo de inventário é elevado.

Existe uma série de classificações para os problemas de dimensionamento de lotes. Um dos trabalhos de revisão mais relevantes neste campo de estudo foi realizado por BAHL, RITZMAN e GUPTA (1987). Nele, os autores dividem os problemas de acordo com o número de estágios e se há ou não restrição quanto aos recursos. O problema é monoestágio se a demanda do produto em questão independe das decisões de dimensionamento para outros produtos, sendo esta situação também chamada de demanda independente. Isso ocorre, por exemplo, quando a empresa se encarrega apenas de fabricar o produto final, a partir de insumos



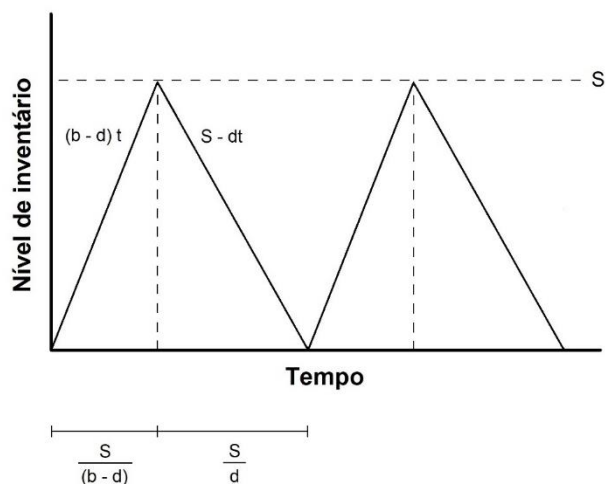
**Figura 5** - Balanço de custos no dimensionamento de lotes. Fonte: Adaptada de GLOCK, GROSSE e RIES (2014).

adquiridos externamente. Por sua vez, o problema é multiestágio ou dependente se a fabricação do produto em questão depende da produção de intermediários na mesma empresa.

Em um trabalho de revisão mais recente, escrito por GLOCK, GROSSE e RIES (2014), os autores dividem os problemas de dimensionamento clássicos entre dois estágios, multiestágio e integrados. O problema de dois estágios é semelhante ao problema monoestágio de BAHL, RITZMAN e GUPTA (1987), mas os autores classificam a distribuição do produto como um dos estágios. Os problemas integrados são aqueles em que as decisões a respeito do dimensionamento de lotes são tomadas no nível da *supply chain*, em vez de apenas em uma unidade fabril. Nesse trabalho, também ressalta-se que os problemas podem ser divididos de acordo com o número de itens produzidos na indústria: um ou múltiplos. Se são múltiplos itens, há competição entre as máquinas, o que influencia no sequenciamento. O livro de POCHEE e WOLSEY (2006) trata do planejamento de produção de maneira mais abrangente e apresenta capítulos separados para problemas de um item e de múltiplos itens, abordando o dimensionamento de lotes em conjunto com outras etapas de otimização.

Um dos problemas mais conhecidos envolvendo dimensionamento de lotes é o chamado problema dos lotes econômicos. Ele consiste em definir o nível ótimo de inventário  $S$ , balanceando os custos de estoque e de liberação de ordem. Essa formulação conta com dois momentos: o carregamento e o consumo dos estoques. Na Figura 6, há um exemplo do perfil do nível de estoque para a situação em que as taxas de carregamento e de consumo são constantes, respectivamente dadas por  $b$  e  $d$ .  $S$  corresponde ao nível máximo de estoque, e  $t$  corresponde ao tempo transcorrido na etapa vigente.





**Figura 6** - Modelagem do estoque para carregamento e consumo a taxas constantes. Fonte: Elaborada pela autora.

O problema de lotes econômicos é cíclico, de estágio único – pois apenas uma operação é considerada – e compreende um período infinito de tempo. Com essa formulação, o problema possui solução analítica. Aprimoramentos foram feitos ao longo dos anos, e um deles é o problema de dimensionamento de lotes com restrição de capacidade (*CLSP – capacitated lot-sizing problem*), proposto em MANNE (1958), em que são produzidos múltiplos produtos em uma única máquina – o que justifica a restrição de capacidade –, e o horizonte de tempo, desta vez, é finito. MELEGA, ARAÚJO e JANS (2018) realizaram uma revisão e propuseram uma classificação para trabalhos sobre o problema integrado de *CLSP* e de corte. O corte corresponde à divisão de bobinas em unidades de menor tamanho, de acordo com a demanda dos clientes. O problema de corte consiste em decidir os padrões a serem utilizados, de maneira a reduzir a geração de aparas. Ele está fortemente presente nas indústrias têxteis, metalúrgicas e de papel, dentre outras, e as decisões a respeito do dimensionamento de lotes costumeiramente dependem desse problema.

### 2.2.2. Sequenciamento

O artigo de revisão escrito por HARJUNKOSKI *et al.* (2014) é um trabalho detalhado a respeito dos problemas de sequenciamento, trazendo informações sobre métodos de resolução e sobre aplicações bem sucedidas do sequenciamento à realidade industrial. Nesse artigo, os autores definem o sequenciamento por meio das decisões de onde realizar as tarefas, em que sequência e em qual tempo. No trabalho, são discutidas as diferenças de modelagem para

indústrias com processos contínuos ou em batelada, os métodos de solução – exatos ou heurísticos –, as diferentes maneiras para a modelagem do tempo, as possíveis representações para os recursos e os processos da planta, dentre outros aspectos interessantes para o sequenciamento de unidades industriais.

Uma unidade fundamental no sequenciamento são as máquinas, definidas como elementos que realizam tarefas. Convencionou-se utilizar a letra  $n$  para o número de *jobs* e a letra  $m$  para o número de máquinas. Neste contexto, *job* pode ser entendido como um produto a ser entregue. Cada passagem de um produto por uma máquina corresponde à execução de uma tarefa (*task*). O subscrito  $i$  é utilizado para se referir a uma máquina, e o subscrito  $j$  é utilizado para se referir a um *job* (PINEDO, 2016).

Existe uma variedade de classificações para os ambientes industriais conforme a disposição das máquinas, e o entendimento da dinâmica é crucial para a formulação adequada dos problemas de sequenciamento. Na Tabela 1, estão elencadas algumas definições importantes, de acordo com PINEDO (2016).

**Tabela 1** - Definições aplicadas aos ambientes de máquinas

<b>Terminologia</b>	<b>Definição</b>
Máquina simples	Situação mais elementar, caso particular de todos os problemas mais complexos.
Máquinas paralelas	Máquinas que realizam a mesma tarefa, e um produto precisa passar por apenas uma delas. São idênticas se cada tarefa possui o mesmo tempo de execução em todas as máquinas e não-relacionadas se os tempos são diferentes.
<i>Flow shop</i>	Ambiente em que todos os produtos devem passar pelas mesmas máquinas, em série, na mesma ordem. Se o ambiente é do tipo <i>FIFO</i> ( <i>first in, first out</i> – primeiro a entrar, primeiro a sair), ele é chamado de <i>flow shop</i> permutacional.
<i>Flow shop</i> flexível	Ambiente semelhante ao <i>flow shop</i> , mas, em vez de máquinas em série, os produtos devem passar por estágios. Cada estágio conta com um número de máquinas paralelas idênticas.
<i>Job shop</i>	Cada produto possui uma sequência de máquinas pelas quais deve passar. Existem ambientes em que cada produto passa apenas uma vez por cada máquina e ambientes em que uma máquina pode ser visitada mais de uma vez pelo mesmo produto.
<i>Job shop</i> flexível	Ambiente semelhante ao <i>job shop</i> , mas, em vez de máquinas em série, cada produto possui sua própria sequência de estágios, e cada um dos estágios conta com um número de máquinas paralelas idênticas.
<i>Open shop</i>	A sequência em que cada produto percorre as máquinas é livre.

Tipicamente, cada máquina processa um *job* por vez e possui uma capacidade finita de processamento, ressaltando a importância do dimensionamento de lotes. Pode ou não existir um tempo de ajuste entre o processamento de dois *jobs* distintos, chamado tempo de *setup*. Caso o tempo de *setup* seja o mesmo para todas as trocas entre um *job* e outro, diz-se que o tempo de *setup* é independente da sequência. Se, por sua vez, esse tempo for diferente para cada par de *jobs*, o tempo de *setup* é chamado de dependente da sequência.

Em ALFARO-FERNÁNDEZ *et al.* (2020), os autores visam testar a formulação automática de algoritmos de resolução. O ambiente de produção é o *flow shop* flexível, chamado de *flow shop* híbrido no artigo. Nesse trabalho, três objetivos distintos são testados: a minimização do tempo em que o último *job* é finalizado (*makespan*), a minimização da soma dos tempos de processamento dos jobs (*total flowtime*) e a minimização do tempo total de adiantamento e atraso. Em GUIMARÃES *et al.* (2019), o ambiente de produção é um *flow shop* com duas semi-linhas paralelas e uma operação final de sincronização. O objetivo consiste em minimizar o *makespan*, utilizando as meta-heurísticas GRASP e recozimento simulado (*simulated annealing*). Em ZHANG *et al.* (2020), o ambiente proposto é um *job shop* para a fabricação de diferentes válvulas. Nesse trabalho, um problema multiobjetivo é proposto, com os objetivos de minimizar o *makespan*, o atraso total e a carga de trabalho (*workload*) total, utilizando a meta-heurística da colônia de formigas (*ant colony optimization*).

Segundo HARJUNKOSKI *et al.* (2014), problemas de sequenciamento na indústria de papel são particularmente desafiadores por conta da necessidade de resolução simultânea do problema de corte – citado anteriormente –, minimizando a perda de aparas e atendendo às demandas dos clientes. Em LE HESRAN *et al.* (2019), é realizada uma revisão sobre trabalhos de sequenciamento visando a minimização da geração de rejeitos, e um dos temas citados é o problema de corte. Os autores mencionam que o uso apenas dos padrões de corte mais eficientes tipicamente demanda um número mais elevado de trocas, elevando o tempo e o custo de *setup*. Sendo assim, é necessário balancear os custos operacionais e os custos decorrentes da perda de material.

O problema do caixeiro viajante, com ou sem coleta de prêmios, pode ser utilizado para problemas de sequenciamento com *setup* dependente da sequência, em que as trocas entre dois produtos são entendidas como o caminho entre duas cidades. HORN *et al.* (2021) propuseram um método para a construção de diagramas de decisão, visando a resolução de problemas de sequenciamento com coleta de prêmios, em que um prêmio mínimo deve ser coletado por meio da execução de um número de tarefas. CHARITOPoulos, DUA e PAPAGEORGIU (2017) utilizaram o problema do caixeiro viajante para programar a produção em um problema integrado de planejamento, sequenciamento e controle em processos contínuos.

### 2.2.3. Problema integrado de dimensionamento e sequenciamento de lotes

A resolução integrada de problemas de otimização tem se tornado cada vez mais comum, em função da disponibilidade de computadores com maior capacidade de processamento. Um exemplo são os problemas de dimensionamento e sequenciamento de lotes.

Um dos trabalhos mais citados nesse campo de estudo, com 414 citações em artigos publicados em periódicos até março de 2021 de acordo com o portal Scopus, é o artigo de revisão publicado por DREXL e KIMMS (1997). Neste artigo, os autores organizam as variadas formulações para problemas integrados de dimensionamento e sequenciamento de acordo com sua complexidade e fornecem mais de uma centena de referências sobre o assunto. Mais recentemente, COPIL *et al.* (2017) realizaram uma revisão bibliográfica sobre trabalhos com esta temática e propuseram uma formulação geral que pode ser adaptada aos modelos.

Um dos modelos visitados por DREXL e KIMMS (1997) é o chamado problema de sequenciamento de lotes econômicos (*ELSP – economic lot sizing and scheduling problem*), proposto por ROGERS (1958) e entendido por GLOCK, GROSSE e RIES (2014) como uma extensão do problema de dimensionamento de lotes, também de horizonte infinito de planejamento, estágio único e máquina única. Caso os ciclos possuam a mesma duração para todos os produtos, o problema de lotes econômicos possui solução analítica. Se o *setup* entre os produtos for dependente da sequência, é necessária a resolução de uma variante do problema do caixeiro viajante para a determinação do menor tempo total de *setup* (LOPEZ e KINGSMAN, 1991). No entanto, a obrigatoriedade de ter ciclos da mesma duração usualmente não leva à solução ótima. O problema com ciclos de durações distintas não possui solução analítica e é classificado como NP-difícil (DREXL; KIMMS, 1997). No entanto, ele leva, na maioria das vezes, a soluções melhores que o problema com ciclos de mesmo tamanho, pois este está contido no problema geral (GLOCK; GROSSE; RIES, 2014).

O trabalho de GLOCK (2012) consiste em uma revisão do problema conjunto de dimensionamento de lotes (*joint economic lot sizing problem*, em inglês), o qual trata do problema de lotes econômicos aplicado a uma *supply chain*, em vez de a uma única empresa. Em alguns trabalhos visitados nesta revisão, ocorre a fabricação de múltiplos produtos em uma única máquina, o que faz com que decisões de sequenciamento devam ser tomadas.

Como exemplos de trabalhos de pesquisa para problemas desta natureza, KIRSCHSTEIN (2018) propôs um método heurístico para a resolução do problema de sequenciamento de lotes econômicos para transporte em tubulações, situação existente em

indústrias químicas e petrolíferas. Em problemas dessa natureza, o *setup* é dependente da sequência, pois a extensão da interface depende dos produtos que se sucedem. Sendo assim, esse problema é classificado como de dimensionamento e sequenciamento de lotes econômicos dependentes da sequência (*sELSP – sequence-dependent economic lot scheduling problem*). Na heurística proposta, os produtos mais frequentes são alocados primeiro, e os restantes são adicionados posteriormente. O encaixe dos produtos é feito gradativamente por meio do problema do caixeiro viajante.

Ainda em lotes econômicos, ZOHALI *et al.* (2019) tratam de um ambiente *flow shop* híbrido, em que o objetivo consiste em minimizar custos de inventário e de *setup*. Os autores aprimoraram um modelo já existente de programação não-linear inteira mista e propuseram um método de linearização para transformar o modelo existente e o criado por eles em problemas de programação linear inteira mista. Os modelos lineares foram eficientemente empregados para resolver instâncias pequenas e médias. Para as instâncias maiores, foi proposto um algoritmo de busca local iterada híbrida.

A respeito do problema integrado de dimensionamento e sequenciamento de lotes com *setup* dependente da sequência, GUIMARÃES, KLABJAN e ALMADA-LOBO (2014) propuseram um método de classificação para abordagens desta natureza e destacaram o problema do caixeiro viajante assimétrico como uma fonte de ideias para o desenvolvimento de métodos e modelos mais eficientes.

Ao longo dos anos, o *ELSP* sofreu uma série de adaptações a fim de contemplar instâncias mais realistas. De modo geral, as formulações atualmente utilizadas contam com horizonte finito de tempo, o qual é dividido em intervalos menores chamados de microperíodos. A cada microperíodo, há a fabricação de apenas um produto ou há apenas um *setup*, a depender da metodologia adotada. Em modelos *small-bucket*, o inventário dos produtos é verificado a cada microperíodo. Por sua vez, nos modelos *large-bucket*, o inventário é verificado a cada macroperíodo, que é um conjunto de microperíodos, e o número de microperíodos em cada macroperíodo é um parâmetro a ser definido.

Existem variados modelos para a resolução dos problemas de dimensionamento e sequenciamento de lotes, e o mais utilizado é o problema geral de dimensionamento e sequenciamento de lotes (*GLSP – general lot sizing and scheduling problem*), por ser uma generalização de outros modelos existentes. A Tabela 2 mostra características de alguns dos modelos, bem como as referências em que foram publicados.

**Tabela 2** - Modelos para o problema integrado de dimensionamento e sequenciamento de lotes

<b>Categoria</b>	<b>Modelo</b>	<b>Máximo de itens por período</b>	<b>Trocas por período</b>	<b>Observações</b>
<i>Small-bucket</i>	<i>DLSP</i> <sup>1</sup>	1	0	Tudo ou nada: ou se produz durante o período todo, ou nada é produzido.
	<i>CSLP</i> <sup>2</sup>	1	1	A máquina fica ociosa se o produto não ocupar todo o tempo.
	<i>PLSP</i> <sup>3</sup>	2	1	Preenche o tempo ocioso deixado no <i>CSLP</i> .
<i>Large-bucket</i>	<i>GLSP</i> <sup>4</sup>	Definido pelo operador	Definido pelo operador	Formulação genérica, que pode ser transformada nas demais.
	<i>CLSD</i> <sup>5</sup>	Definido pelo operador	Definido pelo operador	Modelagem feita com macroperíodos, em que pode haver troca de configuração dentro do microperíodo.

<sup>1</sup>Problema discreto de dimensionamento e sequenciamento de lotes (*Discrete lot sizing and scheduling problem*) - (FLEISCHMANN, 1990)

<sup>2</sup>Problema de dimensionamento de lotes com *setup* contínuo (*Continuous setup lot sizing problem*) - (KARMAKAR e SCHRAGE, 1985)

<sup>3</sup>Problema proporcional de dimensionamento e sequenciamento de lotes (*Proportional lot sizing and scheduling problem*) - (DREXL e HAASE, 1995)

<sup>4</sup>Problema geral de dimensionamento e sequenciamento de lotes (*General lot sizing and scheduling problem*) - (FLEISCHMANN e MEYR, 1997)

<sup>5</sup>Problema capacitado de dimensionamento de lotes com *setup* dependente da sequência (*Capacitated lot sizing problem with sequence-dependent setups*) - (HAASE, 1996)

No trabalho de WICHMANN, JOHANNES e SPENGLER (2019), uma formulação chamada de problema geral de dimensionamento e sequenciamento de lotes orientado à energia (*EOGLSP – energy-oriented general lot sizing and scheduling problem*) é utilizada para realizar a otimização em uma situação com armazenamento de energia. O trabalho parte da premissa de que o preço da energia está sujeito a diversas flutuações, então seu armazenamento permitiria um maior controle sobre seu preço.

Como exemplos de empregos da formulação *GLSP* nas indústrias de produtos derivados da celulose, o trabalho de MARTÍNEZ, MORABITO e TOSO (2018) trata do dimensionamento e sequenciamento integrados na fabricação de polpa moldada, que são embalagens descartáveis para uma série de insumos. A função objetivo utilizada neste trabalho foi a minimização dos custos totais de *setup*, inventário e *backlog*. O trabalho de FIGUEIRA *et al.* (2015) propõe o dimensionamento de lotes e sequenciamento na produção de papéis de diferentes gramaturas e composições. O objetivo do trabalho consistiu em minimizar os gastos, considerando *setup*, inventário e *backlog*. Nesse caso, a arrecadação foi incluída na função objetivo como um gasto negativo. Uma planta integrada de celulose e papel é objeto de estudo

em SANTOS e ALMADA-LOBO (2012). Nesse trabalho, a geração de energia a partir do licor negro é considerada nas restrições. O objetivo consiste em minimizar a diferença entre custo e arrecadação, utilizando o método *relax-and-fix*, em que algumas das variáveis inteiras são relaxadas a cada iteração.

Em FURLAN *et al.* (2015), não houve a categorização do problema nos modelos apresentados acima. No trabalho, uma planta integrada de celulose e papel com múltiplas máquinas de papel é estudada, e propõe-se a resolução do dimensionamento e sequenciamento, utilizando-se a programação inteira mista e a meta-heurística do algoritmo genético.

### 3. REFERENCIAL TEÓRICO

#### 3.1. PROBLEMA DO CAIXEIRO VIAJANTE

O problema do caixeiro viajante (PCV) – em inglês, *traveling salesman problem (TSP)* – é um dos problemas clássicos em Pesquisa Operacional e pode ser aplicado a uma ampla gama de situações na área de sequenciamento. O problema recebe este nome porque pode ser ilustrado por meio da situação em que um caixeiro deve visitar um conjunto de cidades, partindo de uma cidade de origem e retornando a ela no final, percorrendo o caminho de menor custo.

Este problema pode ser explicado por meio da Teoria dos Grafos. Um grafo  $G = (V, A)$  consiste em um conjunto  $V = \{1, 2, \dots, n\}$  finito e não-vazio de vértices e em um conjunto  $A = \{(i, j) : i, j \in V\}$  de arcos. O grafo pode ser orientado ou não-orientado. Ele é não-orientado se o conjunto  $A$  é formado por pares não ordenados de  $V$ , e orientado em caso contrário. A cada arco  $(i, j)$  está associado um peso  $c_{ij}$ . No grafo não-orientado, costuma-se denominar os elementos de  $A$  por arestas. Outro conceito importante para o problema em questão é o caminho Hamiltoniano, definido como um percurso  $G^* = (V, A^*)$  dentro do grafo  $G$  em que todos os vértices são visitados exatamente uma vez. O peso total  $c^*$  associado a esse caminho é dado pela Equação (3.1):

$$c^* = \sum_{(i,j) \in A^*} c_{ij} \quad (3.1)$$

O PCV consiste em encontrar o caminho Hamiltoniano com o menor peso total. O problema pode ser simétrico ou assimétrico, sendo simétrico quando o grafo é não-orientado e assimétrico quando o grafo é orientado.

Na modelagem matemática do problema, utilizam-se variáveis binárias para definir se um arco  $(i, j)$  pertence ou não à rota ótima. A função objetivo do problema do caixeiro-viajante é dada pela Equação (3.2).

$$\text{Minimizar} \quad \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (3.2)$$

As variáveis  $x_{ij}$  são variáveis de decisão, as quais valem 1, se o arco de  $i$  até  $j$  é percorrido, e 0, caso contrário. O somatório fornece o custo total da rota.

As restrições são dadas pelas Equações (3.3), (3.4) e (3.5), em conjunto com equações para a eliminação de sub-rotas, esclarecidas mais adiante.



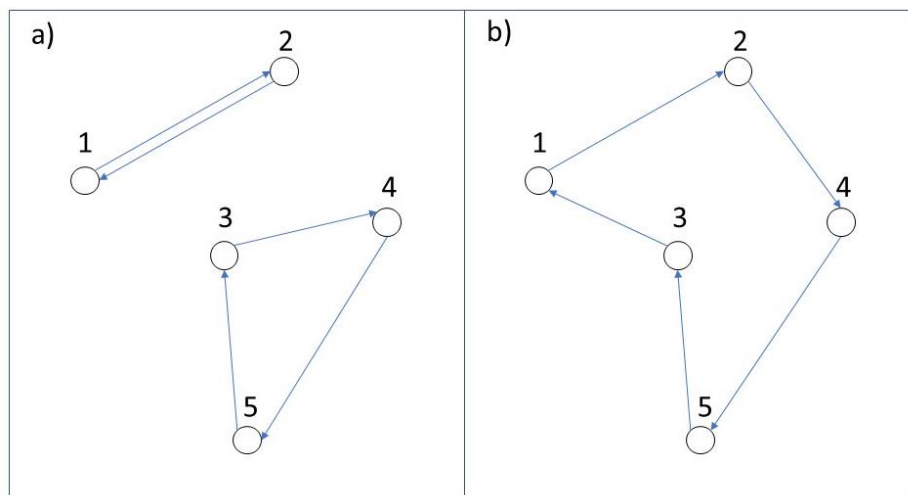
$$\sum_{i=1, i \neq j}^n x_{ij} = 1; \quad j = 1..n \quad (3.3)$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1; \quad i = 1..n \quad (3.4)$$

$$x_{ij} \in \{0,1\}; \quad i, j = 1..n \quad (3.5)$$

As restrições (3.3) impõem que cada vértice funciona como chegada apenas uma vez, e as restrições (3.4), de maneira similar, impõem que cada vértice funciona como partida apenas uma vez. As restrições (3.5) fornecem o domínio das variáveis  $x_{ij}$ .

