

Universidade Federal Fluminense

MOYSÉS DUTRA DA SILVA

Desenvolvimento de modelos exato e heurística
para sequenciamento de pedidos de uma linha
de galvanização contínua considerando
restrições de qualidade

VOLTA REDONDA

2021

MOYSÉS DUTRA DA SILVA

Desenvolvimento de modelos exato e heurística
para sequenciamento de pedidos de uma linha
de galvanização contínua considerando
restrições de qualidade

Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional em Ciência e Tecnologia da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre em Modelagem Computacional em Ciência e Tecnologia. Área de Concentração: Modelagem Computacional e Pesquisa Operacional.

Orientador:

Tiago Araújo Neves

Coorientador:

Wesley Luiz Da Silva Assis

UNIVERSIDADE FEDERAL FLUMINENSE

VOLTA REDONDA

2021

Ficha catalográfica automática - SDC/BEM
Gerada com informações fornecidas pelo autor

S586d Silva, Moysés Dutra da
Desenvolvimento de modelos exato e heurística para
sequenciamento de pedidos de uma linha de galvanização
contínua considerando restrições de qualidade / Moysés
Dutra da Silva ; Tiago Araújo Neves, orientador ; Wesley
Luiz Da Silva Assis, coorientador. Volta Redonda, 2021.
107 f. : il.

Dissertação (mestrado)-Universidade Federal Fluminense,
Volta Redonda, 2021.

DOI: <http://dx.doi.org/10.22409/PPG-MCCT.2021.m.09536927799>

1. Otimização de processo. 2. Heurística. 3. Otimização
Combinatória. 4. Metaheurística. 5. Produção intelectual.
I. Neves, Tiago Araújo, orientador. II. Assis, Wesley Luiz
Da Silva, coorientador. III. Universidade Federal Fluminense.
Escola de Engenharia Industrial e Metalúrgica de Volta
Redonda. IV. Título.

CDD -

Desenvolvimento de modelos exato e heurística para sequenciamento de pedidos de uma linha de galvanização contínua considerando restrições de qualidade.

Moysés Dutra da Silva

Dissertação apresentada ao Programa de Pós-graduação em Modelagem Computacional em Ciência e Tecnologia da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre em Modelagem Computacional em Ciência e Tecnologia. Área de Concentração: Modelagem Computacional e Pesquisa Operacional.

Aprovada por:



Prof. Tiago Araújo Neves, D.Sc. / MCCT - UFF (Orientador)



Prof. Wesley Luiz da Silva Assis, D.Sc. / MCCT - UFF (Coorientador)



Prof. Antonio Iacono, D.Sc. / DEQUI - USP



Profª. Giselle de Mattos Araújo, D.Sc. / UFF



Prof. Puca Huachi Vaz Penna, D.Sc. / DECOM - UFOP



Prof. Luís Alberto Duncan Rangel, D.Sc./ PPGEP-UFF

Volta Redonda, 03 de maio de 2021.

Dedicatória. Para os meus pais e minha família.

Agradecimentos

A Deus que sempre esteve e está presente nos fortalecendo e iluminado as nossas mentes permitindo que isso tudo fosse possível!

A minha família, em especial a minha esposa Marcela M. F. Dutra, meus pais pelo apoio e incentivo em todas as horas, principalmente nos momentos de grandes lutas. Ao meu filho Miguel M. Dutra, que sem imaginar, sempre foi o combustível que faltava durante os momentos de cansaço extremo.

Ao amigo e colega de classe, Vitor Tomaz de Aquino que além de estar em sala apoiando esteve também fora dela incentivando e trazendo idéias para os problemas durante o desenvolvimento desta dissertação.

Aos professores Tiago Araújo Neves e Wesley Luiz da Silva Assis, que dedicaram um grande tempo nos ensinando e orientando, e que também foram guias nos momentos em que estivemos sem rumo.

A todos os demais que contribuíram para a realização deste trabalho.

Resumo

Neste trabalho foi abordado o problema de sequenciamento de máquina única com penalidades por antecipação e atraso em uma linha de galvanização contínua de uma indústria siderúrgica. Foi proposta uma nova abordagem adicionando ao problema a avaliação das características físicas dos materiais que forem sequenciados respeitando os limites estabelecidos pelo equipamento. Este sequenciamento objetiva minimizar as antecipações e atrasos das entregas. Uma segunda finalidade desse sequenciamento é a minimização do custo de processamento dos pedidos que necessitam de uma etapa extra chamada tratamento químico. O processamento do material com esta característica requer um trabalho extra na linha e por isso torna-se imprescindível que o sequenciamento seja capaz de agrupar estes materiais para evitar a elevação desnecessária do seu custo de fabricação. Foram incluídas restrições que fazem o modelo trabalhar reduzindo as diferenças dimensionais entre itens sequenciados e também formando uma sequência de itens com este tratamento. Outra consideração foi o agrupamento de itens de ciclo térmico compatíveis. Ação com foco na garantia da qualidade do material. Para tratar o problema foram desenvolvidos modelos em programação linear inteira mista baseados nas formulações clássicas de sequenciamento de máquina única. Além disso, por se tratar de um problema do tipo *NP-Difícil*, foi também utilizada uma meta-heurística baseada no algoritmo genético de chaves aleatórias viciadas ou **BRKGA**. A performance e os resultados obtidos pelos dois modelos foram comparados e analisados. Encontraram-se resultados satisfatórios com equilíbrio entre as antecipações, atrasos e as mudanças dimensionais dos materiais nas sequências geradas pelos modelos implementados. A heurística foi capaz de alcançar o resultado ótimo global em aproximadamente 80% das simulações.