Além destas três restrições, devem ser incluídas as formulações para se evitar a formação de sub-rotas. Sub-rotas são percursos em que não se visitam todos os vértices do grafo, e portanto não correspondem a caminhos Hamiltonianos, devendo ser descartados. A Figura 7 ilustra essa situação para um grafo contendo 5 vértices. A Figura 7 (a) mostra a formação das sub-rotas  $1 \rightarrow 2 \rightarrow 1$  e  $3 \rightarrow 4 \rightarrow 5 \rightarrow 3$ , resultado não desejado para o problema do caixeiro viajante. Em 7 (b), é mostrada a rota  $1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 1$ , considerada válida para este problema, por percorrer todos os vértices.



**Figura 7** - Exemplo de problema do caixeiro viajante com 5 vértices: (a) percurso com sub-rotas e (b) percurso sem sub-rotas. Fonte: elaborada pela autora.

O primeiro modelo proposto na literatura para eliminação de sub-rotas foi apresentado em DANTZIG, FULKERSON e JOHNSON (1954) – DFJ – e ele demanda um número exponencial de restrições para eliminação de sub-rotas. Posteriormente, modelos com número polinomial de restrições foram propostos. O primeiro deles foi enunciado por MILLER,

TUCKER e ZEMLIN (1960) – MTZ. Em 1978, GAVISH e GRAVES propuseram um modelo – GG – baseado em *single-commodity flow*, com relaxação linear mais forte que o MTZ e mais fraca que o DFJ. CLAUS (1984) propôs uma formulação de *multi-commodity flow* com  $n - 1$  *commodities*. Esta formulação possui relaxação linear equivalente à de DFJ.

Na sequência, ilustra-se o modelo de GAVISH e GRAVES (1978), que, como o utilizado neste trabalho, baseia-se em *single-commodity flow*. Nele, as variáveis  $g_{ij} \in \mathbb{R}_+$  informam o número de arcos do vértice 1, definido como raiz, até o arco  $(i, j)$ . O modelo compreende as Equações (3.2) a (3.5) e as Equações (3.6) e (3.7):

$$\sum_{j=1}^n g_{ij} - \sum_{j=2}^n g_{ji} = 1; \quad i = 2..n \quad (3.6)$$

$$0 \leq g_{ij} \leq (n - 1)x_{ij}; \quad i = 2..n, j = 1..n \quad (3.7)$$

Em NIQUINI *et al.* (2020), os três modelos polinomiais citados foram utilizados para resolver um problema preliminar ao apresentado neste trabalho, em que o tempo total de *setup* foi minimizado por meio da alteração na ordem dos lotes de uma sequência fornecida pela indústria. As formulações para diversos modelos de eliminação de sub-rotas, inclusive os citados, podem ser vistas em ROBERTI e TOTH (2012).

### 3.2. PROBLEMA DO CAIXEIRO VIAJANTE COM COLETA DE PRÊMIOS

O problema do caixeiro viajante com coleta de prêmios (PCVCP) – em inglês, *prize collecting traveling salesman problem (PCTSP)* – corresponde a uma variação do problema do caixeiro viajante em que há um prêmio para cada vértice visitado e uma punição para cada vértice não visitado. Como definido por BALAS (1989), o objetivo consiste em definir o caminho de menor custo – levando em conta o custo de cada arco e as punições pelos vértices não visitados – para se atingir uma dada arrecadação. A função objetivo é dada como descrito em (3.8).

$$\text{Minimizar} \quad \sum_{i=1}^n \sum_{j=1}^n c_{ij}x_{ij} + \sum_{j=1}^n p_j(1 - y_j) \quad (3.8)$$

A primeira parcela da soma é idêntica à do problema do caixeiro viajante original, com  $c_{ij}$  correspondendo ao custo relativo ao arco  $(i, j)$ , e  $x_{ij} \in \{0,1\}$  indicando se o arco é ou não percorrido. A segunda parcela traz  $p_j$ , que corresponde ao valor da punição caso o ponto  $j$  não seja visitado, e  $y_j \in \{0,1\}$ , que vale 1 se o ponto  $j$  é visitado e 0 caso contrário.

As restrições são dadas por (3.9), (3.10), (3.11) e (3.12), além das restrições de sub-rotas. O conjunto de restrições (3.9) diz que, se um ponto  $j$  é visitado ( $y_j = 1$ ), ele serve como chegada exatamente uma vez. Caso ele não seja visitado ( $y_j = 0$ ), ele não serve como chegada. As restrições (3.10) funcionam de maneira análoga, porém tratando do número de partidas. A restrição (3.11) define que a soma dos prêmios  $w_i$  coletados em cada ponto visitado deve ser superior a um valor  $W_0$  predefinido. As restrições (3.12) fornecem o domínio de  $x_{ij}$  e  $y_j$ .

$$\sum_{i=1, i \neq j}^n x_{ij} - y_j = 0; \quad j = 1..n \quad (3.9)$$

$$\sum_{j=1, j \neq i}^n x_{ij} - y_i = 0; \quad i = 1..n \quad (3.10)$$

$$\sum_{j=1}^n w_j y_j \geq W_0 \quad (3.11)$$

$$x_{ij}, y_j \in \{0,1\}; \quad i, j = 1..n \quad (3.12)$$

Como exposto anteriormente, existe uma série de modelos para a eliminação de sub-rotas no problema do caixeiro viajante, os quais podem ser adaptados para o problema com prêmios. Neste trabalho, foi utilizado um modelo de *single commodity*, proposto por PANTUZA JUNIOR e SOUZA (2020), descrito pelas restrições de (3.13) a (3.16).

$$\sum_{j \in V \setminus \{i\}} (f_{ij} - w_i x_{ij}) - \sum_{j \in V \setminus \{r, i\}} f_{ji} = 0; \quad \forall i \in V \setminus \{r\} \quad (3.13)$$

$$\sum_{i \in V \setminus \{r\}} f_{ir} \geq W_0 \quad (3.14)$$

$$f_{ij} \leq \bar{W} x_{ij}; \quad \forall i, j \in V, i \neq j, i \neq r \quad (3.15)$$

$$f_{ij} \geq 0; \quad \forall i, j \in V, i \neq j, i \neq r \quad (3.16)$$

O valor  $f_{ij}$  é definido como o fluxo da *commodity* pelo arco  $(i, j)$ , deixando o vértice  $i$  e chegando ao vértice  $j$ . O vértice  $r$  é definido como a raiz do fluxo de *commodity*. As restrições (3.13) fazem o balanço do fluxo da *commodity* em cada um dos vértices  $i$ , tomando em consideração o prêmio por visitar esses vértices. A restrição (3.14) diz que a soma dos fluxos de *commodity* chegando ao nó raiz deve ser maior ou igual ao prêmio mínimo a ser coletado. As restrições (3.15), em que  $\bar{W} = \sum_{i \in V \setminus \{r\}} w_i$ , definem a ordem de grandeza das variáveis  $f_{ij}$  e as atrelam às variáveis  $x_{ij}$  do problema do caixeiro viajante: se  $x_{ij}$  é nula,  $f_{ij} \leq 0$ . As restrições (3.16) definem o domínio de  $f_{ij}$ .

### 3.3. HEURÍSTICAS DE CONSTRUÇÃO

Os métodos heurísticos possuem como objetivo produzir boas soluções para problemas de otimização em um tempo menor que o demandado para a obtenção da solução exata. Os procedimentos adotados para a resolução de um problema de otimização por métodos heurísticos devem ser adequados à natureza do problema e, se implementados de maneira apropriada, são capazes de fornecer soluções satisfatórias. Para problemas com número elevado de variáveis inteiras, métodos exatos são proibitivos em função do tempo computacional, e as heurísticas são amplamente utilizadas. Uma das heurísticas mais antigas para o problema do caixeiro viajante, descrito na [Seção 3.1](#), é o algoritmo do vizinho mais próximo. Nele, escolhe-se um vértice como referência, e acrescenta-se ao percurso o arco de menor valor incremental que una o vértice escolhido a um dos vértices ainda não visitados. Feito isso, o vértice de referência é atualizado para o último adicionado ao percurso. O procedimento é repetido até que todos os vértices tenham sido conectados. O algoritmo do vizinho mais próximo é considerado uma heurística de construção, pois parte-se de uma solução vazia, e esta é incrementada até que uma solução completa seja obtida. Outra modalidade de heurística são os procedimentos de busca. Nestes, parte-se de uma solução completa, e são realizadas tentativas de aprimorá-la por meio de buscas em seu entorno.

Dentro dos métodos heurísticos, existem as meta-heurísticas. Segundo GENDREAU e POTVIN (2019), meta-heurísticas são métodos de resolução que promovem a interação entre procedimentos locais de busca e estratégias de alto nível para escapar de ótimos locais e promover uma busca robusta em um espaço de soluções.

Os métodos multipartida (em inglês, *multi-start methods*) consistem em uma categoria de meta-heurísticas. Esses métodos contam com duas fases: a fase construtiva e a fase de melhoria. Na fase construtiva, são geradas soluções viáveis para o problema. Na fase de melhoria, são realizadas buscas em torno dessas soluções, de modo a aprimorar os resultados obtidos. O algoritmo retorna a melhor solução encontrada a partir da busca em torno de todas as soluções iniciais viáveis. Em problemas de sequenciamento, os métodos multipartida são uma opção interessante, uma vez que os espaços de busca são criados em torno de soluções viáveis (AVALOS-ROSALES; ANGEL-BELLO; ALVAREZ, 2015).

A meta-heurística mais conhecida para o método multipartida é a chamada GRASP (*Greedy Randomized Adaptive Search Procedure*), proposta por FEO e RESENDE (1989). Ela conta com uma fase de construção chamada *Greedy Randomized Adaptive* (gulosa, aleatorizada e adaptativa) e com uma fase de melhoria em que é realizada uma busca local. O pseudocódigo para a meta-heurística é dado na Figura 8. Em cada iteração, uma solução é gerada, e o número máximo de iterações é dado por `Max_Iteracoes`. A entrada `Semente` diz respeito ao número que serve de partida para a geração de números pseudoaleatórios, utilizados na fase aleatorizada da etapa de construção. Caso a solução gerada na etapa de construção não seja viável, ela deve ser ajustada de modo a se tornar viável. Encontrada uma solução inicial viável, é realizado um procedimento de busca local, em que o objetivo consiste em encontrar um ponto ótimo na vizinhança da solução gerada. Feitas todas as iterações, o algoritmo retorna a melhor solução encontrada.

```

Função Melhor_Solucao = GRASP(Max_Iteracoes, Semente)
  Para k = 1, ..., Max_Iteracoes
    Solucao ← Construção_Gulosa_Aleatorizada(Semente)
    Se Solucao não é viável
      Solucao ← Reparar(Solucao)
    Fim Se
    Solucao ← Busca_Local(Solucao)
    Atualizar_Solucao(Solucao, Melhor_Solucao)
  Fim Para
  Retorna Melhor_Solucao
Fim Função

```

**Figura 8** - Pseudocódigo para a meta-heurística GRASP. Fonte: adaptada de RESENDE e RIBEIRO (2019).

Na fase de construção, cujo pseudocódigo é dado na Figura 9, tem-se como ponto de partida um conjunto-solução vazio. É criada uma lista restrita de candidatos (LRC) – em inglês, *restricted candidate list (RCL)* –, contendo os elementos que possuam os menores, se minimização, ou maiores, se maximização, valores incrementais para a função objetivo. A criação desta lista corresponde à etapa gulosa (ou *greedy*) do algoritmo, podendo o método para

a construção ser cardinalidade ou qualidade. No caso da cardinalidade, são escolhidos os  $p$  melhores elementos dentre os candidatos, em que  $p$  é um número inteiro definido pelo usuário. Caso o método seja a qualidade, é primeiro estabelecido um valor de corte, com base nos valores mínimo e máximo do custo incremental, e a lista é construída com todos os valores abaixo deste valor de corte, se minimização, ou acima, se maximização. Para o critério de qualidade, é necessário um parâmetro  $\alpha \in [0,1]$ , o qual definirá a proporção entre gulosidade e aleatoriedade da lista. Em problemas de minimização, em que o menor e o maior custo incrementais são respectivamente  $c^{min}$  e  $c^{max}$ , o valor de corte é dado por  $c^{min} + \alpha(c^{max} - c^{min})$ . Caso  $\alpha = 0$ , apenas  $c^{min}$  é incluído na LRC, o que corresponde a uma construção completamente gulosa. Por outro lado, se  $\alpha = 1$ , todos os elementos são colocados na LRC, tratando-se de uma lista completamente aleatorizada.

Formada esta lista, é realizado um sorteio dentre seus elementos, utilizando a Semente, e o elemento sorteado é adicionado à sequência. Esta etapa corresponde à parte “*randomized*” ou aleatória do algoritmo. Feito isso, o custo incremental para cada um dos elementos é atualizado, de modo que uma LRC diferente deve ser construída a cada iteração. Esta parte é a “*adaptive*” ou adaptativa. Essas etapas são repetidas até que todos os elementos candidatos sejam incluídos na sequência.

```

Função Solucao = Construcao_Gulosa_Aleatorizada(Semente)
  Solucao  $\leftarrow$   $\emptyset$ 
  Inicializar o conjunto de elementos candidatos
  Avaliar os custos incrementais de cada elemento candidato
  Enquanto existir pelo menos um elemento candidato
    Construir a lista restrita de candidatos(LRC)
    Selecionar aleatoriamente um elemento  $s$  da LRC
    Solucao  $\leftarrow$  Solucao  $\cup$   $\{s\}$ 
    Atualizar o conjunto de elementos candidatos
    Reavaliar os custos incrementais
  Fim Enquanto
  Retorna Solucao
Fim Função

```

**Figura 9** - Pseudocódigo para a etapa de construção do método GRASP. Fonte: adaptada de RESENDE e RIBEIRO (2019).

Caso a solução obtida na etapa de construção não seja viável, é necessário aplicar um procedimento para reparo da solução não-viável ou executar novamente o código até que se obtenha uma sequência viável. Feito isso, parte-se para o procedimento de busca local. A busca local parte de uma solução candidata – no caso do GRASP, a solução obtida na etapa de construção – e se desloca por sua vizinhança até que um ótimo local seja encontrado. Diversas heurísticas e meta-heurísticas de busca local podem ser utilizadas nesta etapa. Em SANTOS *et*

al. (2019), foi realizada uma análise de métodos de busca local estocástica, como recozimento simulado (*simulated annealing*) e busca local iterada (*iterated local search*), para o problema de agendamento em máquinas paralelas não-relacionadas. Em GUIMARÃES *et al.* (2019), foi resolvido o problema de *flow shop* para duas semilinhas paralelas e uma operação final de sincronização, utilizando o método GRASP e o recozimento simulado. Para o GRASP, foram utilizadas trocas de posição entre dois elementos como procedimento de busca local. Em AVALOS-ROSALES, ANGEL-BELLO e ALVAREZ (2015), foi resolvido o problema de sequenciamento de tarefas em máquinas paralelas não-relacionadas, e a meta-heurística utilizada na etapa de melhoria da solução foi a busca em vizinhança descendente (*VND – variable neighborhood descent*).

O pseudocódigo para o procedimento de busca local para problema de minimização, em que  $V$  corresponde à vizinhança da solução, é dado na Figura 10.

```

Função Solucao = Busca_Local (Solucao)
  Enquanto Solucao não for um ótimo local
    Encontrar  $s' \in V(\text{Solucao})$  com  $f(s') < f(\text{Solucao})$ 
    Solucao  $\leftarrow s'$ 
  Fim Enquanto;
  Retorna Solucao
Fim Função

```

**Figura 10** - Pseudocódigo para o procedimento de busca local no método GRASP. Fonte: adaptada de RESENDE e RIBEIRO (2019).

O algoritmo completo – construção e busca local – é repetido por `MaxIteracoes` vezes, e as soluções obtidas são distintas entre si devido à parte aleatória do algoritmo. O algoritmo retorna o melhor ótimo local obtido. Para uma explicação mais detalhada sobre o método GRASP, a autora sugere a referência original (FEO; RESENDE, 1989) e o capítulo de livro escrito por RESENDE e RIBEIRO (2019).

## 4. PROBLEMA ABORDADO

### 4.1. DEFINIÇÃO DO PROBLEMA

O problema de sequenciamento de lotes explorado neste trabalho parte de uma situação-problema presente em uma indústria de papel para embalagens localizada no Brasil. Esta indústria produz papéis-cartão destinados a caixas de bebidas longa vida, de alimentos congelados e caixas de uso geral – medicamentos, cereais, chás, dentre outros –, além de embalagens para conjuntos de cervejas.

A produção é planejada mensalmente, em conformidade com a previsão de demanda, enviada ao setor de planejamento pelo setor comercial. Há um cliente em particular que realiza retiradas frequentes de dois dos produtos, de maneira que, para estes, é necessária a manutenção de estoque. Define-se um nível mínimo para o estoque, chamado de estoque de segurança, destinado a atender aumentos imprevistos no consumo dos produtos. Para os demais produtos, os prazos dentro do mês são flexíveis, e os lotes são entregues aos seus clientes conforme finalizados. Embora não haja prazos rígidos para esses produtos, é interessante que sua produção seja distribuída ao longo de todo o mês, intercalando com os produtos de retiradas periódicas. Desta forma, o acúmulo excessivo do estoque de produtos com retiradas periódicas é evitado – o que poderia causar sobrecarga no armazenamento –, e os clientes dos outros produtos não necessariamente devem aguardar até o final do mês para que suas demandas sejam atendidas.

Duas máquinas, as quais operam em regime contínuo, são destinadas à fabricação de papel-cartão nesta indústria, e a alocação dos lotes nas máquinas é pré-determinada de acordo com as dimensões desejadas para o produto. Neste trabalho, serão tratados os produtos alocados na máquina de maior dimensão. Para a troca entre dois produtos distintos, há um período de tempo durante o qual a máquina produz refugo ou papel-cartão de qualidade inferior, devido aos ajustes na gramatura e na matéria-prima utilizada para cada categoria de papel. Esse tempo, chamado tempo de *setup*, é dependente dos elementos envolvidos na troca e da sequência de produção dos lotes: o tempo de *setup* para a troca entre o produto A e o produto B pode ser diferente do tempo entre B e A. As sequências mensais devem ser planejadas levando-se em conta o tempo total de *setup*, uma vez que a redução do mesmo implica menor geração de subprodutos e maior disponibilidade de tempo de produção, gerando benefícios dos pontos de vista ambiental e econômico.



O planejamento deve levar em conta também a dimensão dos lotes: lotes de poucas toneladas são desvantajosos em função da complexidade do processo produtivo ininterrupto em uma máquina de grande porte. Nesta indústria, o lote mínimo desejável é de 350 toneladas, porém exceções são feitas quando a demanda mensal de um dado produto é inferior a esse valor. Do outro lado, lotes demasiado extensos são inviáveis devido à possibilidade de esgotamento da polpa de celulose e da geração de gargalos nas rebobinadeiras (ver [Figura 3](#), Capítulo 2). A dimensão máxima dos lotes varia conforme a classe do produto, sendo a maior delas igual a 5000 toneladas. Os valores efetivamente utilizados neste trabalho serão dados no [Capítulo 5](#): Estudo computacional. A decisão a respeito da dimensão dos lotes impacta diretamente no tempo total de *setup*, pois, primeiramente, quanto menor o número de lotes, menor será o número de trocas necessárias. Em segundo lugar, o estoque dos produtos de retiradas frequentes deve ser observado a todo tempo, para indicar a necessidade de fabricação deles em um determinado instante. Embora haja essas ponderações a respeito do dimensionamento de lotes, ele não foi considerado na função objetivo, pois os custos de *setup* da indústria são significativamente maiores que os custos de armazenamento.

Trocas entre determinados pares de produtos são indesejadas, devido à elevada diferença de gramatura e matéria-prima. Mesmo que haja um tempo de *setup* informado para essas trocas, na prática elas não ocorrem e devem ser vetadas no planejamento, porque há elevada possibilidade de que o segundo produto apresente imperfeições estruturais. As trocas não permitidas serão detalhadas no [Capítulo 5](#).

Nas instâncias consideradas neste trabalho, foram utilizados os produtos dados na Tabela 3. Os produtos K205 e K274 são os adquiridos pelo cliente que realiza retiradas frequentes, embora nem toda a produção desses dois itens seja destinada a esse cliente em particular. Neste trabalho, as retiradas frequentes foram tratadas como contínuas, e a taxa de retirada de cada um deles é dada em toneladas por hora. Daqui por diante, essa demanda será chamada de demanda contínua, e as demais, de demanda discreta. Para fins de clareza, a nomenclatura utilizada na indústria foi alterada para ordem alfabética, e os produtos da categoria em que há retiradas periódicas foram chamados de K.

**Tabela 3** - Produtos considerados nas instâncias

K205	K227	K274	K290	A298	B358	C205	C276	D276	D300
D330	E205	E227	E250	E262	E285	E312	E332	E360	F190

## 4.2. ABORDAGEM DE PROGRAMAÇÃO INTEIRA MISTA

A modelagem matemática de programação inteira mista (PIM) – em inglês, *mixed integer programming (MIP)* – foi inspirada no problema do caixeiro viajante com coleta de prêmios (*prize collecting traveling salesman problem – PCTSP*), descrito no [Capítulo 3](#). A modelagem foi feita da seguinte forma: o sequenciamento é dividido em ciclos, tantos quanto necessários, e, ao final de cada um deles, o estoque dos produtos K205 e K274 é verificado. Cada ciclo é iniciado por um dos produtos da categoria K e contém pelo menos um produto das demais categorias. Como os produtos da categoria K – em particular K205 e K274 – são os únicos aos quais se aplicam restrições de manutenção de estoque, seu posicionamento como primeiro de cada ciclo permite que o estoque deles seja verificado apenas ao final dos ciclos, uma vez que, se apropriadamente verificada a restrição, é garantido que não houve violação do estoque mínimo no decorrer do ciclo. Optou-se por permitir que os produtos K227 e K290, aos quais restrições de estoque não se aplicam, também encabeçassem ciclos como *placeholders*, por pertencerem à categoria K. Isso foi feito para que esses produtos pudessem iniciar a produção e, assim, aumentar as possibilidades de sequências, e também para que houvesse maior espaçamento dos lotes de produtos desta categoria, possibilitando maior alternância na produção.

Para a resolução do problema, foram criados dois conjuntos a partir de produtos da categoria K, conforme seu perfil de consumo. O primeiro deles, chamado conjunto  $T$ , contém a porção dos produtos K destinada ao cliente que realiza retiradas constantes. A esse conjunto, aplicam-se as restrições de estoque, e é o conjunto que efetivamente encabeça os ciclos. Para os produtos K227 e K290, que não estão sujeitos às restrições de estoque, o estoque inicial e de segurança foram definidos como 0, bem como sua taxa de demanda.

O conjunto  $L$  traz os mesmos produtos K, porém trata da produção destinada a outros clientes. Como não há necessidade de verificação de estoque para esse conjunto, ele não encabeça os ciclos. O tempo de *setup* entre produtos iguais do conjunto  $T$  e do conjunto  $L$  foi definido como 0, porque, caso haja a justaposição de lotes de produtos iguais, na prática trata-se apenas de um único lote, em que parte da produção será destinada às retiradas contínuas, e outra parte às retiradas discretas. O conjunto com todos os produtos, da categoria K e das demais categorias, é chamado de  $V$ .

Os parâmetros, com exceção de  $Q$ , foram fornecidos pela indústria, e os valores numéricos são mostrados na [Seção 5.2](#).  $Q$  corresponde ao número de ciclos para o cenário em

a demanda de cada produto pertencente a  $T$  é dividida no maior número possível de lotes. Para cada um desses produtos, o número máximo de lotes é dado pelo piso do quociente entre a demanda do produto e o lote mínimo. Por exemplo, se a demanda de um dado produto é de 2000 toneladas, e o lote mínimo é de 360 toneladas, o número máximo de lotes corresponde a  $\left\lfloor \frac{2000}{360} \right\rfloor = \lfloor 5,56 \rfloor = 5$  lotes.