Abstract

In this work the problem of single machine scheduling with penalties for earliness and tardiness in a continuous galvanizing line of a steel industry will be addressed, but a new approach has been added to the problem that must also evaluate the physical characteristics of the materials that are sequenced respecting the limits established by the equipment that processes them. This sequencing aims to minimize the earliness and tardiness of deliveries. A second purpose of this sequencing would be to minimize the cost of processing orders that require an extra step called chemical treatment. The processing of the material with this feature requires extra work on the line and so it is essential that the sequencing is able to group these materials to avoid unnecessary increase in the cost of this material. Constraints were included that make the model work by reducing the dimensional differences between sequenced items and also forming a sequence of items with this treatment. Another consideration was the grouping of similar thermal cycle items. This grouping will avoid quality deviations due to the temperature difference in the annealing stages of each of the orders. To deal with this problem, models were developed in mixed integer linear programming based on the classic machine scheduling formulations. Because it is characterized as a NP-Hard problem, a meta-heuristic based on **BRKGA** - Biased Random-Key Genetic Algorithm. The performance and results obtained by the two models were compared and analyzed. Finally, satisfactory results were found with a balance between earliness, tardiness and dimensional changes of materials in the sequences generated by the model. The heuristic found the optimal result in approximately 80% of the simulations.

Palavras-chave

1. Otimização de processos
2. Formulação matemática
3. Heurística BRKGA
4. Problema de sequenciamento de máquina única
5. Linha de galvanização contínua

Glossário

- abs : Função que retorna o valor absoluto de um número.
- AG : Algoritmo Genético.
- BRKGA : (*Biased Random-Key Genetic Algorithm*) - AG de chaves aleatórias viciadas.
- CIS : Comunidade dos Estados Independentes formada pelos países Armênia, Azerbaijão, Bielorrússia, Cazaquistão, Quirguistão, Moldávia, Rússia, Tadjiquistão, Turcomenistão e Uzbequistão.
- GRASP : *Greedy Randomized Adaptive Search Procedure* - Procedimento de busca adaptativa gulosa e randomizada.
- LC : Lingotamento contínuo.
- LTF : Laminador de tiras a frio.
- LTQ : Laminador de tiras a quente.
- JIT : *Just in Time*.
- MAX : Função que retorna o maior valor entre os valores informados.
- MLC : Máquina de lingotamento contínuo.
- PLIM : Programação linear inteira mista.
- PCP : Planejamento e controle da produção.
- PMP : Plano mestre de produção.
- PSMP : Problema de sequenciamento em máquinas paralelas.
- PSMU : Problema de sequenciamento de máquina única.
- PSMUA : Problema de Sequenciamento de máquina única com antecipação e atraso da produção.
- UFF : Universidade Federal Fluminense.
- NAFTA : Grupo de países composto pelo Canadá, México e Estados Unidos

Sumário

Lista de Figuras	xi
Lista de Quadros	xii
1 Introdução	13
1.1 Contextualização	13
1.2 Objetivos da Dissertação	18
1.3 Classificação e Estrutura do Trabalho	19
2 Revisão Bibliográfica	21
2.1 Etapas da Produção do Aço	21
2.2 Processo da Linha de galvanização Contínua	23
2.3 Problemas de Sequenciamento	25
2.3.1 Sequenciamento de Máquina Única	30
2.3.2 Problemas Associados a Tempo de Processamento e Estoque	34
2.3.3 Problemas Associados a Data de Conclusão	36
2.3.4 Problemas de Minimização de Atrasos e Estoque	37
2.4 Programação Linear Mista	40
2.5 Heurísticas para a Resolução do PSMU	43
2.6 Meta-heurística <i>BRKGA</i> - Algoritmo Genético de Chaves Aleatórias Viciadas	47
2.7 Trabalhos Relacionados	54
3 Metodologia	59
3.1 Formulação para o Problema de Sequenciamento da Linha de Galvanização Contínua	59
3.1.1 Formulação para a Minimização dos Atrasos e das Antecipações . .	63
3.1.2 Formulação para o Tratamento das Características Dimensionais . .	64
3.1.3 Formulação para Tratamento do Ciclo Térmico, Laminador de En- cruamento e Aplicação de Tratamento Químico	66
3.2 Abordagem Heurística	73

3.3	Banco de Dados e Instâncias	77
4	Resultados	80
4.1	Resultados do Modelo Matemático	80
4.1.1	Resultados - Pedido 1	81
4.1.2	Resultados - Pedido 2	83
4.1.3	Resultados - Pedido 5	84
4.1.4	Resultados - Pedido 10	87
4.2	Resultados do algoritmo <i>BRKGA</i>	88
4.2.1	Resultados da heurística para o Pedido 1	88
4.2.2	Resultados da heurística para o Pedido 2	90
4.2.3	Resultados da heurística para o Pedido 5	92
4.2.4	Resultados da heurística para o Pedido 10	92
5	Conclusões e Trabalhos Futuros	95
5.1	Conclusões	95
5.2	Sugestões para Trabalhos Futuros	97
	Referências	98
	Apêndice A – Construção do Banco de Dados	103