Os conjuntos considerados são:

$V$ : conjunto com todos os produtos

$T \subset V$ : porção dos produtos  $K$  destinada às retiradas contínuas

$L \subset V$ : porção dos produtos  $K$  destinada às retiradas discretas

São utilizados os seguintes parâmetros:

$d_j$ : demanda total (ton) de um produto  $j \in V$

$a_j$ : taxa de produção (ton/h) de um produto  $j \in V$

$b_j$ : taxa de demanda (ton/h) de um produto  $j \in T$

$s_{ij}$ : tempo de *setup* (h) de um produto  $i \in V$  para um produto  $j \in V$

$\bar{l}_j, l_j$ : lotes máximo e mínimo (ton), respectivamente, para um produto  $j \in V$

$h_j$ : estoque inicial (ton) para um produto  $j \in T$

$m_j$ : estoque de segurança (ton) para um produto  $j \in T$

$Q = \sum_{j \in T} \left\lfloor \frac{d_j}{l_j} \right\rfloor$ : limite superior para o número total de ciclos

E as variáveis:

$y_{jq}$ : 1, se o produto  $j \in V$  é fabricado no ciclo  $q$ . 0, caso contrário.

$x_{ij}^q$ : 1, se o produto  $j \in V$  é o sucessor imediato do produto  $i \in V$  no ciclo  $q$ . 0, caso contrário.

$z_{ij}^q$ : 1, se o produto  $i \in V \setminus T$  é o último no ciclo  $q$ , e o produto  $j \in T$  é o primeiro no ciclo  $q + 1$ . 0, caso contrário.

$f_{ij}^q$ : variável de fluxo entre  $i, j \in V$  no ciclo  $q$ , para evitar a formação de sub-rotas no ciclo do caixeiro viajante.

$t_{jq}$ : tempo de produção de  $j \in V$  no ciclo  $q$ .

Função objetivo:

$$\min \sum_{q=1}^Q \sum_{i \in V} \sum_{j \in V \setminus T} s_{ij} x_{ij}^q + \sum_{q=1}^Q \sum_{i \in V \setminus T} \sum_{j \in T} s_{ij} z_{ij}^q \quad (4.1)$$

Sujeito a:

$$\sum_{j \in T} y_{j1} = 1 \quad (4.2)$$

$$\sum_{j \in T} y_{jq} - \sum_{j \in T} y_{jq-1} \leq 0, \quad q = 2 \dots Q \quad (4.3)$$

$$y_{iq} - \sum_{j \in T} y_{jq} \leq 0, \quad \forall i \in V \setminus T; q = 1 \dots Q \quad (4.4)$$

$$\sum_{i \in V \setminus (T \cup L)} y_{iq} - \sum_{j \in T} y_{jq} \geq 0, \quad q = 1 \dots Q \quad (4.5)$$

$$\sum_{j \in V \setminus \{i\}} x_{ij}^q - y_{iq} = 0, \quad \forall i \in V; q = 1 \dots Q \quad (4.6)$$

$$\sum_{i \in V \setminus \{j\}} x_{ij}^q - y_{jq} = 0, \quad \forall j \in V; q = 1 \dots Q \quad (4.7)$$

$$\sum_{j \in V \setminus \{i\}} (f_{ij}^q - x_{ij}^q) - \sum_{j \in V \setminus (T \cup \{i\})} f_{ji}^q = 0, \quad \forall i \in V \setminus T; q = 1 \dots Q \quad (4.8)$$

$$f_{ij}^q - |V| x_{ij}^q \leq 0, \quad \forall i \in V \setminus T; j \in V; q = 1 \dots Q \quad (4.9)$$

$$x_{iu}^q + y_{jq+1} - z_{ij}^q \leq 1, \quad \forall i \in V \setminus T; u, j \in T; q = 1 \dots Q - 1 \quad (4.10)$$

$$a_j t_{jq} - \bar{l}_j y_{jq} \leq 0, \quad \forall j \in V; q = 1 \dots Q \quad (4.11)$$

$$a_j t_{jq} - l_j y_{jq} \geq 0, \quad \forall j \in V; q = 1 \dots Q \quad (4.12)$$

$$a_i t_{iq} + a_j t_{jq} + a_i t_{iq+1} + a_j t_{jq+1} \leq \bar{l}_j, \quad \forall i \in T, j \in L, i = j; q = 1 \dots Q - 1 \quad (4.13)$$

$$\sum_{q=1}^Q a_j t_{jq} = d_j, \quad \forall j \in V \quad (4.14)$$

$$h_j + \sum_{r=1}^{q-1} a_j t_{jr} - \sum_{r=1}^{q-1} \left( \sum_{i \in V} t_{ir} + \sum_{i \in V} \sum_{u \in V \setminus T} s_{iu} x_{iu}^r + \sum_{i \in V \setminus T} \sum_{u \in T} s_{iu} z_{iu}^r \right) b_j \geq m_j, \quad (4.15)$$

$$\forall j \in T; q = 2 \dots Q$$

$$f_{ij}^q \geq 0, \quad \forall i, j \in V; q = 1 \dots Q \quad (4.16)$$

$$t_{jq} \geq 0, \quad \forall j \in V; q = 1 \dots Q \quad (4.17)$$

$$z_{ij}^q \geq 0, \quad \forall i \in V \setminus T; \forall j \in T; q = 1 \dots Q \quad (4.18)$$

$$x_{ij}^q \in \{0,1\}, \quad \forall i, j \in V; q = 1 \dots Q \quad (4.19)$$

$$y_{jq} \in \{0,1\}, \quad \forall j \in V; q = 1 \dots Q \quad (4.20)$$

A função objetivo (4.1) minimiza o tempo total de *setup*. A primeira parcela trata do tempo de *setup* entre produtos no mesmo ciclo, e a segunda parcela trata do *setup* entre o último produto do ciclo  $q$  e o primeiro do ciclo  $q + 1$ .

As restrições de (4.2) a (4.5) definem a disposição dos produtos nos ciclos. A restrição (4.2) diz que exatamente um produto  $T \subset V$  é produzido no ciclo  $q = 1$ . As restrições (4.3) se asseguram de que, em cada ciclo  $q$ , produz-se no máximo o número de produtos pertencentes a  $T$  fabricados no ciclo  $q - 1$ , de modo que não haja ciclos vazios entre ciclos preenchidos e, quando a demanda for atingida, os ciclos subsequentes fiquem vazios. Essa restrição também garante que no máximo um produto pertencente a  $T$  é produzido em cada ciclo, pois todas elas estão atreladas, direta ou indiretamente, à restrição (4.2). As restrições (4.4) impõem que os produtos que não encabeçam os ciclos (conjunto  $L$  e produtos dos grupos A a F) podem ser fabricados apenas em ciclos contendo um produto pertencente ao conjunto  $T$ . As restrições (4.5) se asseguram de que no mínimo um produto dos grupos A a F ( $V \setminus (T \cup L)$ ) é fabricado em um ciclo que contenha um produto pertencente a  $T$ .

As restrições de (4.6) a (4.9) criam as rotas do problema do caixeiro viajante com prêmios, em conformidade com a descrição dada no referencial teórico ([Seção 3.2](#)). As restrições (4.6) e (4.7) dizem que, se o produto  $i$  ou  $j \in V$  é fabricado em um ciclo  $q$ , ele funciona como partida (4.6) e chegada (4.7) exatamente uma vez. As restrições (4.8) e (4.9) funcionam para evitar a formação de sub-rotas dentro de cada ciclo. As restrições (4.8) são uma adaptação das restrições ([3.13](#)), em que o prêmio por cada ponto visitado foi definido como  $w_i = 1$ , e o nó raiz foi considerado como o produto pertencente a  $T$  presente em cada ciclo  $q$ . As restrições (4.9) são iguais às restrições ([3.15](#)) para eliminação de sub-rotas, novamente considerando o nó raiz como o produto pertencente a  $T$ . Cabe destacar que, como cada ciclo  $q$  corresponde a um problema do caixeiro viajante com prêmios, as restrições do modelo explicado no Capítulo 3 foram escritas para cada um dos ciclos, de modo que  $x_{ij}$ ,  $y_j$  e  $f_{ij}$  tornaram-se, respectivamente,  $x_{ij}^q$ ,  $y_j^q$  e  $f_{ij}^q$ .

As restrições (4.10) adicionam as variáveis  $z_{ij}^q$ , que dizem se o produto  $i$  é o último do ciclo  $q$ , e o produto  $j$  é o primeiro do ciclo  $q + 1$ . Essas variáveis são necessárias para corrigir o cômputo do tempo de *setup*, pois, caso fossem utilizadas apenas as variáveis  $x_{ij}^q$ , seria considerado o *setup* do último produto do ciclo  $q$  para o primeiro do ciclo  $q$  e não haveria a ligação entre os ciclos  $q$  e  $q + 1$ . Então, na desigualdade  $x_{iu}^q + y_{jq+1} - z_{ij}^q \leq 1$ , com  $i \in V \setminus T$ ,  $u, j \in T$  e  $q = 1..Q - 1$ ,  $z_{ij}^q$  é forçada a ser maior ou igual a 1 se e somente se  $x_{iu}^q$  e  $y_{jq+1}$  forem iguais a 1, o que significa, respectivamente, que o produto  $i$  no ciclo  $q$  retorna a um produto  $T \subset V$  e que o produto  $j \in T$  está presente no ciclo  $q + 1$ . Como  $z_{ij}^q$  está presente na função objetivo, essa variável assume o valor 1 nessas condições. Caso uma delas ou ambas não se apliquem,  $z_{ij}^q$  passa a valer 0, pois está limitada pelo domínio da variável – restrição (4.18) – e está presente na função objetivo.

As restrições (4.11) e (4.12) funcionam como restrições de ativação e limitam o tamanho dos lotes entre os valores máximo (4.11) e mínimo (4.12). As restrições (4.13) limitam ao lote máximo a fabricação de produtos  $K$  (conjuntos  $T$  e  $L$ ) iguais em ciclos consecutivos. As restrições (4.14) asseguram o cumprimento da demanda total para cada produto, e as restrições (4.15) estabelecem a manutenção de um estoque de segurança para os produtos  $T \subset V$ . A parcela  $h_j$  contabiliza o estoque inicial dos produtos, e a parcela  $\sum_{r=1}^{q-1} a_j t_{jr}$  soma a produção do produto  $j$  até a finalização do ciclo  $q - 1$ . Por fim, a parcela  $\sum_{r=1}^{q-1} (\sum_{i \in V} t_{ir} + \sum_{i \in V} \sum_{u \in V \setminus T} s_{iu} x_{iu}^r + \sum_{i \in V \setminus T} \sum_{u \in T} s_{iu} z_{iu}^r) b_j$  desconta o consumo do produto  $j \in T$  até o final do ciclo  $q - 1$ , contabilizando o tempo total de produção e de *setup* em todos os ciclos predecessores.

As restrições de (4.16) a (4.20) fornecem o domínio das variáveis. Embora as variáveis  $z_{ij}^q$  sejam conceitualmente binárias, elas podem ser programadas como contínuas, pois estão limitadas a 0 ou 1 pelas restrições (4.10), pelo seu domínio – restrições (4.18) – e pela função objetivo.

Além dessas restrições, é necessário incluir outras cuja implementação varia conforme as instâncias, a respeito de trocas que devem ser vetadas, posicionamento de produtos da categoria E e posicionamento de um produto em particular. Determinadas trocas são desencorajadas pela indústria porque, apesar de existir um tempo de *setup* tabelado para elas, na prática elas não são realizadas, pois o segundo produto da sequência apresentaria imperfeições. Trocas entre produtos  $K$  distintos também foram vetadas, para que não houvesse

sequências prolongadas desses produtos. Apesar de não haver essa restrição nas sequências fornecidas pela indústria, optou-se por implementá-la neste trabalho, porque foram sempre alocados os maiores lotes possíveis, até o limite de 5 mil toneladas, enquanto os lotes das sequências fornecidas pela indústria são tipicamente menores. Foram incluídas limitações para o número de lotes de produtos E por ciclo ou produtos E consecutivos, a pedido da indústria. Por fim, para uma das instâncias, o produto D300 deveria ser incluído no primeiro ciclo, também a pedido da indústria. A implementação dessas restrições foi detalhada, instância a instância, na [Seção 5.3](#).

#### 4.3. ABORDAGEM HEURÍSTICA

Sequências viáveis foram construídas por um método baseado na etapa de construção da meta-heurística GRASP. Foi dada prioridade aos dois produtos com restrição de estoque, de modo que o estoque de segurança fosse sempre respeitado. Como alternativa à divisão da produção em ciclos e verificação do estoque ao final de cada um deles, realizadas no método anterior, o estoque dos produtos K205 e K274, neste método, foi verificado a cada inserção de produto.

A rotina se inicia com o cálculo do tempo da duração do estoque desses dois produtos. O produto de menor tempo preenche o primeiro espaço da sequência, e o lote máximo possível para ele é feito a menos que não haja tempo suficiente para alocar um lote de outro produto entre ele e o próximo atingir o estoque de segurança. O encaixe de outro produto é necessário porque dois produtos desta natureza não podem suceder um ao outro, por serem ambos da categoria K. Alocado o primeiro produto, os demais são alocados por meio de sorteio dentre os sucessores possíveis que possuem menor tempo de *setup* em relação ao antecessor. O número de elementos presentes neste sorteio é um parâmetro de entrada e, quanto menor esse número, mais guloso é o algoritmo. Esses outros produtos são alocados até que um produto com restrição de estoque seja sorteado ou que não seja mais possível alocar lotes de outros produtos sem que o estoque de segurança seja violado. O procedimento é repetido até que todos os produtos tenham sido alocados na sequência.

Finalizada a construção da sequência, foi implementada uma verificação quanto à viabilidade da mesma, por meio de uma reinterpretação das restrições implementadas para lotes de produtos K em ciclos consecutivos, número de lotes de produtos E por ciclo ou em sequência

e posicionamento do produto D300 no primeiro ciclo. Feita essa verificação, o algoritmo retorna uma sequência viável.

Na heurística de construção, foi dado tratamento diferente apenas para os produtos K205 e K274. Como o conceito dos ciclos foi desfeito, os produtos K227 e K290 foram tratados como os produtos das outras categorias.

Os conjuntos considerados na heurística de construção foram:

$V$ : produtos a serem sequenciados

$U \subset V$ : produtos da categoria K sujeitos a retiradas contínuas

A heurística de construção se aproveita de alguns parâmetros já utilizados no modelo do problema do caixeiro viajante com coleta de prêmios. São eles:

$d_j$ : demanda total (ton) de um produto  $j \in V$

$a_j$ : taxa de produção (ton/h) de um produto  $j \in V$

$b_j$ : taxa de demanda (ton/h) de um produto  $j \in U$

$s_{ij}$ : tempo de *setup* (h) de um produto  $i \in V$  para um produto  $j \in V$

$\bar{l}_j$ : lote máximo (ton) para um produto  $j \in U$

$h_j$ : estoque inicial (ton) para um produto  $j \in U$

$m_j$ : estoque de segurança (ton) para um produto  $j \in U$

Pela experiência a partir das execuções do algoritmo de PIM baseado no problema do caixeiro viajante com coleta de prêmios, o lote máximo foi considerado apenas para os produtos  $U \subset V$ , e lote mínimo não foi utilizado na heurística. Observou-se, no PCVCP, que não havia a divisão de um produto em múltiplos lotes a não ser que a demanda fosse superior a 5000 toneladas, o que ocorre apenas para os produtos pertencentes a  $U \subset V$ .

Adicionou-se um parâmetro para fornecer a quantidade, em toneladas, dos produtos do conjunto  $U \subset V$  destinados às retiradas contínuas:

$dk_j$ : demanda (ton) de um produto  $j \in U$  destinada às retiradas contínuas

As trocas não permitidas foram implementadas por meio de uma matriz  $f$ , em que:

$$f_{ij} = \begin{cases} 1, & \text{se a troca é permitida} \\ 0, & \text{se a troca não é permitida} \end{cases}, i, j \in V$$



A heurística retorna a matriz *seq*, que contém a sequência obtida. Ela é dada por uma matriz contendo duas colunas e tantas linhas quanto necessárias, uma para cada lote fabricado. Na primeira coluna, é dado o elemento que foi alocado e, na segunda coluna, o dimensionamento do lote alocado. A heurística também retorna o escalar *tst*, que fornece o tempo total de *setup* da sequência gerada. O pseudocódigo é dado a seguir. Uma versão com mais detalhes computacionais é dada no [Apêndice A](#), ao final do trabalho.

1:	<b>Função</b> [ <i>seq</i> , <i>tst</i> ] = Sequência( <i>tam<sub>lrc</sub></i> , <i>d</i> , <i>a</i> , <i>b</i> , <i>s</i> , $\bar{l}$ , <i>h</i> , <i>m</i> , <i>dk</i> , <i>f</i> )
2:	Calcular <i>dt</i> : a demanda dos produtos em função do tempo
3:	Calcular <i>td</i> : tempo para que os produtos de retirada constante atinjam o estoque de segurança
4:	Definir <i>v</i> : o primeiro produto em <i>U</i> a atingir o estoque de segurança e a ser alocado
5:	<b>Enquanto</b> houver demanda para retiradas contínuas a alocar ( <i>soma</i> ( <i>dk</i> ) > 0)
6:	<b>Enquanto</b> <i>v</i> ∈ <i>U</i>
7:	Alocar <i>v</i> na sequência <i>seq</i>
8:	<b>Se</b> ainda houver demanda do outro produto em <i>U</i> para retiradas constantes por cumprir
9:	Calcular o tempo necessário para alocar um produto $j \in V \setminus U$ entre os dois produtos pertencentes a <i>U</i>
10:	<b>Se</b> não houver tempo suficiente para alocar o lote máximo de <i>v</i> e o menor lote possível dentre os outros produtos antes que o outro produto em <i>U</i> atinja o estoque de segurança
11:	Alocar a quantidade possível de <i>v</i>
12:	Alocar o produto de menor tempo de produção e <i>setup</i>
13:	Atualizar <i>dt</i> : demanda dos dois produtos alocados
14:	Atualizar <i>dk</i> : demanda destinada a retiradas contínuas
15:	Atualizar estoque dos produtos em <i>U</i>
16:	Recalcular <i>td</i> : duração do estoque dos produtos em <i>U</i>
17:	Definir <i>v</i> como o produto que atinge primeiro o estoque de segurança
18:	<b>Senão</b>
19:	Alocar o lote máximo possível
20:	<b>Fim Se</b>
21:	<b>Senão</b>
22:	Alocar o lote máximo possível
23:	<b>Fim Se</b>
24:	<b>Se</b> o último produto alocado foi um dos pertencentes a <i>U</i>
25:	Atualizar <i>dt</i> : demanda total do produto alocado
26:	Atualizar <i>dk</i> : demanda destinada a retiradas contínuas
27:	Atualizar estoque dos produtos em <i>U</i>
28:	Recalcular <i>td</i> : duração do estoque dos produtos em <i>U</i>
29:	Calcular <i>tl<sub>j</sub></i> : tempo para fabricar um lote de um produto $j \in V \setminus U$ entre o último produto alocado e o próximo produto em <i>U</i> a esgotar
30:	Definir um valor arbitrário para <i>v</i> , para entrada no <i>loop</i>
31:	<b>Fim Se</b>
32:	<b>Fim Enquanto</b>

33:	<b>Enquanto</b> $\min(tl > 0) < \min(td) \& soma(dk) > 0 \& v \notin U$
34:	Definir os possíveis sucessores
35:	Construir a LRC com os $tam_{trc}$ elementos de menor tempo de <i>setup</i>
36:	Realizar o sorteio dentre os elementos da LRC para encontrar o elemento $v \in V$ a ser alocado
37:	<b>Se</b> $v \notin U$
38:	Alocar em <i>seq</i> o produto sorteado, com o lote igual à demanda por cumprir
39:	Atualizar <i>dt</i> : demanda do produto alocado
40:	Atualizar estoque dos produtos em <i>U</i>
41:	Calcular duração do estoque dos produtos em <i>U</i>
42:	Calcular $tl_j$ : tempo para fabricar um lote de um produto $j \in V \setminus U$ entre o último produto alocado e o próximo produto em <i>U</i> a esgotar
43:	<b>Fim Se</b>
44:	<b>Fim Enquanto</b>
45:	<b>Fim Enquanto</b>
46:	<b>Enquanto</b> $soma(dt) > 0$
47:	Definir os possíveis sucessores
48:	Construir a LRC com os $tam_{trc}$ elementos de menor tempo de <i>setup</i>
49:	Realizar o sorteio dentre os elementos da LRC para encontrar o elemento $v \in V$ a ser alocado
50:	Alocar em <i>seq</i> o produto sorteado, com o lote igual à demanda por cumprir
51:	Atualizar <i>dt</i> : demanda do produto alocado
52:	<b>Fim Enquanto</b>
53:	Verificar se as restrições de produtos K em ciclos consecutivos são respeitadas
54:	Verificar se as restrições de produtos E em um mesmo ciclo ou em sequência são respeitadas
55:	Calcular <i>tst</i> : tempo total de <i>setup</i>
56:	Retorna <i>seq</i> e <i>tst</i>
57:	<b>Fim Função</b>

A linha 1 chama a função Sequência, a qual possui como variáveis de entrada: o tamanho da lista restrita de candidatos, a demanda e a taxa de produção de todos os papéis-cartão, a taxa de demanda dos produtos com retirada contínua, a matriz contendo os tempos de *setup* para cada par de produtos, o lote máximo permitido para os produtos de retirada contínua, o estoque inicial, de segurança e a demanda em toneladas destinada às retiradas contínuas e a matriz informando os pares de trocas permitidas ou não.

Na linha 2, é calculada a demanda temporal dos produtos, dividindo-se a demanda  $d$  pela taxa de produção  $a$ . Dessa forma, a maior parte das variáveis do problema passa a estar em unidade de horas. Na linha 3, é calculado o tempo para que cada um dos produtos do conjunto  $U$  atinja o estoque de segurança. Na linha 4,  $v$  recebe o primeiro produto em  $U$  a atingir o estoque de segurança, que será o primeiro a ser alocado.

Na linha 5, inicia-se o *loop* **Enquanto** a ser percorrido até que as demandas destinadas às retiradas contínuas sejam alocadas, o qual é encerrado na linha 45. Dentro deste *loop*, ocorre a verificação do nível de estoque dos produtos do conjunto  $U$  a cada inserção de um novo produto. Ele é composto por outros dois *loops*: no primeiro, ocorre a alocação de produtos pertencentes ao conjunto  $U$ . No segundo, ocorre a alocação dos outros produtos.

Na linha 6 é iniciado o *loop* **Enquanto** para alocar produtos com demandas constantes, os quais demandam procedimentos particulares para o dimensionamento dos lotes. Este *loop* encerra-se na linha 32. Na linha 7, é alocado o produto  $v$ , definido anteriormente (linha 4) como o que atinge o estoque de segurança em menos tempo. Na linha 8, é colocada uma condição **Se** para verificar se o outro produto pertencente a  $U$  ainda possui demanda constante por cumprir, pois, nesse caso, pode ser necessário encurtar o lote do produto  $v$  para que o outro não atinja nível de estoque inferior ao mínimo. Caso o outro produto ainda possua demanda constante por cumprir, prossegue-se para a linha 9. Nela, determina-se o tempo necessário para alocar cada um dos produtos  $j \in V \setminus U$  possíveis e disponíveis entre os dois produtos pertencentes a  $U$ , contando tempo de produção e tempo de *setup*. Isso é necessário, pois, de acordo com a matriz  $f$ , os dois produtos pertencentes a  $U$  não podem ser sucessores um do outro, então, para que a sequência seja viável, deve haver pelo menos um produto entre eles.