Lista de Figuras

1.1	Estrutura da Dissertação	19
2.1	Representação da linha de galvanização contínua	24
2.2	Problemas de sequenciamento e suas características	28
2.3	Valor do atraso e da antecipação	38
2.4	Forma tabular do método simplex	41
2.5	Cruzamento de 1 ponto - Inválido	47
2.6	Cruzamento de 1 ponto - Válido	48
2.7	Problema da mochila com violação de restrição	49
2.8	Cruzamento no <i>BRKGA</i>	50
2.9	Evolução da População no <i>BRKGA</i>	51
2.10	<i>Framework</i> do <i>BRKGA</i>	52
2.11	Avaliação do algoritmo <i>BRKGA</i>	53
3.1	Distribuição dos pedidos por espessura	78
3.2	Distribuição dos pedidos por largura	79

Lista de Quadros

2.1	Resumo dos Trabalhos	56
3.1	Definição das instâncias para as simulações	77
4.1	Composição das instâncias testadas	81
4.2	Características dos pedidos testados	81
4.3	Características dos resultados para o Pedido PED.1	82
4.4	Sequenciamento com Pedido inicial 1 e 6 posições - INST.A	82
4.5	Atributos dimensionais, térmicos, TQ e laminador. Fila 6 - PED.1	83
4.6	Características dos resultados para o Pedido PED.2	83
4.7	Sequenciamento com Pedido inicial 2 e 8 posições - INST.A	84
4.8	Atributos dimensionais, térmicos, TQ e laminador. Fila 8 - PED.2	84
4.9	Características dos resultados para o Pedido PED.5	85
4.10	Sequência com 12 pedidos iniciando pelo PED.5 - INST.D	86
4.11	Atributos dimensionais, térmicos TQ e laminador - PED.5	86
4.12	Compilação dos resultados com PED.10 na primeira posição	87
4.13	Comparativo entre o <i>BRKGA</i> e o método exato - PED.1	89
4.14	Resultados incompletos do <i>CPLEX</i> - PED.1	90
4.15	Comparativo entre o <i>BRKGA</i> e o método exato - PED.2	91
4.16	Resultados incompletos do <i>CPLEX</i> - PED.2	91
4.17	Comparativo entre o <i>BRKGA</i> e o método exato - PED.5	92
4.18	Comparativo entre o <i>BRKGA</i> e o método exato - PED.10	93
4.19	Resultados incompletos do <i>CPLEX</i> - PED.10	93

Capítulo 1

Introdução

1.1 Contextualização

O aço caracteriza-se como uma importante matéria prima, principalmente nos dias atuais. Ele é aplicado em veículos, aviões, sistemas de distribuição de água, sistemas de telefonia. Na indústria, está presente em virtualmente todos os lugares, sejam em equipamentos e máquinas, sejam nas construções de alvenaria ou galpões metálicos. Possui uma grande importância na economia de um país, pois sua exploração e beneficiamento gera riquezas, empregos em todas as fases de sua fabricação e comercialização. Segundo Júnior (2002) é impensável imaginar o mundo contemporâneo sem a presença do aço.

Segundo Carvalho, Mesquita e Araújo (2015) o aço consiste em uma liga metálica composta de ferro e carbono com no mínimo noventa e oito por cento de ferro e no máximo dois por cento de carbono. Possui muitas características interessantes para a sua aplicação em engenharia estrutural, das quais podem ser destacadas a ductibilidade, resistência mecânica e maleabilidade.

Existem grandes indústrias que trabalham em todas as etapas de produção do aço, desde a produção do ferro gusa a partir do minério de ferro à produção de laminados a frio ou galvanizados. Este último é intensamente utilizado na indústria automobilística, fabricação de bens de capital, na produção de eletrodomésticos (aplicação também conhecida como linha branca). Além dos aços planos existe também outro processo produtivo siderúrgico que é de grande importância econômica, cujos seus produtos são classificados como aços longos. Um grande consumidor dos aços longos é ao ramo da construção civil, que consome principalmente o vergalhão. Há também a produção de tubos de aço com ou sem costura.

Um importante fator que permitiu o desenvolvimento da indústria siderúrgica foi a grande disponibilidade da sua principal matéria prima no país, o minério de ferro (VIANA, 2018), que vale ressaltar, é um dos melhores do mundo pois possui alto teor de ferro.

Carvalho, Mesquita e Araújo (2015) apresentam informações sobre a etapa produtiva de conformação através da laminação. Nela ocorre a conformação mecânica dos tarugos e placas produzidos anteriormente. O processo se dá através da passagem dos insumos (barras, placas ou tarugos) através de grandes e poderosos equipamentos, chamados de laminadores. Neste processo de laminação o principal objetivo é a redução da espessura do material para posterior beneficiamento. Além da espessura, cada material possui características que são definidas conforme os pedidos dos clientes e podem ser citadas como outros exemplos a largura e a liga do aço.

Na primeira parte desta fase do beneficiamento ocorre o processamento no laminador de tiras a quente (LTQ). Ele recebe as placas ou tarugos, que foram aquecidos para permitir uma maior ductibilidade ao aço, os processa passando entre cilindros até a obtenção das dimensões desejadas pelo cliente ou pelo processo/equipamento seguinte no fluxo produtivo.

Um dos processos que pode suceder o LTQ é a laminação a frio (LTF), que consiste em passar noutro tipo de laminador uma bobina para obtenção de um novo material, com características diferentes, entre elas a espessura muito menor. Os materiais do LTF são aplicados em linhas de recozimento contínuo, linhas de galvanização contínuas e também processo do tipo batelada.

O processamento das bobinas nos equipamentos que sucedem o LTF na linha de produção não ocorre de forma aleatória. É necessário realizar a ordenação dos pedidos respeitando as regras de processamento, necessidades dos clientes, demandas de insumos, tempos de preparação do equipamento (*setup*), entre outras características. Uma ordenação mal feita pode provocar perdas de performance, quedas na qualidade do material, até descarte de matéria prima por problemas no processamento.

Levi, Kaminsky e Levi (2003) e Bretas (2013) apresentam o planejamento da produção como uma atividade realizada em três níveis hierárquicos da empresa, o estratégico, tático e o operacional.

As decisões tomadas no nível estratégico normalmente estão associadas ao planejamento da capacidade da empresa e acabam também direcionando os níveis tático e operacional. Neste nível também são tratados os assuntos relacionados ao crescimento da

organização, políticas da empresa, entre outros. O planejamento de médio e longo prazo também é desenvolvido pela equipe deste nível.

No nível tático ocorrem as decisões atreladas ao planejamento das operações, as vendas e também o estoque. As decisões feitas aqui são baseadas nas instruções recebidas do nível estratégico e nos dados oriundos do nível operacional.

No nível operacional a abrangência das decisões é menor, porém muito importantes, pois elas podem acontecer em curto prazo e devem respeitar as definições dos níveis superiores. As decisões tomadas neste nível alteram a ordem ou sequência de produção dos itens, modificam a quantidade e variam em função do estoque, insumos, prazos dos pedidos, tempo disponível do equipamento para produzir, entre outros fatores. Um dos principais itens que sofre grande impacto das decisões tomadas é a ordem de execução da produção. Assunto que é estudado neste trabalho.

Em Viana (2017) podem ser observados dados a respeito da melhoria nos processos produtivos, proporcionando reduções expressivas no consumo de energia elétrica por tonelada de aço produzida, melhor aproveitamento dos insumos, redução na geração de resíduos e a reciclagem. Estas ações são fundamentais para a manutenção da indústria siderúrgica nacional no mapa mundial de produtores de aço. Todo este esforço pode ser desperdiçado se a produção não for ordenada corretamente. Uma sequência de produção desordenada (ou não otimizada) pode provocar perdas de tempo com *setup*, perdas de qualidade, gastos desnecessários que provocam o aumento do custo de fabricação, atrasos na produção, geração de estoque de materiais produzidos com muita antecedência, desabastecimento de insumos que foram consumidos fora do planejado.

Como citado, são diversos fatores que podem afetar a competitividade da empresa. Bretas (2013) destaca que o sequenciamento da produção, que também é conhecido como *scheduling*, é o processo que ordena as atividades de execução dos pedidos respeitando regras previamente estabelecidas. O *scheduling* contempla as restrições do equipamento de produção, as necessidades do cliente e possui um ou mais objetivos a serem alcançados.

Bretas (2013) observa também a discrepância entre a capacidade produtiva da siderurgia e o consumo de produtos siderúrgicos, provocando uma concorrência elevada entre os grandes e pequenos produtores.

Neste cenário de elevada competitividade e margens de lucro apertadas, faz-se necessário reduzir os desperdícios, em todas as etapas dos processos, inclusive a geração de estoques intermediários. Rodrigues (2012) destaca uma série de motivos que justificam

os trabalhos de melhoria no sequenciamento produtivo, como exemplo o prazo de entrega solicitado pelo cliente é cada vez mais apertado, aumento da flexibilidade da linha de produção frente as mudanças repentinas de demanda, mapeamento rápido dos problemas de qualidade e suas origens.

O problema estudado nesta pesquisa é conhecido na literatura como sequenciamento de máquina única (PSMU) ou *Single Machine scheduling* de uma linha de galvanização contínua. Trata-se de um tema tratado por alguns trabalhos, porém a abordagem, os objetivos e restrições adotadas são divergentes entre os autores, mas compatíveis com os cenários estudados em cada trabalho.

A elaboração de uma sequência de pedidos da linha de galvanização contínua deve ser executada levando em conta uma série de fatores, tais como a data de entrega exigida para o pedido, as características associadas à qualidade requisitada em cada um dos pedidos, seus atributos dimensionais e suas especificações térmicas. Também devem ser contempladas as demandas de aplicação de tratamento químico e uso do laminador.

Cada um destes fatores impacta diretamente no resultado da produção da linha de galvanização, influenciando no atendimento dos prazos de entrega e na geração de estoques de material processado, também afetando a qualidade final dos produtos processados.

Os principais impactos dos fatores associados à qualidade do processamento das bobinas de aço em uma linha de galvanização contínua podem ser vistos a seguir.

- Dimensões do material (largura e espessura): a temperatura do forno é afetada de acordo com a massa de aço que entra, em temperatura ambiente, para ser processada, e com isso alguns defeitos podem ocorrer, tais como rugas e repuxados (normalmente ocorrem quando há uma variação brusca da largura). Também podem ocorrer defeitos no processo de recristalização do aço, nos casos de bobinas sequenciadas com variações elevadas de espessura.
- Ciclo térmico a ser utilizado: afeta diretamente a recristalização do aço e como consequência de uma ordenação inadequada, pode-se ter materiais de dureza fora da especificação do cliente gerando prejuízos para a empresa e, em alguns casos, defeitos nos processos dos clientes. O ciclo térmico inadequado também provoca rugas e repuxados na tira de aço. Além dos defeitos citados, as variações de temperatura excessivas impactam no equipamento afetando o consumo de combustíveis, acelerando a degradação do material refratário existente no forno e também podem afetar a rotina de trabalho dos operadores de processo.