Na linha 10, há outro condicional **Se**, que verifica se é ou não possível alocar o maior lote do produto pertencente a  $U$ . Isso é feito somando-se o tempo do lote máximo ao menor tempo para alocar um produto  $j \in V \setminus U$  entre os dois produtos  $U$ . Caso a soma seja inferior ao tempo para esgotamento do outro produto pertencente a  $U$ , é possível alocar o lote máximo, e o algoritmo segue para a linha 18. Caso contrário, não é possível tal alocação, e prossegue-se para a linha 11. Nela, é alocada a quantidade possível do produto, que corresponde à diferença entre tempo para esgotamento do outro produto pertencente a  $U$  e o menor tempo para alocação de um produto  $j \in V \setminus U$ . Nessa situação, não é necessário realizar o procedimento para sorteio do sucessor do produto alocado, pois, da maneira como o código foi construído, o único produto possível de ser posicionado antes que o outro pertencente a  $U$  se esgote é justamente o de menor tempo de alocação, que foi utilizado no cálculo da linha 11. Sendo assim, na linha 12, ocorre a alocação desse produto. Na linha 13, são atualizadas as demandas dos dois produtos alocados. Na linha 14, atualiza-se a demanda do produto pertencente a  $U$  destinada às retiradas constantes. Na linha 15, são atualizados os estoques dos produtos com retiradas constantes e, na linha 16, recalcula-se o tempo para que ambos atinjam o estoque de segurança. Na linha 17, a variável  $v$  recebe o próximo produto a atingir o estoque de segurança.

Na linha 18, há o **Senão** para a condição imposta na linha 10, que contempla a situação em que é possível fabricar o lote máximo do produto pertencente a  $U$ . Na linha 19, é alocado o maior lote disponível, que corresponde ao valor máximo entre  $\bar{l}_j$  e  $dt_j$  naquele momento. Na linha 20, encerra-se o **Se** iniciado na linha 10.

Na linha 21, há o **Senão** como alternativa para o **Se** colocado na linha 8. Ele corresponde à situação em que a demanda do outro produto em  $U$  destinada às retiradas constantes já foi atendida, de modo que não é necessário realizar o encurtamento do lote. Na linha 22, aloca-se o maior lote disponível. Na linha 23, encerra-se o **Se** iniciado na linha 8.

Na linha 24, há um condicional **Se** para o caso em que não houve alocação automática de um produto  $j \in V \setminus U$  devido às restrições de estoque (linha 12). Das linhas 25 a 28, realiza-se procedimento similar ao feito entre as linhas 13 e 16, com atualização das demandas, do estoque e da duração do estoque. Na linha 29, é calculado para cada produto  $j \in V \setminus U$  o tempo  $tl$  necessário para alocar um lote deste entre o último produto alocado e o produto pertencente a  $U$  que atingirá o estoque de segurança em menos tempo. Caso o produto já tenha sido alocado ou não seja um possível sucessor do último produto alocado devido às trocas não permitidas,  $tl_j$  recebe o valor de 0. Na linha 30, define-se um valor arbitrário para a variável  $v$  (como  $v = 0$ ), de modo que ela respeite a condição  $v \notin U$  para entrada no *loop* iniciado na linha 33. Na linha 31, encerra-se o condicional **Se** iniciado na linha 24. Para a situação em que houve alocação automática de um produto  $j \in V \setminus U$ , esse condicional não é percorrido, e retorna-se para a linha 7 para a alocação de um novo produto  $v \in U$ . Na linha 32, encerra-se o *loop* **Enquanto** iniciado na linha 6.

Na linha 33, inicia-se o *loop* **Enquanto** para alocação dos produtos não sujeitos a restrição de estoque, que irá ser percorrido caso as demandas contínuas não tenham sido atendidas. Nele, é realizada a verificação do nível de estoque dos produtos em  $U$  a cada iteração. Este *loop* é interrompido caso pelo menos uma das seguintes hipóteses seja verdadeira: se não houver tempo suficiente para alocar um produto  $j \in V \setminus U$  antes que um produto pertencente a  $U$  atinja o estoque de segurança ou se um elemento pertencente a  $U$  for sorteado dentro do *loop*.

Na linha 34, são definidos os possíveis sucessores do último elemento alocado. São excluídos os produtos com a demanda já cumprida, os de troca não permitida e o produto em  $U$  que não é o próximo a esgotar. Na linha 35, é construída a lista restrita de candidatos. Dentre os possíveis sucessores do último elemento alocado, são incluídos os  $tam_{trc}$  produtos com menor tempo de *setup*. Se houver empate no tempo de *setup*, todos os elementos com aquele tempo são incluídos. Exemplo: caso os possíveis candidatos possuam os valores incrementais

dados em  $l = \{1,3,3,4,6,7\}$ , e  $tam_{lrc}$  seja igual a 2, o valor de corte é igual a 3. Porém, como há dois elementos de valor 3, ambos são incluídos na LRC.

Na linha 36, é sorteado um dentre os elementos da LRC para que seja o próximo produto a ser alocado. Na linha 37, é colocada uma condição para verificar se o produto sorteado foi ou não o produto sujeito à restrição de estoque. Em caso afirmativo, o código retorna para a linha 6, porque ele não passa pelo procedimento do **Se** entre as linhas 38 e 42 e deixa de atender à condição  $v \notin U$  do **Enquanto** colocada na linha 33. Se o produto sorteado não for sujeito às restrições de estoque, passa-se para a linha 38, em que o produto sorteado é alocado. Na linha 39, é atualizada a demanda daquele produto. Nas linhas 40 e 41, atualizam-se os estoques dos produtos pertencentes a  $U$  e calcula-se a duração do estoque dos mesmos. Na linha 42, repete-se o procedimento da linha 29, em que é calculado o tempo para se alocar cada produto entre o último posicionado e o próximo produto em  $U$  a atingir o estoque mínimo. Na linha 43, finaliza-se o **Se** iniciado na linha 37. Na linha 44, é finalizado o *loop* iniciado na linha 33. Na linha 45, é finalizado o *loop* iniciado na linha 5. O *loop* das linhas 5 a 45 é encerrado assim que a porção dos produtos  $U$  destinados a retiradas contínuas é posicionada na sequência.

A partir do momento em que a demanda destinada às retiradas contínuas é atendida, tem-se a garantia de que o estoque desses produtos é suficiente para atender à demanda até o final do período de planejamento. Dessa forma, deixa de ser necessária a verificação do estoque a cada produto posicionado. Sendo assim, das linhas 46 a 52, há uma versão simplificada do *loop* mostrado entre as linhas 33 a 44, sem a atualização do estoque dos produtos pertencentes a  $U$ , e em que é possível alocar qualquer um dos produtos em  $U$ , desde que a troca seja permitida e que ainda haja demanda deles por cumprir.

Na linha 46, inicia-se o *loop* **Enquanto** para alocação dos produtos após a demanda destinada às retiradas constantes ter sido atendida. Nas linhas 47 a 49, são repetidos os procedimentos das linhas 34 a 36, para construção da LRC e sorteio do próximo produto. Nas linhas 50 e 51, são feitos os mesmos procedimentos das linhas 38 e 39, para alocação do produto sorteado e atualização da lista de demandas. Na linha 52, encerra-se este *loop*.

Nesse ponto do código, existe uma sequência já formada na variável de saída *seq*, porém é necessário verificar se ela atende às restrições relativas aos ciclos. Na linha 53, verifica-se a restrição para produtos  $K$  em ciclos consecutivos. Na linha 54, verifica-se o número de produtos da categoria  $E$  por ciclo ou consecutivos, conforme a instância. Os detalhes de como a reinterpretação dos ciclos foi feita são dados na [Seção 5.4](#).

Caso a sequência gerada seja viável, na linha 55 é calculado o tempo total de *setup* da mesma. Na linha 56, são retornados a matriz contendo a sequência dos produtos com seus respectivos tempos de produção e o escalar contendo o tempo total de *setup*. Na linha 57, a função é encerrada.

No [Apêndice B](#) deste trabalho, um pequeno exemplo numérico, contendo dois produtos com retiradas constantes e 7 produtos no total, é resolvido detalhadamente por meio da heurística desenvolvida.

## 5. ESTUDO COMPUTACIONAL

### 5.1. AMBIENTES DE TESTE

O método de solução por programação inteira mista, inspirado no problema do caixeiro viajante com coleta de prêmios, foi implementado em linguagem OPL, e as instâncias foram resolvidas no *software* CPLEX Studio IDE 12.10.0 na configuração padrão. Os testes foram realizados em um computador com processador Intel i7 de memória RAM 16 GB, CPU de 3,60 GHz e sistema operacional Windows 10. Para o método heurístico de construção gulosa, aleatorizada e adaptativa, foi utilizado o *software* Matlab R2010a, e os códigos foram executados em um computador com processador Intel i5 de memória RAM igual a 8 GB, CPU de 2,20 GHz e sistema operacional Windows 10.

### 5.2. PARÂMETROS UTILIZADOS NAS INSTÂNCIAS

A resolução integrada dos problemas de dimensionamento e sequenciamento de lotes foi realizada para quatro instâncias, cada uma delas correspondendo a um mês de planejamento na indústria. Os dados utilizados para cada mês foram fornecidos pela indústria, a saber: demanda mensal de todos os produtos e taxa de demanda, estoque inicial e estoque de segurança para os produtos de retiradas contínuas. Houve apenas a troca de nomenclatura em relação aos dados industriais, para fins de clareza deste trabalho, como dito no Capítulo 4. As taxas de produção e os tempos de *setup* se repetem para todas as instâncias e foram também fornecidos pela indústria. A Tabela 4 mostra a demanda, em toneladas, de cada um dos produtos para os meses considerados. Cabe lembrar que os produtos K são divididos entre retiradas contínuas e retiradas discretas. O primeiro grupo trata da parcela da produção retirada periodicamente pelo cliente e está sujeita às restrições de manutenção de estoque, e o segundo grupo está desvinculado dessas restrições, pois são destinados a outros clientes.

A Tabela 5 traz os estoques iniciais e de segurança, em toneladas, para os produtos K205 e K274 destinados a retiradas contínuas, nos quatro meses considerados. O estoque de segurança é combinado com o cliente desses produtos e possui o propósito de atender aumentos imprevistos na demanda. A tabela também mostra a previsão de consumo desses produtos ao

**Tabela 4 - Demanda mensal dos produtos, em toneladas**

		<b>Mês 1</b>	<b>Mês 2</b>	<b>Mês 3</b>	<b>Mês 4</b>
<b>K205</b>	Contínua	3039	2334	3559	4624
	Discreta	4417	5319	4041	1903
<b>K227</b>	Contínua	0	0	0	0
	Discreta	854	2135	1460	1461
<b>K274</b>	Contínua	8964	8810	9110	9611
	Discreta	498	590	590	389
<b>K290</b>	Contínua	0	0	0	0
	Discreta	898	500	500	0
<b>A298</b>		650	1000	1450	1750
<b>B358</b>		700	1039	400	785
<b>C205</b>		1000	2973	2040	3500
<b>C276</b>		2300	1450	400	2473
<b>D276</b>		0	0	0	350
<b>D300</b>		350	350	650	350
<b>D330</b>		0	351	0	0
<b>E205</b>		1147	500	500	468
<b>E227</b>		200	250	0	168
<b>E250</b>		600	0	0	0
<b>E262</b>		500	0	0	598
<b>E285</b>		750	771	1200	935
<b>E312</b>		1000	600	2529	1464
<b>E332</b>		1100	1000	1270	731
<b>E360</b>		1102	1256	1108	1194
<b>F190</b>		750	350	0	345

longo do mês, em toneladas por hora. Na programação inteira mista, os valores de estoque inicial, estoque de segurança e taxa de demanda foram considerados nulos para os produtos K227 e K290, para os quais houve apenas retiradas discretas.

A Tabela 6 traz a taxa de produção, em toneladas por hora, de cada uma das variedades de papéis-cartão produzidos na máquina estudada. As taxas são as mesmas para todas as instâncias.

A Tabela 7 apresenta os tempos de *setup*, em minutos, para cada par de papel-cartão. O tempo é assimétrico, ou seja, depende dos elementos envolvidos na troca e da ordem em que esta é realizada: o tempo de troca de A para B não é necessariamente igual ao tempo de B para A. A tabela mostra o tempo de *setup* do elemento contido na linha para o elemento contido na coluna. Por exemplo, o tempo de *setup* de D330 para C276 é de 52 minutos, enquanto o tempo de C276 para D330 é de 35 minutos.



**Tabela 5** - Estoques iniciais e de segurança e previsão de demanda para os produtos K205 e K274

	<b>Estoque inicial (ton)</b>	<b>Estoque de segurança (ton)</b>	<b>Taxa de demanda (ton/h)</b>
<b>Mês 1</b>			
<b>K205</b>	1860	890	4,6344
<b>K274</b>	2577	2099	10,9341
<b>Mês 2</b>			
<b>K205</b>	547	443	3,0764
<b>K274</b>	3313	1205	8,3708
<b>Mês 3</b>			
<b>K205</b>	760	600	4,6344
<b>K274</b>	3809	1800	14,8374
<b>Mês 4</b>			
<b>K205</b>	913	620	3,6917
<b>K274</b>	2500	1700	10,1194

**Tabela 6** - Taxas de produção, em toneladas por hora, de cada espécie de papel-cartão

<b>Espécie</b>	<b>Taxa de produção (ton/h)</b>	<b>Espécie</b>	<b>Taxa de produção (ton/h)</b>
<b>K205</b>	46,3261	<b>D330</b>	42,6743
<b>K227</b>	47,5078	<b>E205</b>	50,4407
<b>K274</b>	48,0534	<b>E227</b>	53,1469
<b>K290</b>	46,2250	<b>E250</b>	40,5405
<b>A298</b>	46,1538	<b>E262</b>	42,8444
<b>B358</b>	47,6242	<b>E285</b>	47,0153
<b>C205</b>	42,9967	<b>E312</b>	47,2424
<b>C276</b>	48,8821	<b>E332</b>	50,5776
<b>D276</b>	47,9233	<b>E360</b>	48,2823
<b>D300</b>	50,1079	<b>F190</b>	43,7569

Restrições visando bloquear trocas de elementos do conjunto {K205, C205, F190} para o conjunto {A298, B358} e vice-versa foram incluídas nas quatro instâncias, devido à natureza discrepante desses materiais e à possibilidade de falhas estruturais caso sejam colocados em sequência. No mês 4, também foram vetadas trocas entre A298 e B358, uma vez que a produção dos dois em sequência poderia acarretar esgotamento de matéria-prima, em função do volume de produção dessas categorias estipulado para o mês. Trocas entre produtos K de gramaturas diferentes também foram vetadas nas quatro instâncias, para evitar longas sequências, o que poderia comprometer a operação das rebobinadeiras.

Em adição, foram incluídas restrições quanto ao número de produtos E fabricados em cada ciclo, por se tratar da classe de produtos com maior diversidade de gramaturas. Para os meses 1, 2 e 4, o limite foi imposto como 4 desses produtos por ciclo. No mês 3, o limite foi de

Tabela 7 - Tempo de *setup*, em minutos, para os pares de papel-cartão

	K205	K227	K274	K290	A298	B358	C205	C276	D276	D300	D330	E205	E227	E250	E262	E285	E312	E332	E360	F190
K205		20	38	41	27	27	18	37	27	27	27	19	30	36	36	51	36	36	36	36
K227	18		25	41	27	27	18	37	27	27	27	37	21	36	36	36	36	36	36	36
K274	30	28		41	33	33	30	37	27	27	29	23	30	34	20	17	36	32	46	36
K290	41	41	41		27	27	41	37	27	27	27	36	36	36	36	54	36	36	36	36
A298	45	45	45	45		33	45	35	20	20	20	30	34	28	30	28	30	40	54	30
B358	45	45	45	45	25		45	35	20	20	20	30	34	28	30	28	30	40	54	30
C205	38	38	38	38	56	56		41	35	35	35	42	42	42	42	42	42	42	42	42
C276	38	38	38	38	56	56	38		35	35	35	42	42	42	42	42	42	42	42	42
D276	36	37	36	40	35	35	36	52		17	17	49	52	40	40	40	40	40	40	40
D300	88	36	21	36	35	35	88	52	17		17	40	40	40	40	40	40	40	40	40
D330	36	36	28	36	35	35	36	52	17	17		40	40	40	40	40	40	40	42	40
E205	34	35	32	35	25	25	34	45	34	34	34		35	37	32	32	32	32	32	32
E227	25	10	22	35	24	24	25	45	48	34	34	25		35	10	32	32	32	32	32
E250	35	35	35	42	25	25	35	32	40	34	34	21	32		17	32	32	32	32	32
E262	35	17	15	57	28	28	35	45	34	34	34	31	16	24		32	32	26	32	32
E285	35	35	17	41	25	25	35	45	22	34	34	57	32	42	16		32	32	32	32
E312	35	35	35	35	25	25	35	45	34	34	34	32	32	32	32	32		32	32	32
E332	35	35	26	35	29	29	35	45	34	34	34	32	32	32	32	32	32		32	32
E360	35	35	35	35	25	25	35	45	34	34	34	32	32	32	32	31	32	32		32
F190	35	35	35	35	25	25	35	45	34	34	34	32	32	32	32	32	32	32	32	

3 produtos E em sequência. Essas restrições foram assim implementadas por sugestão da indústria indústria, após análise de sequências obtidas preliminarmente. Também no mês 3, o papel-cartão D300 foi alocado no primeiro ciclo, de modo a ser entregue no início do mês para atender a uma demanda de maior urgência.

Para comparação com o resultado deste trabalho, são trazidas as sequências fornecidas pela indústria para as quatro instâncias estudadas, elencadas na Tabela 8. Todas as unidades são dadas em toneladas. Para os produtos K205 e K274, o número fora dos parênteses representa a massa total produzida no lote, e o número entre parênteses representa a massa destinada às retiradas discretas, não sujeitas às restrições de estoque.

Como mencionado na descrição dos modelos, as sequências da indústria contam com lotes consecutivos de produtos K, como 3010 toneladas de K274 seguidas de 3456 toneladas de K205 dentre os últimos produtos do mês 1, totalizando 6466 toneladas. Neste trabalho, optou-se por não permitir tais sucessões, porque foi frequente a fabricação de lotes de 5 mil toneladas – o maior valor permitido –, uma vez que, para a otimização do tempo de *setup*, é mais interessante condensar a produção no menor número possível de lotes.

**Tabela 8 - Sequências fornecidas pela indústria**

Mês 1		Mês 2		Mês 3		Mês 4	
K274	3347 (382)	K205	1675 (1043)	E332	600	K274	2400 (129)
C276	1300	C205	1473	E360	1108	C276	400
E262	500	K205	1500 (1000)	E312	1200	C205	1500
K205	4000 (2538)	K227	672	D300	650	K205	3250 (1903)
E205	590	E227	250	K274	3600 (460)	E205	468
C205	1000	C276	1000	C276	400	D276	350
F190	750	K274	3800 (295)	K290	500	K274	2800 (260)
K227	854	A298	1000	C205	1000	E360	1194
E227	200	E360	1256	K205	3800 (2241)	E332	731
E250	600	E332	1000	K274	1500 (0)	E312	1464
K274	3105 (0)	B358	1039	K227	1460	E285	935
C276	1000	E312	600	A298	1450	K274	2400 (0)
D300	350	E285	771	B358	400	E262	598
K290	898	K274	2000 (295)	E285	1200	K205	3277 (0)
E285	750	C205	1500	K274	2100 (0)	K274	2400 (0)
A298	650	K205	2400 (1198)	C205	1040	K227	1461
B358	700	F190	350	K205	3800 (1800)	B358	785
K274	3010 (116)	K227	1463	E205	500	D300	350
K205	3456 (1879)	K274	3600 (0)	K274	2500 (130)	C205	2000
E205	557	C276	450	E312	1329	C276	2073
E312	1000	K290	500	E332	670	A298	1750
E332	1100	K205	2078 (2078)			F190	345
E360	1102	E205	500			E227	168
		D300	350				
		D330	351				
<b>Tempo total de setup (h)</b>							
12,3413		12,5905		10,4947		12,8746	

As restrições impostas neste trabalho a respeito do posicionamento dos produtos E foram todas observadas nas sequências fornecidas pela indústria. Quanto à restrição a respeito de trocas entre os conjuntos  $\{K205, C205, F190\}$  e  $\{A298, B358\}$ , houve uma inconsistência na sequência para o mês 4, em que há um lote de A298 sucedido por um lote de F190.

### 5.3. IMPLEMENTAÇÃO DO MÉTODO DE PROGRAMAÇÃO INTEIRA MISTA

#### 5.3.1. Implementação das variáveis e restrições

O modelo baseado no problema do caixeiro viajante com coleta de prêmios (PCVCP) foi implementado de acordo com a função objetivo e restrições descritas na [Seção 4.2](#). Parâmetros como taxa de produção ( $a_j$ ) e demanda total ( $d_j$ ) para cada produto foram ditados pela indústria, conforme seção anterior. O lote máximo adotado foi de 5000 toneladas para os produtos K e de tantas toneladas quanto estipuladas no mês para as demais classes, nos quatro meses – como exemplo, o lote máximo de E205 no mês 1 foi igual a 1174 toneladas. Para o lote mínimo, adotou-se, para a maior parte dos produtos, o valor de 350 toneladas. Para os produtos K destinados à demanda constante (conjunto T), o lote mínimo foi igual a 0, para que houvesse a possibilidade de que eles servissem como *placeholders* encabeçando ciclos, mesmo que a demanda deles fosse nula (K227 ou K290). No cálculo inicial do valor de  $Q$  – número limite de ciclos –, foi mantido o valor de 350 toneladas, para que não houvesse divisão por 0 devido aos lotes mínimos nulos para o K227 e o K290. Para os produtos de demanda mensal não nula e inferior a 350 toneladas – E227 nos meses 1, 2 e 4 e F190 no mês 4 –, o lote mínimo foi considerado como exatamente a demanda mensal dos mesmos. No caso de papéis-cartão com produção nula em um mês, o produto foi desconsiderado na implementação.

Na prática, observou-se que o número de ciclos efetivamente utilizados em cada instância foi consideravelmente inferior ao valor dado por  $Q$ , a variável utilizada para quantificar o maior número possível de ciclos, de expressão  $Q = \sum_{j \in T} \left\lfloor \frac{d_j}{l_j} \right\rfloor$ . Para todas as instâncias, foram utilizados menos de 10 ciclos, então esse valor foi utilizado como limite, uma vez que a redução no número possível de ciclos implica redução expressiva no número de variáveis do problema.

O valor de lote máximo igual a 5000 toneladas foi adotado também para produtos K iguais, agrupando a produção destinada a retiradas constantes e discretas, em ciclos consecutivos ( $q$  e  $q + 1$ ) – restrições (4.13). Para o mês 3, em particular, foi incluída uma restrição para limitar a produção de cada produto K a 3500 toneladas no ciclo  $q = 1$ , devido a uma sugestão da indústria em função do histórico da máquina.

A Tabela 9 fornece as equações adicionais ao modelo para a implementação das restrições citadas na [Seção 5.2](#) e no texto imediatamente acima.

**Tabela 9** - Formulação das especificidades das instâncias

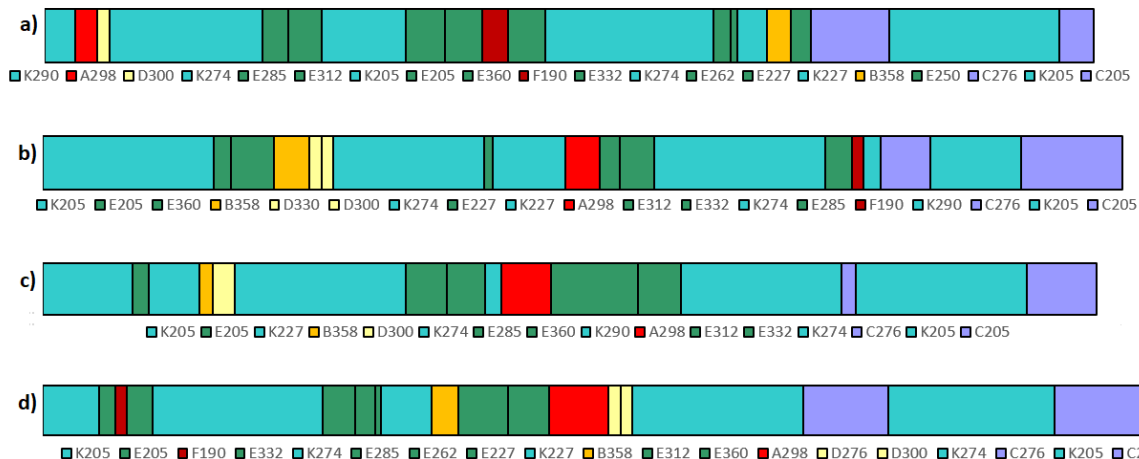
Limite de 3500 toneladas para produtos K no ciclo 1 do mês 3
$a_i t_{i1} + a_j t_{j1} \leq 3500, \quad \forall i \in T, j \in L, i = j$
Papel-cartão D300 obrigatoriamente no primeiro ciclo, no mês 3
$y_{D300,1} = 1$
Trocas vetadas entre K205/C205/F190 e A298/B358
$x_{ij}^q = 0, \forall i \in \{K205_T, K205_L, C205, F190\}, j \in \{A298, B358\}, q = 1 \dots Q$ $x_{jl}^q = 0, \forall j \in \{A298, B358\}, l \in \{K205_L, C205, F190\}, q = 1 \dots Q$ $z_{jm}^q = 0, \forall j \in \{A298, B358\}, m \in \{K205_T\}, q = 1 \dots Q - 1$
Trocas vetadas entre A298 e B358 no mês 4
$x_{A298,B358}^q = 0, q = 1 \dots Q$ $x_{B358,A298}^q = 0, q = 1 \dots Q$
Trocas vetadas entre produtos K de gramaturas diferentes
$x_{ij}^q = 0, \forall i, j \in (T \cup L), i \neq j, q = 1 \dots Q$ $z_{ji}^q = 0, \forall i \in T, j \in L, i \neq j, q = 1 \dots Q - 1$
Limite de 4 produtos E por ciclo nos meses 1, 2 e 4
$\sum_{j \in E} y_{jq} \leq 4, q = 1 \dots Q$
Limite de 3 produtos E em sequência no mês 3
$\sum_{i \in E} \sum_{\substack{j \in E \\ j \neq i}} x_{ij}^q \leq 3, q = 1 \dots Q$

Os resultados para o sequenciamento e dimensionamento dos lotes, para cada mês, são dados na Tabela 10. Para os produtos K205 e K274, a massa total planejada para cada lote é dada fora dos parênteses, e entre parênteses há a massa destinada às retiradas discretas. A massa é dada em toneladas.

Na Figura 11, as mesmas sequências estão representadas graficamente. O comprimento de cada barra é proporcional à dimensão dos lotes, em toneladas, e a legenda é dada na ordem de produção das diferentes espécies de produto.

**Tabela 10 - Sequências obtidas para cada mês pela PIM**

Mês 1		Mês 2		Mês 3		Mês 4	
K290	898	K205	5000 (2666)	K205	2600 (424)	K205	1645 (0)
A298	650	E205	500	E205	500	E205	468
D300	350	E360	1256	K227	1460	F190	345
K274	4502 (0)	B358	1039	B358	400	E332	731
E285	750	D330	351	D300	650	K274	5000 (0)
E312	1000	D300	350	K274	5000 (590)	E285	935
K205	2456 (0)	K274	4400 (0)	E285	1200	E262	598
E205	1147	E227	250	E360	1108	E227	168
E360	1102	K227	2135	K290	500	K227	1461
F190	750	A298	1000	A298	1450	B358	785
E332	1100	E312	600	E312	2529	E312	1464
K274	4960 (498)	E332	1000	E332	1270	E360	1194
E262	500	K274	5000 (590)	K274	4700 (0)	A298	1750
E227	200	E285	771	C276	400	D276	350
K227	854	F190	350	K205	5000 (3617)	D300	350
B358	700	K290	500	C205	2040	K274	5000 (389)
E250	600	C276	1450			C276	2473
C276	2300	K205	2653 (2653)			K205	4882 (1903)
K205	5000 (4417)	C205	2973			C205	3500
C205	1000						
<b>Tempo total de <i>setup</i> (h)</b>							
7,9716		7,7253		6,8576		7,1913	



**Figura 11 - Representação gráfica das sequências obtidas pela PIM: meses 1 a 4 – (a) a (d), respectivamente.**  
 Fonte: elaborada pela autora.

Conforme dito, o menor tempo de *setup* possível foi obtido para as quatro instâncias. Isso não exclui, no entanto, a possibilidade de que haja outras sequências viáveis com o mesmo tempo de *setup* encontrado, dado que há outras possíveis divisões de lotes para os produtos K e que os tempos de *setup* se repetem para diversas trocas.

A Tabela 11 mostra a comparação dos tempos de *setup* obtidos por este modelo com o tempo total de *setup* das sequências fornecidas pela indústria. Para todas as instâncias, foram observadas reduções superiores a 34% no tempo total de *setup*. Para o mês 1, por exemplo, a diferença numérica no tempo de *setup* foi igual a 4,37 horas. Assumindo uma taxa de produção de 45 toneladas por hora, o tempo economizado se traduz na possibilidade de produção de 196 toneladas adicionais de papel-cartão naquele mês.

**Tabela 11** - Comparação das sequências fornecidas pela indústria com os resultados obtidos pela PIM

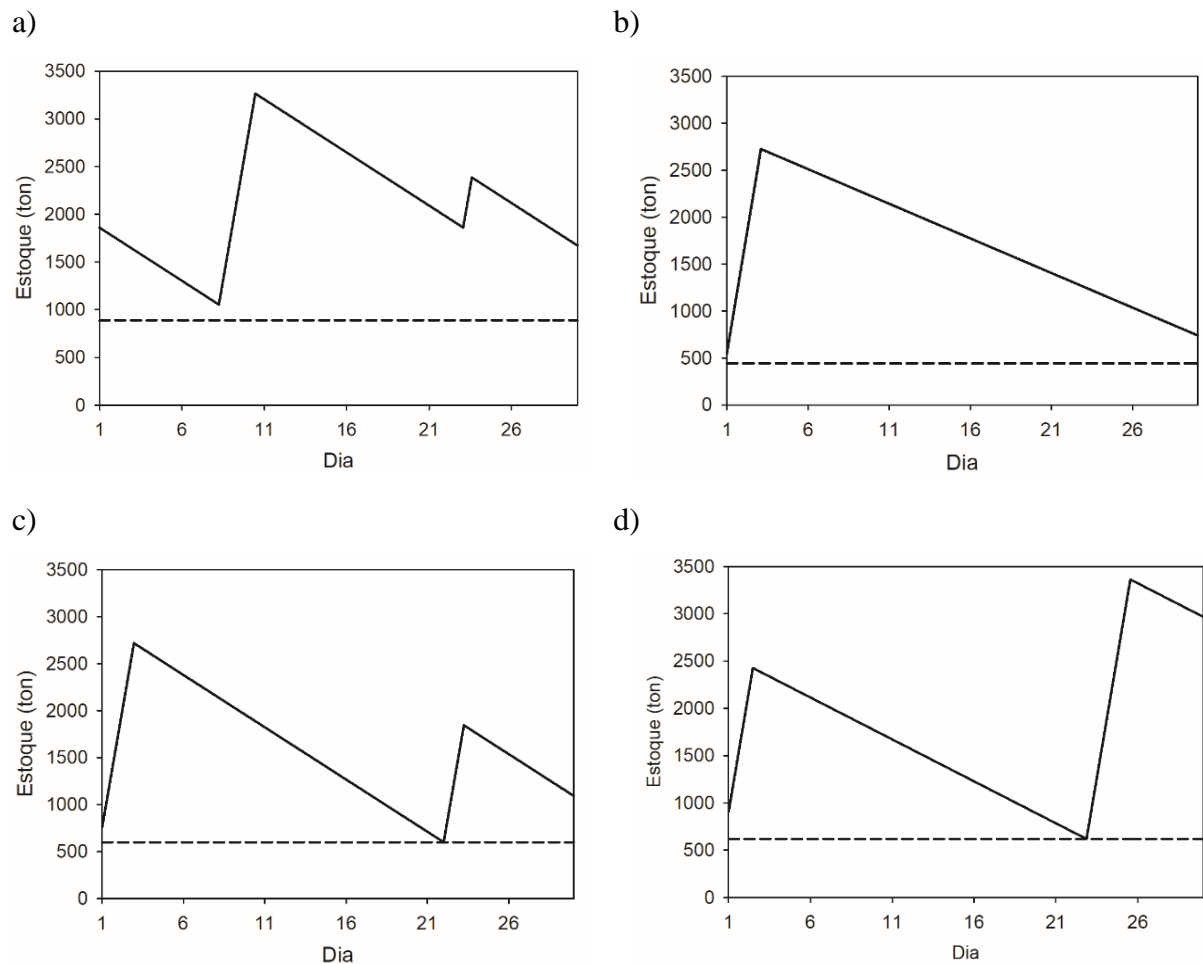
<b>Instância</b>	<b>Tempo de <i>setup</i> da indústria (h)</b>	<b>Tempo de <i>setup</i> do modelo (h)</b>	<b>Redução em relação à sequência da indústria (%)</b>
<b>Mês 1</b>	12,3413	7,9716	35,4
<b>Mês 2</b>	12,5905	7,7253	38,6
<b>Mês 3</b>	10,4947	6,8576	34,7
<b>Mês 4</b>	12,8746	7,1913	44,1

Para se avaliar a aplicabilidade prática do modelo, é necessário considerar, além dos resultados numéricos obtidos, o tempo demandado para a resolução do mesmo. Houve variabilidade considerável dentre as quatro instâncias implementadas, com o tempo variando entre 17 e 154 minutos no computador de processador Intel i7 com memória RAM 16 GB e CPU de 3,60 GHz. Por mais que o tempo seja elevado em comparação com o tempo da heurística, que será apresentado mais adiante, ele é considerado razoável, tendo em vista que o resultado consiste em um planejamento para todo um mês, e este planejamento é realizado com relativa antecedência.

Todavia, cabe mencionar que realizaram-se tentativas de executar os mesmos códigos no computador de memória RAM de 8 GB utilizado para a heurística, porém, devido à limitação na memória (8 GB em comparação com 16 GB da máquina utilizada), não foi possível executar os códigos até que a solução exata fosse encontrada. Sendo assim, destaca-se que a obtenção das sequências ótimas é possível em um computador pessoal, desde que sua memória RAM seja de no mínimo 16 GB.

### **5.3.2. Verificação da manutenção de estoque**

As Figuras 12 e 13 mostram a previsão de estoque para os produtos K205 e K274, respectivamente, ao longo dos dias em cada um dos meses, considerando a taxa de demanda prevista, e o estoque de segurança para os mesmos ([Tabela 5](#)).



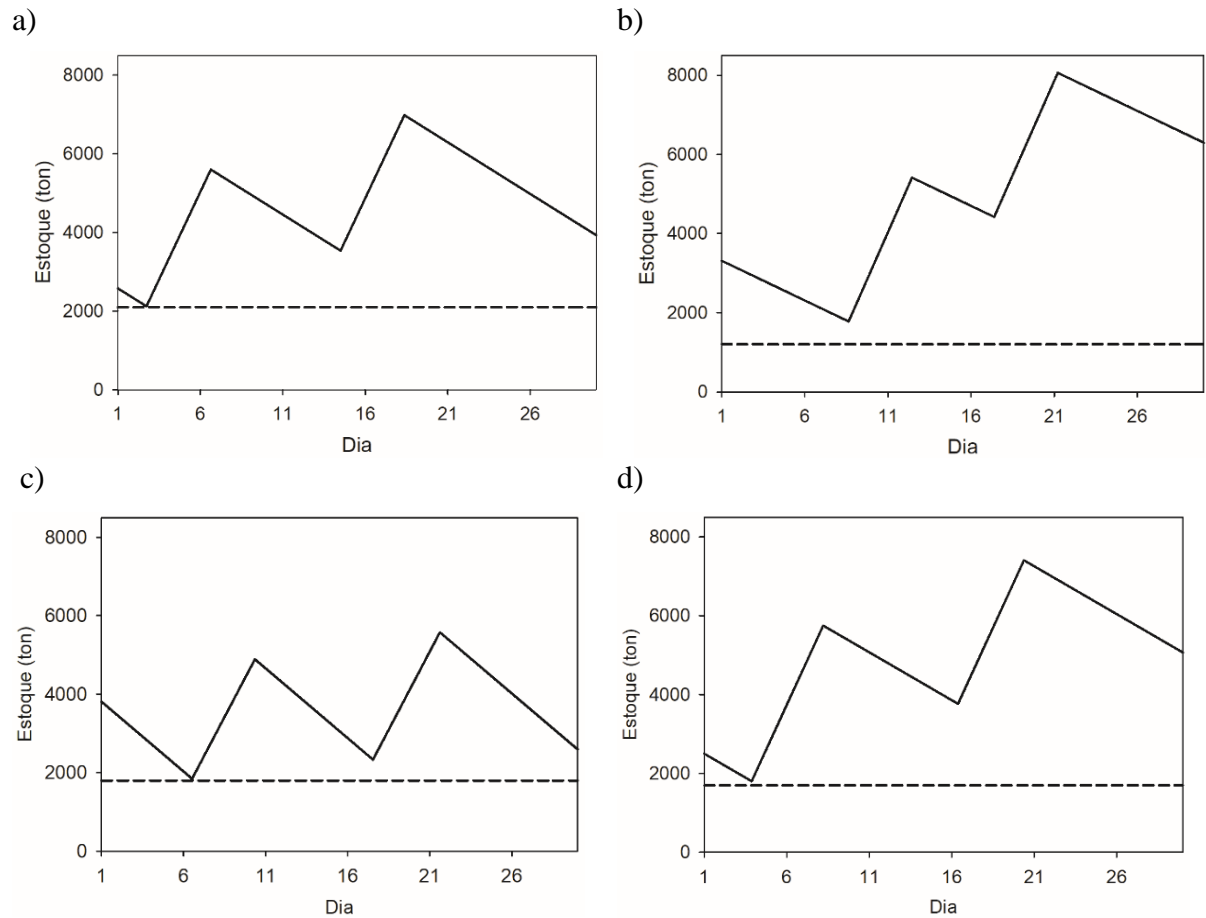
**Figura 12** - Estoque previsto (—) e estoque de segurança (---) dos produtos K205 pela PIM: meses de 1 a 4 – (a) a (d), respectivamente. Fonte: elaborada pela autora.

O estoque de K205 atingiu níveis próximos ao estoque de segurança nos meses 3 e 4. No mês 3, o estoque atingiu exatamente 600 toneladas no dia 22, o valor mínimo permitido. No mês 4, o estoque de segurança também é atingido. Neste caso, o valor é de 620 toneladas, e o estoque atingiu esse nível também no dia 22.

O produto K274, por sua vez, se aproximou do estoque mínimo permitido nos meses 1, 3 e 4. No mês 1, o estoque atingiu o valor mínimo de 2122 toneladas no dia 2, 23 toneladas acima do estoque de segurança. No mês 3, o menor valor atingido foi de 1844 toneladas no dia 6, 44 toneladas acima do permitido para essa instância. Por fim, no mês 4, o estoque de K274 atingiu o nível de 1803 toneladas no dia 3, 103 toneladas a mais que o estoque mínimo permitido.

Observa-se que, como desejado, o estoque previsto, em nenhum momento, atingiu valor inferior ao estoque de segurança, indicando que o sistema de verificação de estoque apenas ao final de cada ciclo  $q$  foi eficaz.





**Figura 13** - Estoque previsto (—) e estoque de segurança (---) dos produtos K274 pela PIM: meses de 1 a 4 – (a) a (d), respectivamente. Fonte: elaborada pela autora.

## 5.4. IMPLEMENTAÇÃO DA HEURÍSTICA DE CONSTRUÇÃO

### 5.4.1. Implementação das variáveis e restrições

A heurística de construção foi executada para as quatro instâncias consideradas. Ao contrário do que foi feito para o modelo inspirado no problema do caixeiro viajante com coleta de prêmios, não houve a separação dos produtos K em porções destinadas a retiradas contínuas ou discretas. Como alternativa, foi criado um vetor contendo a massa destinada às retiradas contínuas, o qual foi atualizado a cada lote alocado desses produtos. Enquanto essa demanda não fosse atingida, as restrições de manutenção de estoque foram verificadas a cada inserção de um novo produto na sequência. Com base na análise dos resultados para o método exato, optou-se por não particionar a demanda dos produtos das categorias de A a F e os produtos K

sem restrição de estoque em mais de um lote, pois tal divisão também não ocorreu no PCVCP. Para os produtos K com restrição de estoque, foi estabelecido o lote máximo de 5000 toneladas, sem valor de lote mínimo.

As trocas não permitidas foram as mesmas do problema do caixeiro viajante com coleta de prêmios. São elas: trocas entre produtos K diferentes, trocas do conjunto  $\{K205, C205, F190\}$  para o conjunto  $\{A298, B358\}$  e vice-versa em todas as instâncias e trocas entre A298 e B358 e vice-versa na instância 4. As restrições envolvendo trocas não permitidas foram implementadas no código por meio da matriz  $f$ , contendo zeros e uns. Foi estabelecido um limite de 3500 toneladas para o produto K alocado na primeira posição na instância do mês 3, em conformidade com a restrição de 3500 toneladas para um mesmo produto K no primeiro ciclo ([Tabela 9](#)).

As restrições que envolvem o conceito dos ciclos, como número de produtos da categoria E por ciclo ou atrelando o produto D300 ao primeiro ciclo na instância do mês 3 ([Tabela 9](#)), bem como limitação de produtos K iguais em ciclos consecutivos (restrição (4.13)), foram implementadas como etapa de verificação nas últimas linhas do código, por meio da análise das sequências geradas. Para isso, foi realizada uma reinterpretação dos ciclos, os quais passaram a ser tratados como blocos iniciados por um produto K (K205, K227, K274 ou K290) e finalizados pelo produto imediatamente anterior ao produto K seguinte. A restrição de lote máximo para produtos K em ciclos adjacentes foi transformada em uma restrição que obrigasse a haver, entre dois lotes de um mesmo produto K, pelo menos um lote de um produto K diferente. A restrição de no máximo 4 produtos E em um mesmo ciclo foi adaptada para no máximo 4 produtos E entre dois produtos K. De maneira semelhante, o produto D300 no primeiro ciclo, para a instância 3, foi adaptado para este produto ter sido alocado antes do segundo lote de algum produto K.

O método heurístico apresenta a importante vantagem de permitir a geração de milhares de sequências viáveis em poucos segundos, tornando possível a escolha do operador por uma dessas sequências de acordo com sua vivência prática e as necessidades da indústria, que nem sempre são eficientemente traduzidas para a linguagem computacional.

#### 5.4.2. Avaliação da dimensão da lista restrita de candidatos

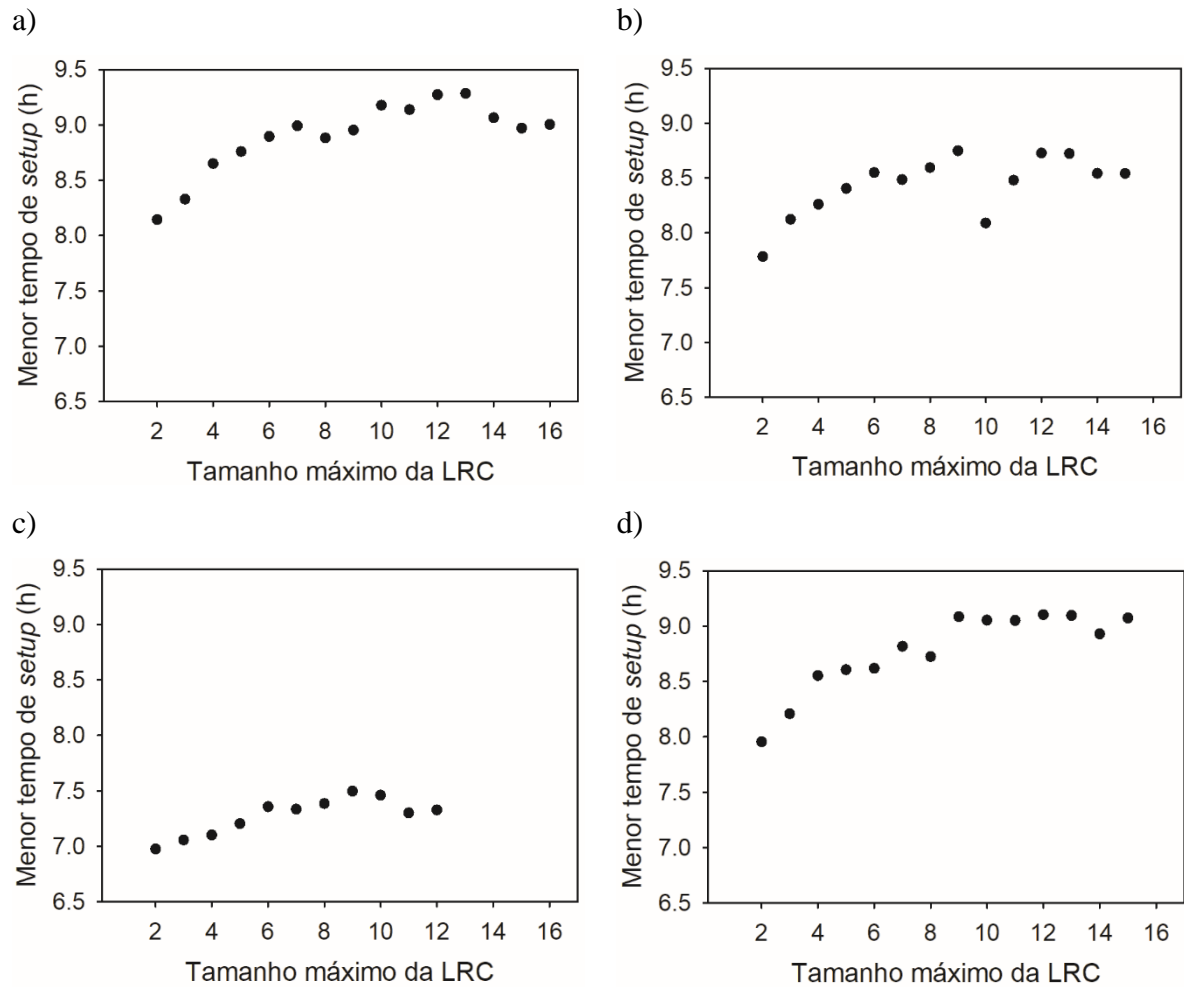
O primeiro experimento a ser realizado possuiu o intuito de avaliar o efeito da dimensão da lista restrita de candidatos (LRC) no resultado obtido. Para as quatro instâncias, o algoritmo foi testado para LRCs contendo de um elemento até o máximo possível de elementos. Para o mês 1, por exemplo, há 18 produtos a serem fabricados, dos quais dois estão sujeitos a restrições de estoque. Alocado o primeiro produto, ele possui no máximo 14 possíveis sucessores, pois são excluídos pelo menos ele próprio e os outros 3 produtos K. Alocado o produto seguinte, ele possui no máximo 16 sucessores, pois estão excluídos ele próprio e o produto K com a maior duração de estoque naquele momento. Dessa forma, o tamanho máximo que a LRC pode assumir é igual ao número de produtos da instância menos dois. Com exceção dos produtos sujeitos à restrição de estoque, cuja demanda é maior, cada produto alocado deixa de ser considerado como sucessor para os elementos seguintes, de modo que a lista de possíveis sucessores tenda a reduzir ao longo da construção da sequência.