- Camada extra de proteção contra corrosão: esta etapa do processo, também chamada de tratamento químico, é aplicada a alguns materiais específicos. A inicialização e a finalização desta aplicação de proteção química não são instantâneas, principalmente no término da aplicação, produzindo um trecho da bobina com manchas até a completa finalização. Sendo assim, as bobinas que necessitem do tratamento devem ser agrupadas para evitar o aparecimento de manchas.
- Laminador de encruamento: utilizado para melhoria da planicidade e aplicação de rugosidade no material. Esta etapa não é utilizada em todos os materiais. É importante o agrupamento dos pedidos com uso do laminador pois a ativação do equipamento produz um trecho de material com planicidade e aspecto visual ruins. Esta parte do material é descartada.

Um sequenciamento que atenda aos critérios de qualidade deve considerar todos os fatores citados anteriormente, bem como, as características associadas ao calendário de produção. Portanto, o profissional responsável pela elaboração da agenda de produção deve levar em conta as características dos pedidos, as suas datas de chegada no estoque de entrada da linha de galvanização e as datas de entrega registradas no sistema, para que seja possível construir uma sequência de produção que minimize as antecipações, os atrasos e atenda aos critérios de qualidade apresentados.

A busca pela minimização dos atrasos deve ser constante pois violações às datas de entrega solicitadas pelos clientes podem reduzir a credibilidade da empresa como fornecedor, podem gerar multas contratuais ou até outras penalidades. Neste sentido, o estudo deste assunto se mostra relevante e importante, pois além da abordagem prática, atua em um dos fatores que afetam a competitividade das empresas, o sequenciamento. Este trabalho também permite a aplicação de conceitos teóricos desenvolvidos em outros trabalhos de pesquisa semelhantes.

Dada toda essa complexidade na tarefa de elaboração de um sequenciamento adequado dos pedidos da linha de galvanização contínua e também, por não terem sido encontrados trabalhos que tratam todos esses aspectos citados do sequenciamento, faz-se relevante a abordagem desse tema nessa pesquisa, que além de propor soluções para um problema real da indústria siderúrgica, contribui para o meio acadêmico através da aplicação de técnicas de modelagem matemática exata e meta-heurística. A metodologia adotada consiste no uso de programação linear inteira mista e algoritmos genéticos baseados em *BRKGA*. Desenvolvem-se modelos iniciais simples e com a evolução do trabalho, as restrições são adicionadas até alcançar todas as exigências do problema real.

1.2 Objetivos da Dissertação

O objetivo principal desta dissertação é desenvolver modelagens matemática e heurística capazes de resolver o problema de sequenciamento da produção de uma linha de galvanização contínua focando na minimização das antecipações e atrasos das entregas dos pedidos e atendendo a todas as exigências relacionadas aos critérios de qualidade dos pedidos. Para alcançar o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

1. Caracterizar a situação atual visando definir claramente o problema e os critérios de qualidade a serem considerados.
2. Elaborar formulações matemáticas e heurísticas para a sua solução, que representem adequadamente o cenário existente e implementá-las.
3. Validar o atendimento de todos os critérios de qualidade identificados.
4. Comparar os métodos exato e aproximado desenvolvidos avaliando os sequenciamentos gerados e o tempo de processamento.
5. Avaliar aplicabilidade dos modelos ao uso diário, ou seja, devem ter tempo de processamentos compatíveis com o uso diário.
6. Os resultados devem permitir a programação da produção das próximas 24h.
7. Avaliar a qualidade dos resultados obtidos.

1.3 Classificação e Estrutura do Trabalho

Esta dissertação, segundo Prodanov e Freitas (2013), pode ser classificada como uma pesquisa aplicada, pois busca gerar conhecimentos para aplicação prática na resolução de um problema específico e também é uma pesquisa explicativa e experimental, pois busca explicar os resultados através de experimentos controlados, identificando a relação entre as variáveis e os resultados encontrados. Além dessas classificações, do ponto de vista da forma de abordagem do problema, essa pesquisa pode ser classificada como quantitativa pois as variáveis do problema foram traduzidas em números. A divisão desse trabalho foi realizada em 5 capítulos como pode ser visto na figura 1.1 distribuídos da seguinte forma:

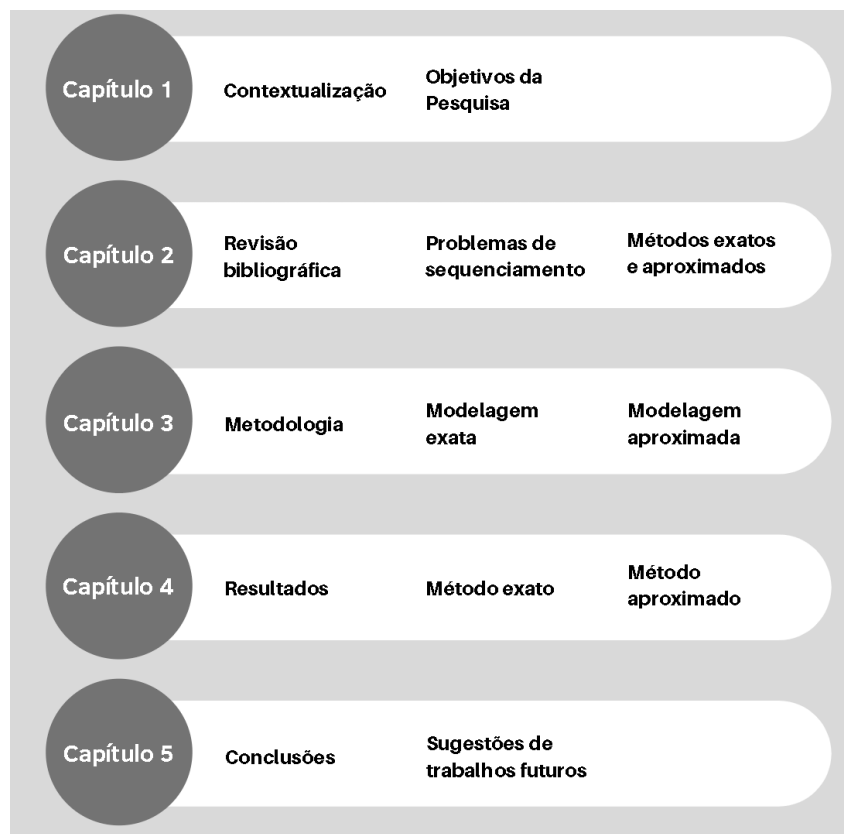


Figura 1.1: Estrutura da Dissertação.

Fonte: Próprio autor

- **Capítulo 2:** tem-se a revisão bibliográfica do assunto. Dividiu-se nas seguintes subseções: Etapas de produção do aço, Processo da linha de galvanização contínua, Problemas de sequenciamento, Sequenciamento de máquina única, Problemas associados a tempo de processamento e estoque, Problemas associados a data de conclusão, Problemas de minimização de atrasos e estoque. Também é feita uma introdução à programação linear inteira mista e por último apresentam-se as solu-

ções basadas em meta-heurística com a utilização do algoritmo genético de chaves aleatórias viciadas - *BRKGA* para a resolução do problema estudado.

- **Capítulo 3:** descreve a metodologia adotada no trabalho, os detalhes das modelagens matemáticas desenvolvidas para atendimento de cada uma das necessidades verificadas, também apresenta-se a construção do banco de dados utilizado no trabalho. Ainda neste capítulo são apresentadas as modelagens adotadas nos algoritmos de resolução do problema de forma exata e aproximada.
- **Capítulo 4:** são detalhados os resultados alcançados com os modelos exatos e heurísticos desenvolvidos. Também são identificados os limites computacionais para estes algoritmos. Ao final do capítulo são comparados os resultados obtidos pelos métodos exatos com os encontrados pela heurística.
- **Capítulo 5:** são apresentadas as conclusões obtidas e são sugeridos novos temas para pesquisas futuras.
- Por fim, são apresentadas as referências bibliográficas usadas no trabalho e os apêndices.

Capítulo 2

Revisão Bibliográfica

Dada a importância do aço na sociedade e o objetivo do trabalho, estudar algumas etapas do seu processo produtivo é importante para uma melhor compreensão do trabalho. A Seção 2.1 visa apresentar o processo de fabricação do aço e suas principais etapas de beneficiamento. São apresentadas na Seção 2.2 as características de uma linha de galvanização contínua, seus principais equipamentos e as características que necessitam de atenção durante o sequenciamento da produção da linha. São expostos na Seção 2.3 os problemas de sequenciamento gerais e o problema associado à linha de galvanização contínua. As técnicas disponíveis para solução dos problemas de sequenciamento apresentados na Seção 2.3 podem ser estudadas na Seção 2.4 e os conceitos associados as heurísticas para resolução do problema de sequenciamento podem ser vistos na Seção 2.5.

2.1 Etapas da Produção do Aço

A fabricação do aço pode ser dividida em algumas etapas, a saber: preparação, redução, refino e laminação (AÇO BRASIL, 2019). Em sua fabricação um dos primeiros processos produtivos é a coqueria, que consiste basicamente no beneficiamento do carvão através da destilação, dando origem ao coque. O coque servirá como combustível para redução do minério de ferro nos alto fornos. Outra matéria prima importante para os Alto fornos é o sinter, que é produzido na sinterização. O sinter consiste na aglomeração de finos de minério de ferro, coque, outros fundentes (calcário, por exemplo) e do próprio sinter com intuito de obter granulometria mínima destes insumos para o aproveitamento nos alto fornos. Estas duas etapas podem ser definidas como preparação.

A redução que ocorre no alto forno, origina-se a partir da reação do oxigênio aquecido com o coque, produzindo calor e fundindo a carga de minério, transformando-o em ferro

gusa. O ferro-gusa passará por algumas etapas de melhoramento até se tornar aço.

Uma das etapas de melhoramento é o refino, que faz a purificação do ferro gusa reduzindo o carbono e eliminando as impurezas. Esta etapa pode ocorrer em aciarias a oxigênio ou elétricas. Podem ser adicionadas sucatas metálicas juntamente com o ferro gusa durante processo para redução da necessidade energética e também do custo de produção. Após o refino, o aço pode ser moldado em lingotes ou levado aos equipamentos de lingotamento contínuo.

A etapa seguinte, chamada de laminação à quente, consiste no beneficiamento dos lingotes produzidos anteriormente, transformando-os em bobinas, chapas ou produtos não planos de acordo com a empresa e o mercado.

Após a laminação à quente, os materiais destinados a produção de aços planos (chapas ou bobinas) são processados em um outro equipamento, chamado de laminador de tiras a frio, ou LTF. Pires (2007) descreve, em seu capítulo 2, a estrutura e o funcionamento de um LTF com grande riqueza de detalhes.

No LTF a tira sofre uma deformação mecânica reduzindo a espessura entre 40% e 80%, de acordo com o equipamento ou cliente seguinte. Esta redução ocorre para obtenção de um material com melhor planicidade, espessura ajustada dentro das tolerâncias requisitadas e com superfície isenta de defeitos tais como riscos e ondulações. Uma das características do processo de redução à frio é a elevação da dureza do aço, efeito chamado de encruamento. Com a dureza elevada o aço acaba sendo pouco útil aos processos de estampagem, corte ou dobras, e por este motivo exige-se um novo processamento com objetivo de restabelecer as propriedades mecânicas. Esta nova etapa é o recozimento do material. Ela provoca o alívio nas tensões internas, o crescimento dos grãos do aço e o aumento da ductibilidade.

Muitos dos materiais planos são beneficiados em linhas que acrescentam algum tipo de revestimento ao aço proporcionando usos mais nobres (aplicações de maior valor agregado). Um dos exemplos é a aplicação de aços em embalagens de alimentos, e para isso ocorre o processo de estanhamento da chapa de aço. Outros lugares onde encontram-se chapas de aço revestidas são na indústria automobilística, indústria de equipamentos domésticos, como refrigeradores, microondas e móveis de aço. Outros tipos de revestimento que podem ser aplicados aos materiais são: eletro-galvanização, revestimento com ligas de alumínio e zinco, pré-pintura e galvanização por imersão a quente.

Os revestimentos são aplicados para a proteção do aço contra corrosão. Wiercinski

(2015) apresentou as origens da oxidação e consequente deterioração do aço, os problemas relacionados a esta reação com o ambiente e os potenciais riscos de falhas oriundas da corrosão. A aplicação de um revestimento de zinco em uma chapa de aço proporciona um aumento da vida útil do material base (aço) visto que o zinco resistente a corrosão entre 10 e 25 vezes mais que o aço em uma aplicação industrial. Na Seção 2.2 são apresentadas as principais características de uma linha de galvanização contínua, bem como os principais atributos que devem ser considerados para o seu sequenciamento adequado.

2.2 Processo da Linha de galvanização Contínua

De acordo com (RENN; STOTT; VASKO, 1999) a galvanização por imersão a quente ocorre simplificada da seguinte forma: o material é inserido na linha através de duas desenroladeiras, sofre uma soldagem, fundindo a ponta da nova bobina com o final da bobina em processamento. Após esta soldagem a nova tira de aço passa através da seção de limpeza para remover sujidades oleosas e finos de ferro, passando em tanques de limpeza alcalina, eletrolítica. Uma outra parte da limpeza acontece nos tanques com rolos escovadores que fazem uma ação mecânica de remoção das partículas remanescentes na tira de aço. O forno de recozimento é composto por quatro seções: pré-aquecimento, aquecimento, resfriamento lento e resfriamento a jato. Toda a atmosfera do forno é livre de oxigênio para evitar a oxidação da superfície do material. Após o forno o material entra no pote de zinco que possui sistema automático de aquecimento indutivo. A espessura do revestimento é ajustada por navalhas de ar/nitrogênio através da distância e pressão do sopro. As etapas de processamento seguintes consistem em:

- Etapa de resfriamento à temperatura ambiente: consiste na passagem da tira em um reservatório com água em temperatura ambiente forçando o seu resfriamento.
- Etapa de encruamento: esta etapa, que não é utilizada em todos os materiais, consiste na aplicação de uma força na chapa visando a impressão de rugosidade, melhoria da planicidade e do aspecto visual da tira. O equipamento utilizado por esta etapa é denominado laminador de encruamento.
- Etapa de estiramento: fase do processo que tem por objetivo tracionar mecanicamente o material provocando um estiramento de até 0,5%, garantindo a eliminação do patamar de escoamento e a planicidade requerida pelo cliente. A estiradeira normalmente é composta por um conjunto de rolos tensores e uma ou mais gavetas com rolos de apoio e trabalho.

- Etapa de passivação: fase também chamada de tratamento químico (TQ), consistindo em um sistema de tanques e rolos para aplicação de um produto químico sobre a tira já galvanizada, visando impedir o aparecimento de ferrugem branca.

Por último, o material é enrolado novamente em formato de bobina nos equipamentos chamados bobinadeiras ou enroladeiras, é embalado e despachado para os depósitos intermediários, antes do envio aos clientes. Pode ser visto na Figura 2.1 uma linha de galvanização contínua.