A LRC com apenas um elemento corresponde ao algoritmo inteiramente guloso, e a lista com todos os possíveis sucessores corresponde ao algoritmo inteiramente aleatório. Em caso de empate, todos os elementos com o *setup* igual ao valor de corte são incluídos na lista. Para cada tamanho de LRC em cada instância, o algoritmo para geração de sequências foi executado 10 mil vezes.

A Figura 14 mostra, para cada instância, o menor tempo de *setup* obtido para os diferentes tamanhos máximos de LRC permitidos. Nas quatro instâncias, não houve a geração de sequências viáveis para o algoritmo completamente guloso (LRC de tamanho 1), por isso não há a representação desses pontos nos gráficos correspondentes.

Nos quatro meses, o melhor tempo total de *setup* foi obtido para LRCs de tamanho 2. Observou-se tendência de crescimento seguido de estabilidade no menor valor encontrado para a função objetivo conforme o aumento no número de elementos da LRC – com exceção de um ponto para LRC de tamanho 10 no mês 2 –, indicando que é interessante potencializar a característica gulosa do algoritmo, utilizando LRCs com menos elementos.

Conforme dito, o algoritmo foi executado 10 mil vezes para cada dimensão da LRC nas quatro instâncias. O tempo de execução foi de cerca de 1 minuto para cada instância, no computador de processador Intel i5, memória RAM igual a 8 GB e CPU de 2,20 GHz. O tempo reduzido para a resolução, em contraste aos até 154 minutos demandados pelo outro método,



**Figura 14** - Menores valores obtidos para o tempo de *setup* para cada tamanho máximo da LRC: meses de 1 a 4 – (a) a (d), respectivamente. Fonte: elaborada pela autora.

executado em uma máquina com maior capacidade de processamento, destaca a potencialidade dos métodos heurísticos no fornecimento de soluções viáveis com valores bons o bastante para a função objetivo, de maneira prática e rápida.

### 5.4.3. Sequências obtidas e comparação com a PIM

Para fins de comparação com a formulação inspirada no problema do caixeiro viajante com coleta de prêmios, foram tomadas as melhores sequências obtidas neste teste para cada instância. Como exposto anteriormente, todas elas foram obtidas para a LRC com dois elementos. A Tabela 12 mostra essas sequências e o tempo total de *setup* para cada uma delas.

Da mesma forma que nas demais tabelas, a massa total de K205 e K274 para cada lote, em toneladas, é dada fora dos parênteses, e a porção destinada às retiradas discretas é dada entre parênteses.

**Tabela 12** - Sequências de menor tempo de *setup* obtidas no teste para dimensão da LRC

Mês 1		Mês 2		Mês 3		Mês 4	
K274	5000 (0)	K205	5000 (2666)	K205	3500 (0)	K274	3618 (0)
E285	750	E205	500	E205	500	E227	168
A298	650	B358	1039	A298	1450	K205	5000 (376)
E250	600	D330	351	D300	650	E205	468
E262	500	D300	350	K274	5000 (0)	A298	1750
E227	200	K274	5000 (0)	E285	1200	D276	350
K227	854	E227	250	B358	400	D300	350
C205	1000	K227	2135	E312	2529	K274	5000 (0)
K205	5000 (1961)	A298	1000	E360	1108	E285	935
E205	1147	E312	600	K290	500	E262	598
B358	700	E360	1256	C276	400	K227	1461
D300	350	F190	350	K274	4700 (590)	B358	785
K274	4462 (498)	E332	1000	E332	1270	E312	1464
E332	1100	K274	4400 (590)	K205	4100 (4041)	F190	345
E312	1000	E285	771	C205	2040	E360	1194
E360	1102	K205	2653 (2653)	K227	1460	E332	731
F190	750	C205	2973			K274	1382 (389)
K205	2456 (2456)	C276	1450			C276	2473
C276	2300	K290	500			K205	1527 (1527)
K290	898					C205	3500
<b>Tempo total de <i>setup</i> (h)</b>							
8,1441		7,7847		6,9768		7,9565	

A Tabela 13 exibe a comparação do tempo total de *setup* em relação ao método de programação inteira mista.

**Tabela 13** - Comparação dos resultados da heurística com as soluções exatas

Instância	Melhor resultado (h)	Diferença em relação à PIM (%)
Mês 1	8,1441	2,16
Mês 2	7,7847	0,77
Mês 3	6,9768	1,74
Mês 4	7,9565	10,64

Para o meses de 1 a 3, os menores tempos de *setup* encontrados no método heurístico para 10 mil execuções em cada tamanho de LRC foram próximos aos respectivos valores para a PIM. Em contrapartida, uma diferença de 10,6% foi observada no mês 4. Comparando os

resultados para os dois métodos, nota-se que, na sequência gerada pelo problema do caixeiro viajante com coleta de prêmios, houve dois lotes de K205 e dois de K274. Nesta sequência gerada pela heurística de construção, por sua vez, a produção de K274 foi dividida em três lotes. Isso se deve ao K274 ter sido o produto que atingiria o estoque de segurança em menos tempo, então, pela maneira como a programação se realizou, ele foi sempre o primeiro a ser alocado. Em conjunto a isso, o K205 se esgotaria em um cenário em que o lote máximo de K274 (5 mil toneladas) fosse realizado, de modo que houve a necessidade de reduzir o lote de K274. Como a demanda deste foi igual a 10 mil toneladas nessa instância, e o lote máximo é de 5 mil toneladas, foi necessário fazer três lotes. Na [Seção 5.4.6](#), é retratado um teste adicional em que o K205 foi forçado a ocupar a primeira posição, em vez do K274, e nesse teste é possível observar a redução no tempo de *setup*. De todo modo, todos os tempos de *setup* dessas sequências, embora superiores ao tempo encontrado pelo PCVCP, são consideravelmente inferiores ao tempo das sequências fornecidas pela indústria.

Em adição, compararam-se os tempos computacionais deste teste com a execução do método exato. No CPLEX, onde foi implementado o PCVCP, o tempo computacional é fornecido ao final da execução do código. No Matlab, onde foi implementada a heurística, foram usados os comandos “tic” e “toc” que, juntos, fornecem o tempo gasto na execução das linhas de código compreendidas entre esses dois comandos. Os tempos de execução estão listados na Tabela 14. Para a PIM, é mostrado o tempo para apenas uma execução. Para a heurística, foi realizado o cálculo da média aritmética entre o tempo de 5 execuções.

**Tabela 14** - Tempo computacional para os testes realizados

<b>Instância</b>	<b>PIM</b>	<b>Heurística*</b>
Mês 1	154,0 min	67,96 s
Mês 2	81,62 min	58,70 s
Mês 3	16,90 min	38,58 s
Mês 4	28,28 min	58,92 s

\*Tempo médio para 5 execuções

O tempo de execução para o método exato apresentou variação considerável de acordo com a instância. Para o mês 3, o teste demandou cerca de 17 minutos, enquanto, para o mês 1, foram necessárias mais de duas horas e meia para que a sequência ótima fosse obtida e que sua otimalidade fosse verificada. Na heurística, o tempo computacional deste teste, em que o código foi executado 10 mil vezes para cada tamanho de LRC, variou entre 38,58 e 67,96 segundos. Observou-se uma certa proporcionalidade entre o tamanho máximo possível da LRC e o tempo

de execução. O mês 3, cujo tamanho máximo de LRC foi igual a 12, foi o teste mais rápido; em contraste, o mês 1, para o qual foram testadas listas restritas com até 16 elementos, foi o teste que demandou mais tempo.

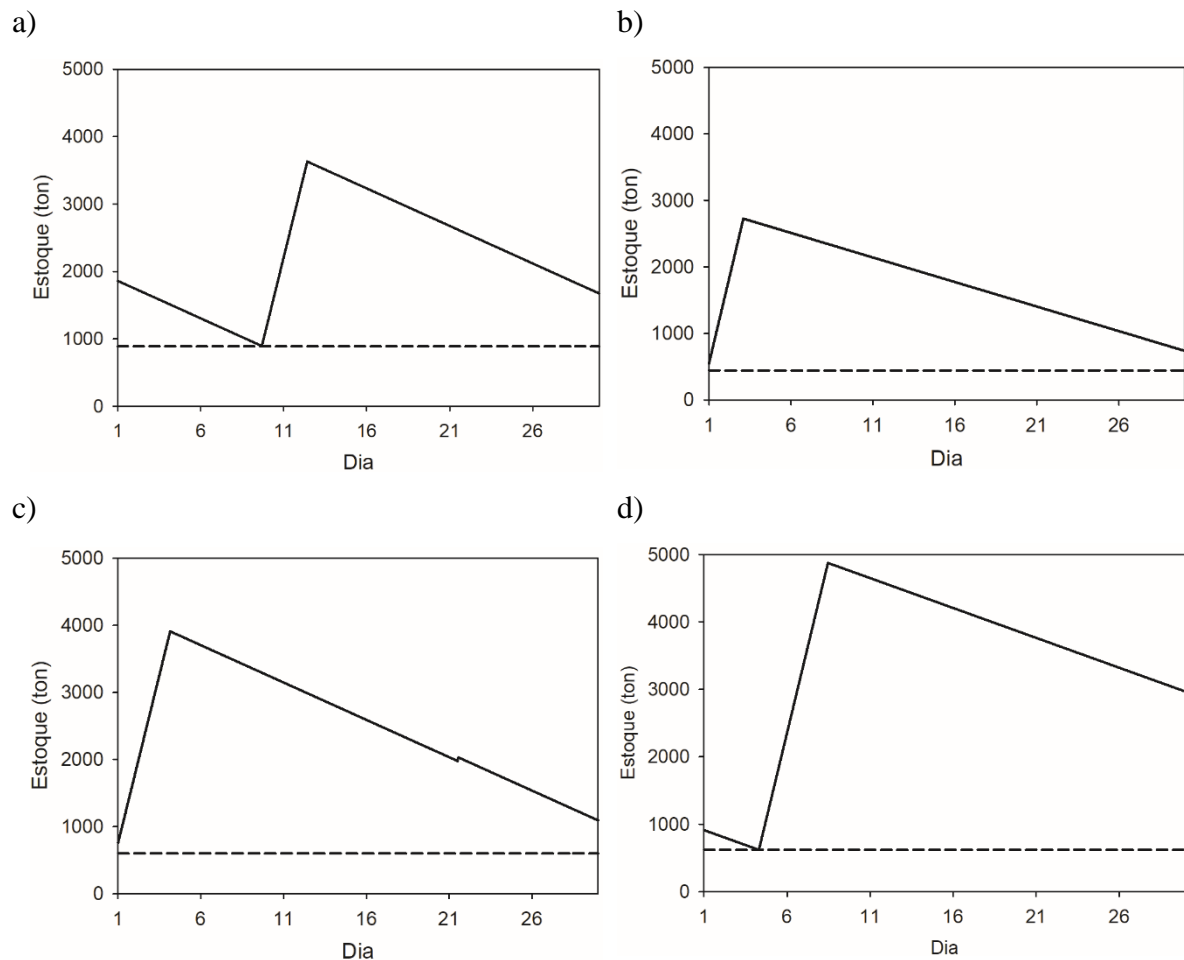
#### 5.4.4. Verificação da manutenção de estoque

As sequências elencadas na [Tabela 12](#) foram utilizadas para ilustrar a eficácia do modelo em garantir a manutenção do estoque de segurança, o que foi feito por meio da verificação do estoque de K205 e K274 a cada produto inserido enquanto houvesse demanda destinada a retiradas contínuas por cumprir. As Figuras 15 e 16 mostram, respectivamente, o nível de estoque para os produtos K205 e K274, bem como o estoque de segurança exigido para cada produto em cada mês.

Os gráficos mostram que o estoque de segurança não foi violado em momento algum para os dois produtos nas quatro instâncias. No mês 1, o estoque de K205 atingiu o menor valor do período no dia 9, correspondente a 894 toneladas, quatro a mais que o estoque de segurança. Em termos de programação, isso indica que, nesse momento, o algoritmo retornou para as linhas iniciais, pois a condição  $\min(tl > 0) < \min(td)$  (linha 33) foi violada, visto que não seria possível alocar qualquer lote antes que o nível de estoque atingisse 890 toneladas.

No mês 4, o estoque do K205 atingiu exatamente o mínimo permitido, de 620 toneladas, no dia 4. Nessa instância, foi necessário abreviar o primeiro lote do K274 para que houvesse tempo de alocar outro produto antes que o K205 se esgotasse. Em vez das 5000 toneladas possíveis, foi confeccionado um lote de 3618 toneladas de K274, valor ajustado para que o estoque de K205 atingisse exatamente o estoque mínimo de segurança.

Quanto ao K274, no mês 3, o estoque atingiu o menor valor do período – 1861 toneladas – no dia 6, e o estoque de segurança era igual a 1800 toneladas (Figura 16 (c)). Nessa situação, a exemplo do que houve para o K205 no mês 1, o algoritmo obrigatoriamente posicionou o produto K274 na sequência.

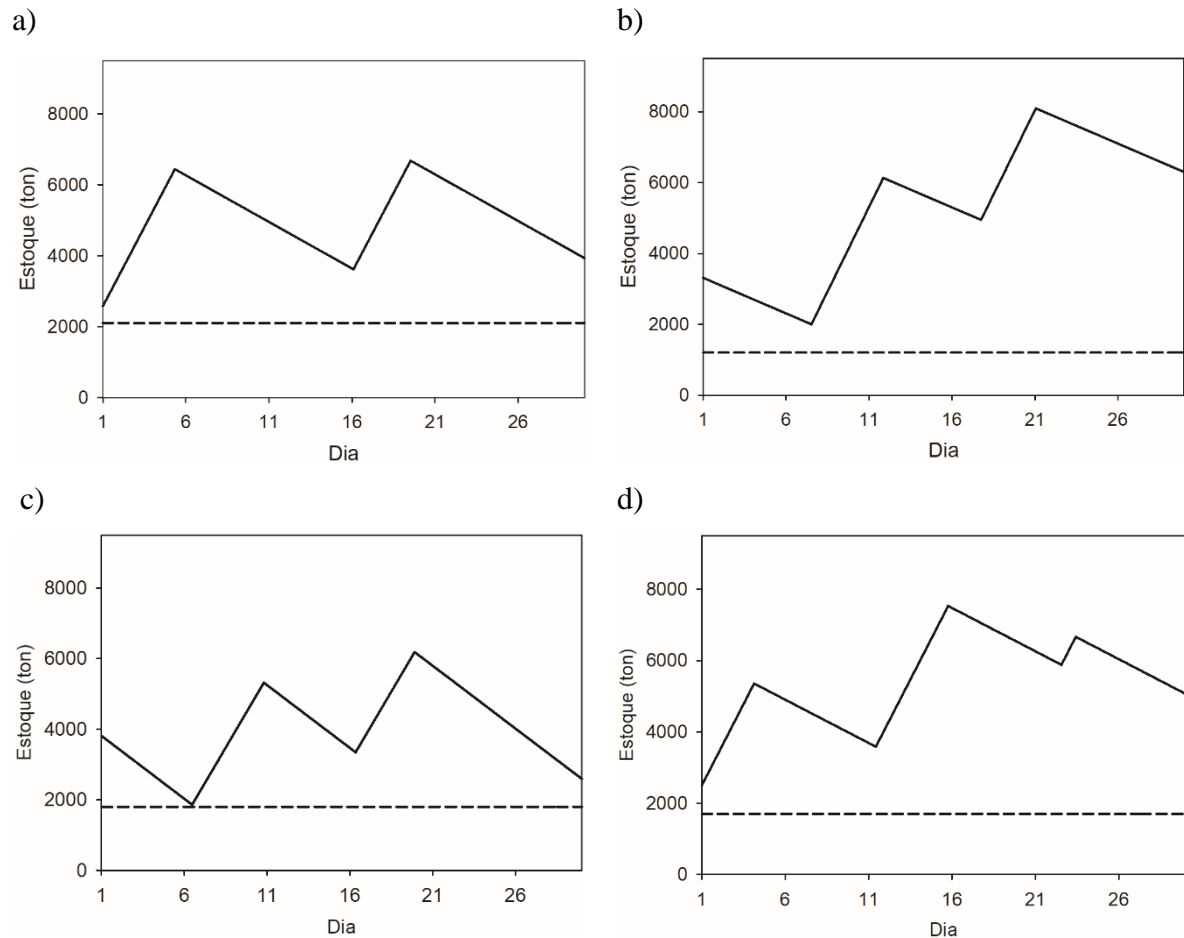


**Figura 15** - Estoque previsto (—) e estoque de segurança (---) dos produtos K205 pela heurística: meses de 1 a 4 – (a) a (d), respectivamente. Fonte: elaborada pela autora.

A respeito da produção, a demanda do K205 destinada às retiradas constantes foi planejada para apenas um lote nos meses 1, 2 e 4 – um trecho ascendente no gráfico. No mês 3, houve um lote de 3500 toneladas no início da sequência e um pequeno lote de 59 toneladas no dia 21. Como a demanda do K205 para retiradas constantes correspondia a 3559 toneladas, e o lote máximo permitido para a primeira alocação foi fixado em 3500 toneladas, houve a necessidade dessa divisão.

O K274, por sua vez, foi planejado para dois lotes nos meses 1, 2 e 3 e para três lotes no mês 4. A diferença do mês 4 se deve à necessidade de encurtamento do primeiro lote para que o estoque do K205 respeitasse o limite mínimo, de modo que houve a necessidade de dividir a demanda de K274 em três lotes.





**Figura 16** - Estoque previsto (—) e estoque de segurança (---) dos produtos K274 pela heurística: meses de 1 a 4 – (a) a (d), respectivamente. Fonte: elaborada pela autora.

#### 5.4.5. Obtenção de sequências alternativas

Uma das ditas vantagens da heurística é a obtenção de diversas sequências em um tempo computacional reduzido. Para a geração de *pools* contendo múltiplas soluções viáveis, o código foi executado 20 mil vezes para cada instância, utilizando-se a LRC de tamanho 2, a qual apresentou os melhores resultados para o tempo de *setup*. Optou-se por aumentar o número de execuções de 10 para 20 mil porque, como apenas a LRC de tamanho 2 foi utilizada nesses testes, o tempo de execução do código foi reduzido, de maneira que a construção de cada *pool* demandou um tempo inferior a 10 segundos. Vale salientar que é possível a obtenção de tempos de *setup* inferiores aos das sequências mostradas na [Tabela 12](#), devido à natureza aleatória do algoritmo.

Em cada execução, foram construídos *pools* com as 20 sequências de menor tempo de *setup*, dentre as 20 mil execuções. Em todas as tabelas, a dimensão do lote é dada em toneladas.

A sequência da esquerda corresponde ao menor tempo de *setup* presente em cada *pool*, e, para a sequência da direita, escolheu-se uma dentre as demais opções que apresentassem maior variabilidade em relação à da esquerda.

- **Pool de soluções para o mês 1**

Duas das sequências obtidas para o mês 1 são dadas na Tabela 15.

**Tabela 15** - Sequências obtidas no *pool* para o mês 1

K274	5000 (0)	K274	5000 (0)
E285	750	E262	500
E262	500	E227	200
E227	200	K227	854
K227	854	C205	1000
C205	1000	K205	5000 (1961)
K205	5000 (1961)	E205	1147
E205	1147	B358	700
A298	650	D300	350
E250	600	K274	4462 (498)
F190	750	E285	750
E360	1102	E360	1102
E312	1000	E312	1000
B358	700	E332	1100
D300	350	F190	750
K274	4462 (498)	K205	2456 (2456)
E332	1100	C276	2300
K290	898	K290	898
C276	2300	A298	650
K205	2456 (2456)	E250	600
<b>Setup (h)</b>	8,1294	<b>Setup (h)</b>	8,2224

A sequência da esquerda apresenta um tempo total de *setup* igual a 8,1294, inferior ao tempo de *setup* da sequência mostrada na [Tabela 12](#). A diferença desta em relação ao tempo obtido pelo PCVCP é de 1,98%. A da direita, por sua vez, possui um tempo de *setup* 3,15% maior que o do método exato.

- **Pool de soluções para o mês 2**

A Tabela 16 ilustra duas sequências obtidas no *pool* para a instância do mês 2.

**Tabela 16** - Sequências obtidas no *pool* para o mês 2

K205	5000 (2666)	K205	5000 (2666)
E205	500	E205	500
A298	1000	B358	1039
D330	351	D330	351
D300	350	D300	350
K274	5000 (0)	K274	5000 (0)
E227	250	E227	250
K227	2135	K227	2135
C205	2973	A298	1000
K290	500	E312	600
B358	1039	E332	1000
E312	600	F190	350
E332	100	E360	1256
K274	4400 (590)	E285	771
E285	771	K274	4400 (590)
F190	350	C276	1450
E360	1256	K205	2653 (2653)
K205	2653 (2653)	C205	2973
C276	1450	K290	500
<b>Setup (h)</b>	7,7324	<b>Setup (h)</b>	7,8561

A sequência da esquerda, a exemplo da instância do mês 1, também apresenta um tempo de *setup* inferior ao da sequência representada na [Tabela 12](#). A diferença do tempo total de *setup* desta para a sequência obtida pelo problema do caixeiro viajante com prêmios é de apenas 0,09%, em que o tempo total destinado às trocas é igual a 7,7253 horas. Quanto à sequência da direita, a diferença em relação ao método exato é de 1,69%.

- **Pool de soluções para o mês 3**

Na Tabela 17, são dadas duas sequências obtidas no *pool* de soluções para a instância do mês 3.

**Tabela 17** - Sequências obtidas no *pool* para o mês 3

K205	3500 (0)		K205	3500 (0)
E205	500		E205	500
A298	1450		A298	1450
D300	650		D300	650
K274	5000 (0)		K274	5000 (0)
E285	1200		E285	1200
B358	400		B358	400
E312	2529		C276	400
E360	1108		K227	1460
K227	1460		C205	2040
E332	1270		K274	4700 (590)
K274	4700 (590)		E332	1270
C276	400		E312	2529
K205	4100 (4041)		K205	4100 (4041)
C205	2040		E360	1108
K290	500		K290	500
<b>Setup (h)</b>	6,9166		<b>Setup (h)</b>	7,0486

No *pool* do mês 3, foram obtidas 6 sequências com o tempo de *setup* igual a 6,9166 horas, que foi o menor tempo desse grupo. Isso é possível porque, na tabela dos tempos de *setup*, há diversos pares de produtos com tempo igual de troca, então sequências com variadas conformações podem possuir o mesmo tempo total de *setup*.

O tempo de *setup* da sequência da esquerda foi 0,86% maior que o tempo da sequência obtida pelo PCVCP, e a da direita apresentou uma diferença de 2,79%.

- **Pool de soluções para o mês 4**

Por fim, na Tabela 18, são dadas duas sequências obtidas no *pool* para o mês 4.

Neste *pool*, ao contrário dos outros, não foram obtidas sequências com tempo de *setup* inferior ao da sequência mostrada na [Tabela 12](#). Novamente, esse resultado passa pela aleatoriedade do método e poderia ter sido diferente para outras construções de *pools*. A sequência da esquerda possui tempo total de trocas 12,1% superior ao encontrado no método exato, e a da direita apresenta uma diferença de 13,3%.

Esta instância apresentou os maiores desvios em relação ao resultado obtido pelo método do caixeiro viajante com coleta de prêmios, uma vez que foram necessários três lotes

de K274 em vez de dois, como no método exato. Na próxima seção, é realizado um teste adicional para esta instância, em que o K205 é fixado na primeira posição, e o dimensionamento deste lote é predefinido.

**Tabela 18** - Sequências obtidas no *pool* para o mês 4

K274	3618 (0)		K274	3618 (0)
E227	168		E227	168
K205	5000 (376)		K205	5000 (376)
E205	468		E205	468
B358	785		B358	785
D276	350		D300	350
D300	350		K274	5000 (0)
K274	5000 (0)		E285	935
E285	935		E262	598
E262	598		K227	1461
K227	1461		C205	3500
E332	731		K274	1382 (389)
K274	1382 (389)		E332	731
F190	345		F190	345
E312	1464		E360	1194
E360	1194		E312	1464
A298	1750		A298	1750
C276	2473		D276	350
K205	1527 (1527)		K205	1527 (1527)
C205	3500		C276	2473
Setup (h)	8,0635		Setup (h)	8,1453

#### 5.4.6. Teste adicional para a instância do mês 4

Como destacado anteriormente, no mês 4, em função de como o algoritmo foi elaborado, houve a divisão do K274 em três lotes, em vez de dois, o que levou uma maior diferença no tempo de *setup* em relação ao método exato.

Visando a obtenção de sequências com menor tempo de *setup*, foi realizado um teste adicional nesta instância, em que o K205 foi fixado como primeiro produto, e seu lote máximo foi limitado a 1527 toneladas. Como a demanda total para esse produto é de 6527 toneladas, torna-se possível que as 5 mil toneladas restantes sejam produzidas em apenas um lote, alocado em um momento posterior no planejamento.

Realizada esta adaptação, foi construído um *pool* de soluções como os anteriores, selecionando-se as vinte sequências com menor tempo de *setup* a partir de 20 mil execuções do código. Duas das sequências obtidas são mostradas na Tabela 19.

**Tabela 19** - Sequências obtidas para o *pool* no mês 4, fixando-se o primeiro produto

K205	1527 (0)	K205	1527 (0)
D276	350	E205	468
D300	350	B358	785
K274	5000 (0)	D276	350
E262	598	D300	350
E227	168	K274	5000 (0)
K227	1461	E285	935
C205	3500	E262	598
K205	5000 (1903)	E227	168
E205	468	K227	1461
A298	1750	E360	1194
E312	1464	E332	731
E332	731	F190	345
K274	5000 (389)	K205	5000 (1903)
E285	935	E312	1464
F190	345	A298	1750
E360	1194	C276	2473
B358	785	K274	5000 (389)
C276	2473	C205	3500
<b>Setup (h)</b>	7,3110	<b>Setup (h)</b>	7,6389

A sequência da esquerda é a de menor tempo de *setup* dentre as contidas no *pool*, com a diferença de 1,66% em relação ao da sequência obtida pelo problema do caixeiro viajante com coleta de prêmios. A da direita, por sua vez, apresenta uma diferença de 6,22%. Ambas apresentam tempo de *setup* inferior à sequência mostrada para esta instância na [Tabela 12](#) (7,9565 h) e inferior às do *pool*, na [Tabela 18](#) (8,0635 h e 8,1453 h). Esse resultado indica que a alocação preferencial do produto que atinge o estoque de segurança em menos tempo, embora eficiente do ponto de vista da manutenção de estoque mínimo, pode caminhar na contramão da redução do tempo total de *setup*. Sendo assim, sugere-se, para trabalhos futuros, a testagem de rotinas que coloquem como alternativas a inclusão de qualquer um dos produtos sujeitos às restrições de estoque, diferentemente do que foi feito neste trabalho, em que apenas o próximo produto a atingir o estoque mínimo poderia ser alocado.

## 6. CONSIDERAÇÕES FINAIS

O objetivo deste trabalho foi a construção e execução de rotinas de otimização para a minimização do tempo total de *setup* na indústria de papel-cartão, partindo de instâncias reais de uma indústria de papel para embalagens. O problema combina sequenciamento da produção com restrições relativas a manutenção de estoque.

Duas metodologias foram desenvolvidas para a geração de planos viáveis de produção. A primeira delas consiste em uma adaptação do problema do caixeiro viajante com coleta de prêmios, e a resolução foi realizada no *software* CPLEX, fornecendo o ponto ótimo exato. A segunda delas consiste em uma heurística de construção gulosa, aleatorizada e adaptativa, baseada no método GRASP, e fornece soluções viáveis, porém não necessariamente o ponto ótimo. Esta foi implementada no *software* Matlab.

O método exato apresenta a vantagem de fornecer a solução ótima para o problema, enquanto o método heurístico tem como mérito o tempo computacional reduzido para fornecer múltiplas soluções viáveis. Neste trabalho, o método exato demandou até 154 minutos para convergir para o ponto ótimo em uma das instâncias, em computador com processador Intel i7, 16 GB de memória RAM e CPU de 3,60 GHz. O método heurístico, por sua vez, demandou tempo inferior a 10 segundos para 20 mil execuções, em todas as instâncias, em uma máquina com processador Intel i5, 8 GB de memória RAM e CPU de 2,20 GHz. Quanto ao resultado numérico para o tempo de *setup*, para a heurística, foram observadas diferenças entre 0,77% e 10,6% em relação ao método exato, para os melhores resultados do primeiro teste realizado. Nos testes adicionais, foi possível encontrar desvios ainda inferiores, como 0,09% para uma das instâncias.

Em conclusão, ambos os métodos mostraram-se eficientes para a obtenção de planejamentos viáveis para a produção, minimizando o tempo total de *setup*. O método exato é capaz de fornecer a sequência de menor tempo de *setup* possível em um tempo computacional considerado hábil, dado que trata-se de um planejamento realizado com antecedência para ser executado ao longo de todo um mês. O método heurístico, embora haja considerável probabilidade de que o ótimo não seja atingido, fornece um conjunto de soluções viáveis em poucos segundos e permite ao operador a escolha por uma dentre elas, por motivos adicionais ao tempo de *setup*, como melhor adequação às demandas.

## 6.1. CONTRIBUIÇÕES DO TRABALHO

- Resolução de um problema industrial, proporcionando a aplicação de métodos de otimização, amplamente estudados no ambiente acadêmico, à realidade fabril.
- Resolução do problema de sequenciamento da produção com condições adicionais, como restrição de determinadas trocas e manutenção de estoque para alguns dos produtos.
- Desenvolvimento de duas abordagens de solução: uma exata, baseada no problema do caixeiro viajante com coleta de prêmios, e uma heurística, inspirada na etapa de construção da meta-heurística GRASP.

## 6.2. SUGESTÕES PARA TRABALHOS FUTUROS

- **Aprimoramento da heurística de construção:** possibilitar que as sequências se iniciem por qualquer um dos produtos com restrição de estoque, e não apenas pelo produto que atingirá primeiro o estoque de segurança.
- **Implementação de um método de busca local após a etapa de construção:** com a implementação completa do GRASP, torna-se possível encontrar valores menores para o tempo total de *setup*.
- **Geração de instâncias contendo maior número de elementos:** neste trabalho, foram resolvidas as instâncias fornecidas pela indústria, com até dezoito produtos, dos quais dois são consumidos de maneira constante. Para melhor avaliação do método heurístico, sugere-se a geração de instâncias contendo número de elementos com variadas ordens de grandeza.
- **Implementação de outros métodos heurísticos:** métodos como o *relax-and-fix* e *fix-and-optimize* são comumente utilizados em problemas de dimensionamento e sequenciamento de lotes.
- **Adição de máquinas paralelas:** neste trabalho, foi considerada apenas uma máquina, porém diversas indústrias contam com máquinas paralelas, idênticas ou não.



## REFERÊNCIAS BIBLIOGRÁFICAS

- ALFARO-FERNÁNDEZ, P.; RUIZ, R.; PAGNOZZI, F.; STÜTZLE, T. Automatic algorithm design for hybrid flowshop scheduling problems. **European Journal of Operational Research**, v. 282, n. 3, p. 835–845, 2020.
- ATTWOOD, B. W. Paperboard. In: YAM, K. (Ed.). **The Wiley Encyclopedia of Packaging Technology** (pp. 913-920). 3rd. ed. Hoboken/NJ: John Wiley & Sons, 2009.
- AVALOS-ROSALES, O.; ANGEL-BELLO, F.; ALVAREZ, A. Efficient metaheuristic algorithm and re-formulations for the unrelated parallel machine scheduling problem with sequence and machine-dependent setup times. **International Journal of Advanced Manufacturing Technology**, v. 76, p. 1705–1718, 2015.
- BAHL, H. C.; RITZMAN, L. P.; GUPTA, J. N. D. Determining lot sizes and resource requirements: a review. **Operations Research**, v. 35, n. 3, p. 329–345, 1987.
- BALAS, E. The prize collecting traveling salesman problem. **Networks**, v. 19, p. 621–636, 1989.
- CAPO, P. (Ed.) **Guia ABTCP de Fornecedores & Fabricantes: celulose e papel 2018/2019**. São Paulo: ABTCP – Associação Brasileira Técnica de Celulose e Papel, 2018. Disponível em: <[http://www.guiacomprascelulosepapel.org.br/publicador/edicoes\\_impresas/12.pdf](http://www.guiacomprascelulosepapel.org.br/publicador/edicoes_impresas/12.pdf)>. Acesso em 1 abr. 2021.
- CAPO, P. (Ed.) **Guia ABTCP de Fornecedores & Fabricantes: celulose e papel 2020/2021**. São Paulo: ABTCP – Associação Brasileira Técnica de Celulose e Papel, 2020. Disponível em: <[http://www.guiacomprascelulosepapel.org.br/publicador/edicoes\\_impresas/17.pdf](http://www.guiacomprascelulosepapel.org.br/publicador/edicoes_impresas/17.pdf)>. Acesso em 3 mai. 2021.
- CHARITOPOULOS, V. M.; DUA, V.; PAPAGEORGIOU, L. G. Traveling salesman problem-based integration of planning, scheduling, and optimal control for continuous processes. **Industrial and Engineering Chemistry Research**, v. 56, n. 39, p. 11186–11205, 2017.
- CLAUS, A. A new formulation for the travelling salesman problem. **SIAM Journal on Algebraic Discrete Methods**, v. 5, n. 1, p. 21–25, 1984.
- COPIE, K.; WÖRBELAUER, M.; MEYER, H.; TEMPELMEIER, H. Simultaneous lotsizing and scheduling problems: a classification and review of models. **OR Spectrum**, v. 39, p. 1–64, 2017.
- DANTZIG, G.; FULKERSON, R.; JOHNSON, S. Solution of a large-scale Traveling-Salesman Problem. **Journal of the Operations Research Society of America**, v. 2, n. 4, p. 393–410, 1954.
- DREXL, A.; HAASE, K. Proportional lotsizing and scheduling. **International Journal of Production Economics**, v. 40, n. 1, p. 73–87, 1995.

- DREXL, A.; KIMMS, A. Lot sizing and scheduling - survey and extensions. **European Journal of Operational Research**, v. 99, n. 2, p. 221–235, 1997.
- FEO, T. A.; RESENDE, M. G. C. A probabilistic heuristic for a computationally difficult set covering problem. **Operations Research Letters**, v. 8, n. 2, p. 67–71, 1989.
- FERREIRA, M.; CAMARGO, V. C. B.; ARAUJO, S. A. The reel allocation problem of a corrugated packaging company. **Pesquisa Operacional**, v. 40, n. e229708, p. 1-21, 2020.
- FIGUEIRA, G.; AMORIM, P.; GUIMARÃES, L.; AMORIM-LOPES, M.; NEVES-MOREIRA, F.; ALMADA-LOBO, B. A decision support system for the operational production planning and scheduling of an integrated pulp and paper mill. **Computers and Chemical Engineering**, v. 77, p. 85–104, 2015.
- FLEISCHMANN, B. The discrete lot-sizing and scheduling problem. **European Journal of Operational Research**, v. 44, n. 3, p. 337–348, 1990.
- FLEISCHMANN, B.; MEYR, H. The general lotsizing and scheduling problem. **OR Spectrum**, v. 19, n. 1, p. 11–21, 1997.
- FURLAN, M., ALMADA-LOBO, B.; SANTOS, M.; MORABITO, R. Unequal individual genetic algorithm with intelligent diversification for the lot-scheduling problem in integrated mills using multiple-paper machines. **Computers and Operations Research**, v. 59, p. 33–50, 2015.
- GAVISH, B.; GRAVES, S. C. **The travelling salesman problem and related problems** [Working paper n. OR 078-78]. Operations Research Center in Massachusetts Institute of Technology. Cambridge, MA, 1978.
- GENDREAU, M.; POTVIN, J. (Eds.). **Handbook of Metaheuristics**. 3rd. ed. Cham (Suíça): Springer International Publishing, 2019.
- GLOCK, C. H. The joint economic lot size problem: A review. **International Journal of Production Economics**, v. 135, n. 2, p. 671–686, 2012.
- GLOCK, C. H.; GROSSE, E. H.; RIES, J. M. The lot sizing problem: A tertiary study. **International Journal of Production Economics**, v. 155, p. 39–51, 2014.
- GROSSMANN, I. Enterprise-wide optimization: A new frontier in process systems engineering. **AIChE Journal**, v. 51, n. 7, p. 1846–1857, 2005.
- GUIMARÃES, I.; OUAZENE, Y.; SOUZA, M. C.; YALAOUI, F. Flowshop scheduling problem with parallel semi-lines and final synchronization operation. **Computers and Operations Research**, v. 108, p. 121–133, 2019.
- GUIMARÃES, L.; KLABJAN, D.; ALMADA-LOBO, B. Modeling lotsizing and scheduling problems with sequence dependent setups. **European Journal of Operational Research**, v. 239, n. 3, p. 644–662, 2014.

HAASE, K. Capacitated lot-sizing with sequence dependent setup costs. **OR Spectrum**, v. 18, n. 1, p. 51–59, 1996.

HARJUNKOSKI, I.; MARAVELIAS, C. T.; BONGERS, P.; CASTRO, P. M.; ENGELL, S.; GROSSMANN, I. E.; HOOKER, J.; MÉNDEZ, C.; SAND, G.; WASSICK, J. Scope for industrial applications of production scheduling models and solution methods. **Computers and Chemical Engineering**, v. 62, p. 161–193, 2014.

HORN, M.; MASCHLER, J.; RAIDL, G. R.; RÖNNBERG, E. A\*-based construction of decision diagrams for a prize-collecting scheduling problem. **Computers and Operations Research**, v. 126, paper 105125, 2021.

IBÁ. **Relatório Anual 2020**. São Paulo: IBÁ – Indústria Brasileira de Árvores, 2020. Disponível em: <<https://iba.org/datafiles/publicacoes/relatorios/relatorio-iba-2020.pdf>>. Acesso em 1 abr. 2021.

KARMAKAR, U.; SCHRAGE, L. The deterministic dynamic product cycling problem. **Operations Research**, v. 33, n. 2, p. 326–345, 1985.

KIRSCHSTEIN, T. Planning of multi-product pipelines by economic lot scheduling models. **European Journal of Operational Research**, v. 264, n. 1, p. 327–339, 2018.

LE HESRAN, C.; LADIER, A.; BOTTA-GENOULAZ, V.; LAFOREST, V. Operations scheduling for waste minimization: A review. **Journal of Cleaner Production**, v. 206, p. 211–226, 2019.

LOPEZ, M. A. N.; KINGSMAN, B. G. The economic lot scheduling problem: theory and practice. **International Journal of Production Economics**, v. 23, p. 147–164, 1991.

MANNE, A. S. Programming of economic lot sizes. **Management Science**, v. 4, n. 2, p. 115–135, 1958.

MARTÍNEZ, K. P.; MORABITO, R.; TOSO, E. A. V. A coupled process configuration, lot-sizing and scheduling model for production planning in the molded pulp industry. **International Journal of Production Economics**, v. 204, p. 227–243, 2018.

MELEGA, G. M.; ARAUJO, S. A.; JANS, R. Classification and literature review of integrated lot-sizing and cutting stock problems. **European Journal of Operational Research**, v. 271, n. 1, p. 1–19, 2018.

MILLER, C. E.; TUCKER, A. W.; ZEMLIN, R. A. Integer programming formulation of traveling salesman problems. **Journal of the ACM (JACM)**, v. 7, n. 4, p. 326–329, 1960.

NIQUINI, G. R.; SOUZA, M. C.; OLIVEIRA, C. C.; VIEIRA, O.; COSTA JUNIOR, E. F. Minimização do tempo total de *setup* entre lotes em uma máquina de papel-cartão. Artigo submetido para publicação, 2020.

PANTUZA JUNIOR, G.; SOUZA, M. C. Formulations and a Lagrangian relaxation approach for the prize collecting traveling salesman problem. Artigo submetido para publicação, 2020.

- PINEDO, M. L. **Scheduling: theory, algorithms, and systems**. New York: Springer, 2016.
- POCHET, Y.; WOLSEY, L. A. **Production Planning by Mixed Integer Programming**. New York: Springer, 2006.
- RESENDE, M. G. C.; RIBEIRO, C. C. Greedy randomized adaptive search procedures: advances and extensions. In: GENDREAU, M.; POTVIN, J. (Eds.). **Handbook of Metaheuristics**. 3rd. ed. Cham (Suíça): Springer International Publishing, 2019.
- ROBERTI, R.; TOTH, P. Models and algorithms for the Asymmetric Traveling Salesman Problem: an experimental comparison. **EURO Journal on Transportation and Logistics**, v. 1, p. 113–133, 2012.
- ROBERTSON, G. L. **Food packaging: principles and practice**. 3rd. ed. Boca Raton/FL: CRC Press, 2013.
- ROGERS, J. A Computational Approach to the Economic Lot Scheduling Problem. **Management Science**, v. 4, n. 3, p. 264–291, 1958.
- SANTIAGO, S. B. *et al.* Aumento da produção e eficiência operacional na área de preparo de cavacos, através de novas ferramentas de controle da picagem. **O Papel**, v. 80, n. 05, p. 92–97, 2019.
- SANTOS, H.; TOFFOLO, T. A. M.; SILVA, C. L. T. F.; VANDEN BERGHE, G. Analysis of stochastic local search methods for the unrelated parallel machine scheduling problem. **International Transactions in Operational Research**, v. 26, n. 2, p. 707–724, 2019.
- SANTOS, M. O.; ALMADA-LOBO, B. Integrated pulp and paper mill planning and scheduling. **Computers and Industrial Engineering**, v. 63, n. 1, p. 1–12, 2012.
- SILVA, D. A. L.; PAVAN, A. L. R.; OLIVEIRA, J. A.; OMETTO, A. O. Life cycle assessment of offset paper production in Brazil: Hotspots and cleaner production alternatives. **Journal of Cleaner Production**, v. 93, p. 222–233, 2015.
- SLACK, N.; BRANDON-JONES, A.; JOHNSTON, R. **Operations Management**. 8th. ed. Harlow, UK: Pearson, 2016.
- WICHMANN, M. G.; JOHANNES, C.; SPENGLER, T. S. Energy-oriented lot-sizing and scheduling considering energy storages. **International Journal of Production Economics**, v. 216, p. 204–214, 2019.
- ZHANG, S.; LI, X.; ZHANG, B.; WANG, S. Multi-objective optimisation in flexible assembly job shop scheduling using a distributed ant colony system. **European Journal of Operational Research**, v. 283, n. 2, p. 441–460, 2020.
- ZOHALI, H.; NADERI, B.; MOHAMMADI, M.; ROSHANA EI, V. Reformulation, linearization, and a hybrid iterated local search algorithm for economic lot-sizing and sequencing in hybrid flow shop problems. **Computers and Operations Research**, v. 104, p. 127–138, 2019.

## APÊNDICE A – Pseudocódigo detalhado da heurística da construção

**Função** [seq, tst, erro] = Sequência( $tam_{irc}, d, a, b, s, \bar{l}, h, m, dk, f$ )

$$dt_j \leftarrow \frac{d_j}{a_j}, j \in V$$

$$td_j \leftarrow \frac{h_j - m_j}{b_j}, j \in U$$

$c \leftarrow 1$  //contador

$i \leftarrow \arg \min(td)$

$v \leftarrow i$

$l \leftarrow \arg \max(td)$

**Enquanto** soma( $dk$ ) > 0

**Enquanto**  $v \in U$

**Se**  $v \neq i$

$c \leftarrow c + 1$

**Fim Se**

$seq_{c,1} \leftarrow i$

$\bar{t}_i \leftarrow \min\left(\frac{\bar{l}_i}{a_i}, dt_i\right)$

**Se**  $dk_i > 0$

$$tt_j \leftarrow \begin{cases} 0, & \text{se } dt_j = 0 \\ f_{ij} \cdot f_{jl} \cdot (s_{ij} + dt_j + s_{jl}), & \text{se } dt_j > 0, j \in V \setminus U \end{cases}$$

**Se**  $(tt > 0) = \emptyset$

$seq \leftarrow \emptyset$

$tst \leftarrow 0$

$erro \leftarrow 1$

**Retorna**

**Fim Se**

**Se**  $\bar{t}_i + \min(tt > 0) > td_i$

$seq_{c,2} \leftarrow td_i - \min(tt > 0)$

$dt_i \leftarrow dt_i - seq_{c,2}$

$dk_i \leftarrow \max(0; dk_i - seq_{c,2} \cdot a_i)$

$c \leftarrow c + 1$

$v \leftarrow \arg \min(tt > 0)$

$seq_{c,1} \leftarrow v$

$seq_{c,2} \leftarrow dt_v$

$dt_v \leftarrow dt_v - seq_{c,2}$

**Se**  $c = 2$

$$h_i \leftarrow h_i + seq_{c-1,2} \cdot (a_i - b_i) - (s_{iv} + seq_{c,2}) \cdot b_i$$

$$h_l \leftarrow h_l - (seq_{c-1,2} + s_{iv} + seq_{c,2}) \cdot b_l$$

**Senão**

$$z \leftarrow seq_{c-2,1}$$

$$h_i \leftarrow h_i + seq_{c-1,2} \cdot (a_i - b_i) - (s_{zi} + s_{iv} + seq_{c,2}) \cdot b_i$$

$$h_l \leftarrow h_l - (s_{zi} + seq_{c-1,2} + s_{iv} + seq_{c,2}) \cdot b_l$$

**Fim Se**

$$td_j \leftarrow \frac{h_j - m_j}{b_j}, j \in U$$

$$i \leftarrow \arg \min(td)$$

$$v \leftarrow i$$

$$c \leftarrow c + 1$$

**Senão**

$$seq_{c,2} \leftarrow \bar{t}_i$$

**Fim Se**

**Senão**

$$seq_{c,2} \leftarrow \bar{t}_i$$

**Fim Se**

**Se**  $seq_{c,1} \in U$

$$dt_i \leftarrow dt_i - seq_{c,2}$$

$$dk_i \leftarrow \max(0; dk_i - seq_{c,2} \cdot a_i)$$

**Se**  $c = 1$

$$h_i \leftarrow h_i + (a_i - b_i) \cdot seq_{c,2}$$

$$h_l \leftarrow h_l - b_l \cdot seq_{c,2}$$

**Senão**

$$z \leftarrow seq_{c-1,1}$$

$$h_i \leftarrow h_i + (a_i - b_i) \cdot seq_{c,2} - s_{zi} \cdot b_i$$

$$h_l \leftarrow h_l - (s_{zi} + seq_{c,2}) \cdot b_l$$

**Se**  $f_{zi} = 0$

$$seq \leftarrow \emptyset$$

$$tst \leftarrow 0$$

$$erro \leftarrow 2$$

**Retorna**

**Fim Se**

**Fim Se**

$$td_j \leftarrow \frac{h_j - m_j}{b_j}, j \in K$$

$$i \leftarrow \arg \min(td)$$

$$l \leftarrow \arg \max(td)$$

$$tl_j \leftarrow \begin{cases} 0, & \text{se } dt_j = 0 \\ s_{vj} + dt_j + s_{ji}, & \text{se } dt_j > 0 \end{cases}$$

$v \leftarrow 0$

**Fim Se**

**Fim Enquanto**

**Enquanto**  $\min(tl > 0) < td_i \& \text{sum}(dk) > 0 \& v \notin U$

$z \leftarrow seq_{c,1}$

$prp_i \leftarrow f_{zi}$

$prp_l \leftarrow 0$

$prp_j \leftarrow f_{zj} \cdot (tl_j > 0) \cdot (tl_j < td_i), j \in V \setminus U$

**Se**  $(prp > 0) = \emptyset$

$seq \leftarrow \emptyset$

$tst \leftarrow 0$

$erro \leftarrow 3$

**Retorna**

**Fim Se**

$sl_j \leftarrow prp_j \cdot s_{zj}, j \in V$

$cut \leftarrow \text{tam}_{lrc}$ -ésimo menor valor em  $(sl > 0)$

$lrc \leftarrow \arg((sl > 0) \& (sl \leq cut))$

$c \leftarrow c + 1$

$v \leftarrow \text{rand}(lrc)$

**Se**  $v \notin U$

$seq_{c,1} \leftarrow v$

$seq_{c,2} \leftarrow dt_v$

$dt_v \leftarrow dt_v - seq_{c,2}$

$h_j \leftarrow h_j - (s_{zv} + seq_{c,2}) \cdot b_j, j \in U$

$td_j \leftarrow \frac{h_j - m_j}{b_j}, j \in U$

$tl_j = \begin{cases} 0, & \text{se } dt_j = 0 \\ s_{vj} + dt_j + s_{ji}, & \text{se } dt_j > 0 \end{cases}$

**Fim Se**

**Fim Enquanto**

**Fim Enquanto**

**Enquanto**  $\text{soma}(dt) > 0$

$prp_j \leftarrow f_{vj} \cdot (dt_j > 0), j \in V$

**Se**  $(prp > 0) = \emptyset$

$seq \leftarrow \emptyset$

$tst \leftarrow 0$

$erro \leftarrow 4$

**Retorna**

**Fim Se**

$sl_j \leftarrow prp_j \cdot s_{vj}, j \in V$   
 $cut \leftarrow tam_{lrc}$ -ésimo menor valor em  $(sl > 0)$   
 $lrc \leftarrow \arg((sl > 0) \& (sl \leq cut))$   
 $c \leftarrow c + 1$   
 $v \leftarrow rand(lrc)$   
 $seq_{c,1} \leftarrow v$   
 $seq_{c,2} \leftarrow dt_v$   
 $dt_v \leftarrow dt_v - seq_{c,2}$

**Fim Enquanto**

Verificar se as restrições de produtos K em ciclos consecutivos são respeitadas

Verificar se as restrições de produtos E em um mesmo ciclo ou em sequência são respeitadas

Para uma das instâncias: Verificar se o produto D300 é alocado antes do primeiro lote do segundo produto U

**Se** alguma dessas restrições não for respeitada

$seq \leftarrow \emptyset$   
 $tst \leftarrow 0$   
 $erro \leftarrow 5$

**Retorna**

**Fim Se**

$erro \leftarrow 0$   
 $tst \leftarrow 0$

**Para** i de 2 a c

$p \leftarrow seq_{i-1,1}$   
 $v \leftarrow seq_{i,1}$   
 $tst \leftarrow tst + s_{pv}$

**Fim Para**

**Fim Função**



## APÊNDICE B – Exemplo numérico da heurística de construção

Para a melhor compreensão da heurística, um exemplo numérico é apresentado. O exemplo contém sete produtos, dos quais dois ( $j = 1$  e  $j = 2$ ) sofrem retiradas periódicas. As listas restritas de candidatos (LRC) foram sempre construídas com dois elementos. Utilizaram-se os dados de entrada fornecidos nas Tabelas B.1, B.2, B.3 e B.4. A Tabela B.1 fornece a demanda total e a taxa de produção para todos os produtos considerados.

**Tabela B.1** - Demanda e taxa de produção para todos os produtos

$j$	$d_j$ (ton)	$a_j$ (ton/h)
1	7000	47
2	9000	45
3	1200	49
4	700	43
5	450	50
6	920	44
7	1050	46

A Tabela B.2 fornece para os produtos com retiradas constantes, respectivamente: a taxa de demanda, o lote máximo, o estoque inicial, o estoque de segurança e a demanda destinada às retiradas contínuas.

**Tabela B.2** - Dados para os produtos com restrição de estoque

$j$	$b_j$ (ton/h)	$\bar{l}_j$ (ton)	$h_j$ (ton)	$m_j$ (ton)	$dk_j$ (ton)
1	4	4000	1100	700	3000
2	10	5000	2900	2000	8500

A Tabela B.3 fornece os tempos de *setup* para todas as trocas, e a Tabela B.4 informa quais pares de trocas são permitidos ( $f = 1$ ) ou não ( $f = 0$ ). As trocas não permitidas são entre produtos iguais, entre os dois produtos com retiradas contínuas e entre os produtos 6 e 7, nos dois sentidos.

**Tabela B.3** - Tempos de *setup* para as trocas (h)

<i>i</i>	<i>s<sub>i1</sub></i>	<i>s<sub>i2</sub></i>	<i>s<sub>i3</sub></i>	<i>s<sub>i4</sub></i>	<i>s<sub>i5</sub></i>	<i>s<sub>i6</sub></i>	<i>s<sub>i7</sub></i>
1	0	0,31538	0,39139	0,60669	0,25702	0,22631	0,33116
2	1,15965	0	1,15697	0,41757	1,04754	0,85711	0,31495
3	0,87656	0,74210	0	0,91506	0,31957	0,57228	0,87500
4	0,69224	0,37975	0,40339	0	0,69181	0,59911	0,72214
5	0,65916	0,27056	0,99288	0,87300	0	0,72206	0,34900
6	0,92704	0,35458	0,76151	0,84563	1,11787	0	0,62654
7	0,29667	0,63600	1,18740	0,50782	0,97299	0,31046	0

**Tabela B.4** - Trocas permitidas ou não

<i>i</i>	<i>f<sub>i1</sub></i>	<i>f<sub>i2</sub></i>	<i>f<sub>i3</sub></i>	<i>f<sub>i4</sub></i>	<i>f<sub>i5</sub></i>	<i>f<sub>i6</sub></i>	<i>f<sub>i7</sub></i>
1	0	0	1	1	1	1	1
2	0	0	1	1	1	1	1
3	1	1	0	1	1	1	1
4	1	1	1	0	1	1	1
5	1	1	1	1	0	1	1
6	1	1	1	1	1	0	0
7	1	1	1	1	1	0	0

O primeiro passo da heurística é calcular a demanda dos produtos em função do tempo, chamada de *dt*. Os resultados para esse cálculo encontram-se na Tabela B.5.

**Tabela B.5** - Demanda temporal inicial dos produtos

<i>j</i>	<i>dt<sub>j</sub></i> (h)
1	148,94
2	200,00
3	24,49
4	16,28
5	9,00
6	20,91
7	22,83

Depois, calcula-se o tempo para que os dois produtos com retiradas constantes atinjam o estoque de segurança. Isso é feito por meio da expressão:

$$td_j = \frac{h_j - m_j}{b_j}$$

Inicialmente, esse tempo é dado pelos valores mostrados na Tabela B.6.

**Tabela B.6** - Duração do estoque dos produtos no início do algoritmo

$j$	$td_j$ (h)
1	100
2	90

O primeiro produto a ser alocado é o que possui menor duração do estoque. No corrente momento, trata-se do produto 2. Feita a escolha, deve-se definir a dimensão do lote com base no tempo de duração do estoque do produto 1, uma vez que ainda há demanda constante para ele. Para isso, calcula-se o tempo necessário para alocar cada produto de 3 a 7 entre os produtos 2 e 1. São excluídos produtos que já tenham tido sua demanda atendida e os que possuem  $f_{2j}$  ou  $f_{j1}$  iguais a 0. Como a sequência ainda não foi iniciada, todos os produtos de 3 a 7 ainda possuem demanda por cumprir. Nas iterações seguintes, elementos são excluídos por esse critério. O tempo necessário para a alocação é dado pela soma do *setup* de 2 para  $j \in V \setminus U$ , o tempo de produção de  $j$  e o *setup* de  $j$  para 1. Para os produtos não viáveis, deseja-se que o resultado retornado seja 0, e isso é feito por meio de expressões lógicas. Sendo assim, o tempo é dado pela expressão:

$$tt_j = (s_{2j} + dt_j + s_{j1}) \cdot (dt_j > 0) \cdot f_{2j} \cdot f_{j1}$$

O primeiro termo entre parênteses corresponde à soma dos tempos de *setup* e de produção. Os demais termos são valores lógicos, respectivamente: se ainda há demanda do produto  $j$  por ser atendida, se é possível a troca do produto 2 para o produto  $j$  e se é possível a troca do produto  $j$  para o produto 1. Caso um deles seja igual a 0,  $tt_j$  passa a valer 0, indicando que aquele produto não pode ser considerado neste cálculo. Os resultados são dados na Tabela B.7.

Escolhe-se o menor valor não-nulo, que foi o do produto  $j = 5$ . É possível fabricar o lote máximo do produto 2 se o tempo para fabricação do lote máximo somado a  $tt_5$  seja menor ou igual ao tempo para que o produto 1 atinja o estoque de segurança. O lote máximo do produto 2, em tempo, é dado por:

$$\bar{lt}_2 = \min\left(\frac{\bar{l}_2}{a_2}; dt_2\right) = \min\left(\frac{5000}{45}; 200,00\right) = \min(111,11; 200,00) = 111,11 h$$

**Tabela B.7** - Cálculo do tempo de alocação de um produto entre os de demanda contínua, após definição do primeiro produto alocado

$j$	$s_{2j}$ (h)	$dt_j$ (h)	$s_{j1}$ (h)	$dt_j > 0$	$f_{2j}$	$f_{j1}$	$tt_j$ (h)
3	1,15697	24,49	0,87656	1	1	1	26,52
4	0,41757	16,28	0,69224	1	1,	1	17,39
5	1,04754	9,00	0,65916	1	1	1	10,71
6	0,85711	20,91	0,92704	1	1	1	22,69
7	0,31495	22,83	0,29667	1	1	1	23,44

Esse valor, somado ao tempo para incluir um lote de 5, resulta em:

$$\bar{lt}_2 + tt_5 = 111,11 + 10,707 = 121,82 \text{ h}$$

O tempo para o produto 1 atingir o estoque de segurança é  $td_1 = 100 \text{ h}$ . Dessa forma, não é possível alocar o lote máximo de 2. O tamanho do lote, portanto, é dado pela diferença entre a duração do estoque do produto 1 e o tempo necessário para alocar um lote de 5:

$$td_1 - tt_5 = 89,293 \text{ h}$$

A sequência, nesse momento, apresenta-se como mostrada na Tabela B.8.

**Tabela B.8** - Sequência após alocação e dimensionamento do primeiro produto

Produto	Lote (h)
2	89,29

Uma vez que, pela maneira como o código foi construído, o único produto que pode ser alocado antes que o 1 atinja o estoque de segurança é o  $j = 5$ , ele já é alocado na sequência, sem que haja a necessidade de construção da LRC. O lote alocado é exatamente igual à sua demanda, e a sequência encontra-se como mostrada na Tabela B.9.

**Tabela B.9** - Sequência após alocação dos dois primeiros produtos

Produto	Lote (h)
2	89,29
5	9,00

Alocados esses dois lotes, são atualizadas as demandas temporais dos dois produtos:

$$dt_2 = 200,00 - 89,293 = 110,71 \text{ h}$$

$$dt_5 = 9,00 - 9,00 = 0$$

Atualiza-se também a demanda de 2, em toneladas, destinada às retiradas constantes:

$$dk_2 = \max(0; 8500 - 89,293 \cdot a_2) = \max(0; 4481,8) = 4481,8 \text{ ton}$$

O estoque dos produtos destinados às retiradas constantes deve também ser atualizado. São considerados o tempo de produção dos produtos 2 e 5 e o tempo de *setup*  $s_{25}$ .

$$h_1 = 1100 - (89,293 + s_{25} + 9,00) \cdot b_1$$

$$h_1 = 702,64 \text{ ton}$$

$$h_2 = 2900 + 89,293 \cdot (a_2 - b_2) - (s_{25} + 9,00) \cdot b_2$$

$$h_2 = 5924,79 \text{ ton}$$

Neste ponto, sabe-se que é necessário alocar o produto 1 imediatamente após o 5, porém, para todos os efeitos, recalcula-se o tempo para que os produtos 1 e 2 atinjam o estoque de segurança. Os resultados encontrados são dados na Tabela B.10.

**Tabela B.10** - Duração do estoque dos produtos após alocação de dois produtos

$j$	$td_j$ (h)
1	0,659
2	392

Definido que o lote 1 deve ser alocado, realiza-se o procedimento para determinar a dimensão do lote, uma vez que ainda há demanda constante do produto 2 por cumprir. Os dados utilizados são dados na Tabela B.11.

**Tabela B.11** - Cálculo do tempo de alocação de um produto entre os de demanda contínua, após definição do terceiro produto alocado

$j$	$s_{1j}$ (h)	$dt_j$ (h)	$s_{j2}$ (h)	$dt_j > 0$	$f_{1j}$	$f_{j2}$	$tt_j$ (h)
3	0,39139	24,49	0,74210	1	1	1	25,62
4	0,60669	16,28	0,37975	1	1	1	17,27
5	0,25702	0	0,27056	0	1	1	0
6	0,22631	20,91	0,35458	1	1	1	21,49
7	0,33116	22,83	0,63600	1	1	1	23,79

O produto 5 assumiu o valor 0 porque sua demanda já foi atendida, então ele não pode mais ser alocado. Dentre os possíveis produtos, o menor tempo é o do produto 4. O lote máximo do produto 1, em unidade de tempo, é igual a:

$$\bar{t}_1 = \min\left(\frac{\bar{l}_1}{a_1}; dt_1\right) = \min\left(\frac{4000}{47}; 148,94\right) = \min(85,106; 148,94) = 85,106 \text{ h}$$

Esse tempo, somado ao tempo para alocar um lote do produto 4 entre os produtos 1 e 2, é igual a 102,37 h. Este valor é inferior ao tempo para que o produto 2 atinja o estoque de segurança (392,48 h), então é possível alocar o lote máximo de 1. A sequência, com a alocação do produto 1, é dada na Tabela B.12.

**Tabela B.12** - Sequência após alocação de três produtos

Produto	Lote (h)
2	89,29
5	9,00
1	85,11

A atualização do nível de estoque dos produtos 1 e 2 deve levar em consideração o tempo de *setup* da troca do produto 5 para o 1 e o tempo de produção de 1.

$$h_1 = 702,64 - s_{51} \cdot b_1 + 85,11 \cdot (a_1 - b_1)$$

$$h_1 = 4359,6 \text{ ton}$$

$$h_2 = 5924,8 - (s_{51} + 85,11) \cdot b_2$$

$$h_2 = 5067,1 \text{ ton}$$

O tempo de duração de estoque nesse momento é dado na Tabela B.13.

**Tabela B.13** - Duração do estoque dos produtos após alocação de três produtos

$j$	$td_j$ (h)
1	915
2	307

Atualizam-se a demanda total e a demanda destinada às retiradas constantes:

$$dt_1 = 148,94 - 85,11 = 63,83 \text{ h}$$

$$dk_1 = \max(0; 3000 - 85,11 \cdot a_1)$$

$$dk_1 = \max(0; -1000)$$

$$dk_1 = 0$$

Neste ponto, a demanda do produto 1 destinada às retiradas constantes foi completamente atendida, e as 1000 toneladas excedentes são destinadas ao atendimento das demandas discretas. O produto 2 ainda possui demanda constante a ser atendida, então o tempo de duração do seu estoque ainda deve ser observado.

Procede-se, então, para a alocação dos outros produtos. Verifica-se primeiro o tempo necessário para alocar cada produto  $j \in V \setminus U$  entre o último alocado e o próximo a atingir o estoque de segurança. Neste momento, optou-se por vetar apenas produtos cuja demanda já foi atendida. A verificação da possibilidade de troca foi realizada no momento da construção da lista de candidatos. O código possui mecanismos para identificar quando trocas não viáveis são colocadas, os quais foram fornecidos com maiores detalhes no Apêndice A deste trabalho. Desta forma, o tempo  $tl_j$ , nesse ponto, é dado por:

$$tl_j = (s_{1j} + dt_j + s_{j2}) \cdot (dt_j > 0)$$

Para este momento no código, o procedimento para o cálculo do tempo  $tl_j$  é dado na Tabela B.14.

**Tabela B.14** - Cálculo do tempo  $tl_j$  após alocação de três produtos

$j$	$s_{1j}$ (h)	$dt_j$ (h)	$s_{j2}$ (h)	$dt_j > 0$	Resultado (h)
3	0,39139	24,49	0,74210	1	25,62
4	0,60669	16,28	0,37975	1	17,27
5	0,25702	0	0,27056	0	0
6	0,22631	20,91	0,35458	1	21,49
7	0,33116	22,83	0,63600	1	23,79

O produto 5 é desconsiderado por já ter sido alocado. Todos os demais produtos possuem tempo para alocação inferior ao tempo para que o produto 2 atinja o estoque de segurança.

A lista dos possíveis sucessores é composta pelo elemento  $j \in U$  que não tenha sido o último deste conjunto a ser alocado, a não ser que a troca não seja permitida, e pelos elementos  $j \in V \setminus U$  que, ao mesmo tempo, não tenham sido alocados, o *setup* a partir do produto anterior seja permitido e possuam tempo de alocação inferior à duração do estoque do próximo produto em  $U$  a atingir o estoque de segurança. Essa análise é mostrada na Tabela B.15.

**Tabela B.15** - Possíveis sucessores do terceiro elemento alocado

$j$	$0 < tl_j < dur$	$f_{1j}$	Resultado	$s_{1j}$ (h)
1	-	-	0	0
2	-	0	0	0,31538
3	1	1	1	0,39139
4	1	1	1	0,60669
5	0	1	0	0,25702
6	1	1	1	0,22631
7	1	1	1	0,33116

Portanto, a lista dos possíveis sucessores é composta por  $l = \{3,4,6,7\}$ , com valores de *setup*  $\{0,39139; 0,60669; 0,22631; 0,33116\}$ . Conforme dito, a LRC deve possuir tamanho 2. Os menores tempos de *setup* são os dos produtos 6 e 7, fazendo com que  $LRC = \{6,7\}$ . Então, escolhe-se aleatoriamente um desses dois elementos. Escolheu-se o elemento 6, do qual será produzida toda a demanda neste lote. A sequência obtida é dada na Tabela B.16.

**Tabela B.16** - Sequência após alocação de 4 produtos

Produto	Lote (h)
2	89,29
5	9,00
1	85,11
6	20,91

A cada produto alocado, atualizam-se os estoques dos produtos com demanda constante e a duração desse estoque, bem como a demanda temporal do produto alocado, como ilustrado em outros pontos desta explicação. O tempo necessário para alocar um produto  $j \in V \setminus U$  passa a ser calculado utilizando-se o *setup* do último produto posicionado:

$$tl_j = (s_{6j} + dt_j + s_{j2}) \cdot (dt_j > 0)$$

O cálculo é feito de maneira semelhante ao realizado na Tabela B.14, e os elementos 3, 4 e 7 possuem  $tl_j$  inferior ao tempo para que o produto 2 atinja o estoque de segurança. Os elementos 5 e 6 possuem  $tl_j$  nulo por sua demanda ter sido cumprida. A Tabela B.17 mostra os possíveis sucessores do quarto elemento alocado, bem como seu tempo de *setup*.



**Tabela B.17** - Possíveis sucessores do quarto elemento alocado

$j$	$0 < tl_j < dur$	$f_{6j}$	Resultado	$s_{6j}$ (h)
1	-	-	-	0,92704
2	-	1	1	0,35458
3	1	1	1	0,76151
4	1	1	1	0,84563
5	0	1	0	1,11787
6	0	0	0	0
7	1	0	0	0,62654

A lista dos possíveis sucessores é composta por  $l = \{2,3,4\}$ , com valores de *setup*  $\{0,35458; 0,76151; 0,84563\}$ . A LRC é composta pelos elementos 2 e 3, os quais possuem os menores tempos de *setup*. Dentre eles, escolhe-se aleatoriamente o elemento 3, e a sequência atual é dada na Tabela B.18.

**Tabela B.18** - Sequência após alocação de cinco produtos

Produto	Lote (h)
2	89,29
5	9,00
1	85,11
6	20,91
3	24,49

Os procedimentos já enunciados, para atualização de demanda e duração de estoque, são repetidos. Os possíveis sucessores de 3 são dados na Tabela B.19.

**Tabela B.19** - Possíveis sucessores do quinto elemento alocado

$j$	$0 < tl_j < dur$	$f_{3j}$	Resultado	$s_{3j}$
1	-	-	-	0,87656
2	-	1	1	0,74210
3	0	0	0	0
4	1	1	1	0,91506
5	0	1	0	0,31957
6	0	1	0	0,57228
7	1	1	1	0,87500

A LRC é composta pelos elementos 2 e 7, que, dentre os possíveis, possuem os menores tempos de *setup*. Sorteia-se, então, o elemento 2. Por ele ser um elemento com retiradas constantes, deve-se dirigir ao *loop* para alocação de elementos dessa categoria, pois os dados relativos à demanda destinada às retiradas constantes devem ser atualizados assim que o lote for definido.

Neste ponto, a demanda do produto 1 destinada às retiradas constantes já foi completamente atendida, então não é necessário passar pela etapa de verificação do menor lote a ser alocado entre os produtos 2 e 1. Sendo assim, o lote alocado é dado por:

$$\bar{l}t_2 = \min\left(\frac{\bar{l}_2}{a_2}; dt_2\right) = \min(111,11; 110,71) = 110,71 h$$

O lote alocado, então, possui duração de 110,71 h, e a sequência até o momento é dada na Tabela B.20.

**Tabela B.20** - Sequência após alocação de seis lotes

Produto	Lote (h)
2	89,29
5	9,00
1	85,11
6	20,91
3	24,49
2	110,71

Feita a alocação, atualiza-se  $dk_2$ :

$$dk_2 = \max(0; 4481,8 - 110,71 \cdot a_2) = \max(0; -500) = 0$$

Como esperado, uma vez que toda a demanda do produto 2 foi atendida, a demanda destinada às retiradas constantes também foi atendida. A partir desse ponto, não é mais necessário verificar as restrições de estoque, e o código parte para o *loop* mais simples. Nesse momento, a lista de possíveis sucessores de cada elemento é composta pelos produtos que ainda apresentem demanda por cumprir e cujo *setup* seja permitido. Os produtos com retiradas constantes passam a ser tratados como os demais. Após a alocação do produto 2, a organização dos possíveis sucessores é dada pela Tabela B.21.

**Tabela B.21** - Possíveis sucessores do sexto elemento alocado

$j$	$dt_j > 0$	$f_{2j}$	Resultado	$s_{2j}$ (h)
1	1	0	0	1,15965
2	0	0	0	0
3	0	1	0	1,15697
4	1	1	1	0,41757
5	0	1	0	1,04754
6	0	1	0	0,85711
7	1	1	1	0,31495

Os únicos produtos possíveis são o 4 e o 7, portanto a LRC é composta por eles. Neste caso, sorteou-se o produto 7. A alocação dos demais produtos é feita da mesma maneira, até que a demanda de todos eles tenha sido atendida. A sequência final é dada na Tabela B.22.

**Tabela B.22** - Sequência final obtida pela heurística de construção

Produto	Lote (h)
2	89,29
5	9,00
1	85,11
6	20,91
3	24,49
2	110,71
4	16,28
1	63,83
7	22,83
<b>Setup (h)</b>	<b>4,8776</b>

Cabe ressaltar que, devido à parte aleatória do algoritmo, é possível a obtenção de sequências diferentes da obtida nesta demonstração.