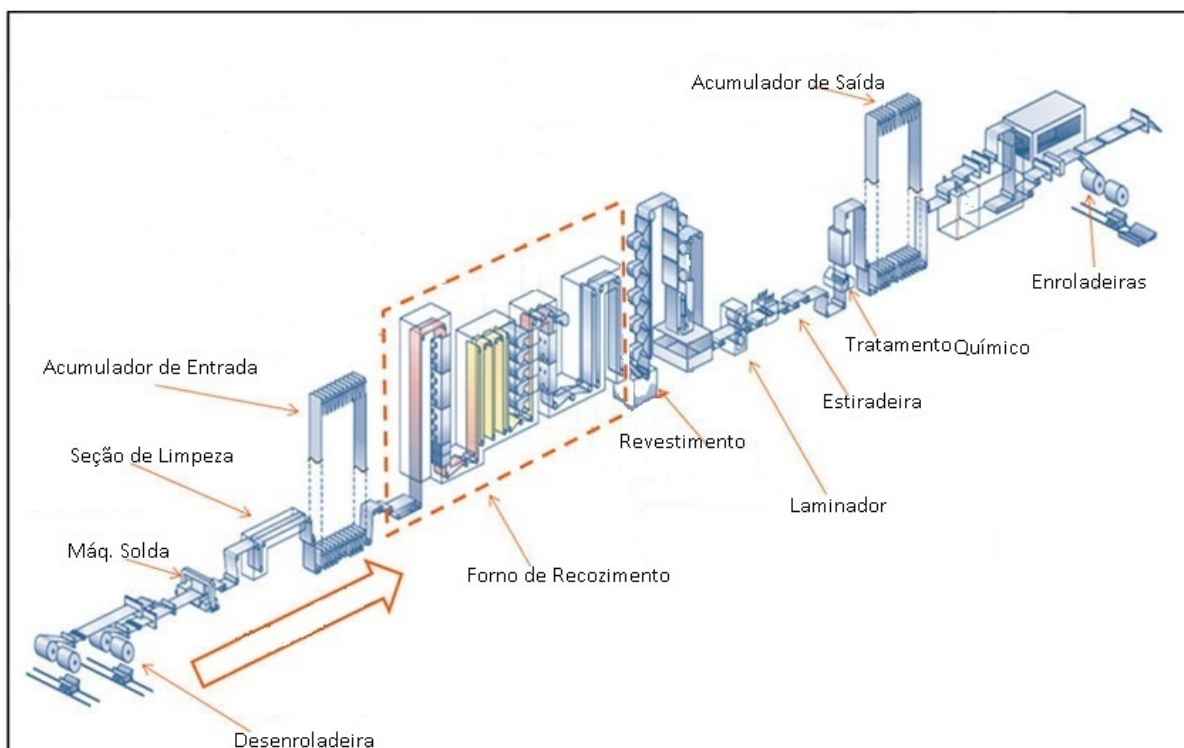


Figura 2.1: Representação da linha de galvanização contínua.

Fonte: Adaptado de Salzgitter AG (2020)

As etapas mencionadas são aplicadas a diversas bobinas sequencialmente e ininterruptamente. O fim de uma bobina é soldado ao início de outra para dar continuidade ao fluxo. Contudo, existem bobinas que não podem ou não devem ser soldadas indiscriminadamente a quaisquer outras. Esta impossibilidade de solda pode estar associada principalmente às características dimensionais ou ciclo térmico de recozimento. Já em alguns casos, não é economicamente produtivo sequenciar bobinas com alternância de requisitos, tais como aplicação do TQ ou uso do laminador de encruamento. Para a mudança de ciclo térmico existe a necessidade dos ajustes de temperatura no forno, que não são feitos instantaneamente. Há um tempo até as temperaturas se equilibrarem, e todo o material que passa no forno fora do equilíbrio adequado é considerado material processado fora dos parâmetros

de qualidade.

Para o correto processamento dos materiais existe a necessidade de observância de todas estas características para garantir a compatibilidade entre a bobina que está em processamento e a subsequente. Esta compatibilidade tem por objetivo a minimização do aparecimento de defeitos associados à qualidade mecânica e superficial do material.

Após a apresentação das etapas de produção do aço e das principais características da linha de galvanização contínua faz-se necessário apresentar os conceitos envolvidos nos problemas de sequenciamento. Para isso foi elaborada a Seção 2.3 que contém as definições importantes para o entendimento do problema de sequenciamento.

2.3 Problemas de Sequenciamento

Segundo Lopes et al. (2018) as atividades desenvolvidas pelas pessoas no ambiente produtivo vem se tornando cada vez mais complexas, por causa do desenvolvimento das tecnologias e do conhecimento humano. Com isso, tem-se a necessidade de encontrar soluções e decisões mais eficientes, observando o tempo a ser gasto e os resultados a serem alcançados. Os problemas de otimização são abordados em inúmeros trabalhos, como pode ser visto em Rodrigues et al. (2014) que utilizou programação linear na otimização dos custos da empresa ou em Pimenta (2008) que otimizou a produção em um processo do tipo *job shop* e também Tang e Gao (2009) que propôs um novo modelo matemático para sequenciamento de uma linha de galvanização contínua. Uma parte destes problemas é estudada em pesquisa de otimização combinatória, inclusive os problemas de alocação e de sequenciamento.

Os autores Baker e Trietsch (2009) apresentam a seguinte ideia sobre os problemas de sequenciamento: "O problema de sequenciamento consiste em um problema de agendamento especializado, no qual uma ordem dos trabalhos determina completamente um agendamento". Isto resume a essência dos problemas de sequenciamento, mas trata-se de uma simplificação didática apresentada.

A alocação de recursos ou equipamentos está presente nas organizações, tanto nas ordens de serviço quanto na utilização de recursos ou equipamentos para execução das atividades. De acordo com Pinedo (2008) e Ravetti (2003), o processo decisório de alocação de recursos para execução de tarefas em um determinado tempo utilizando uma ou várias máquinas sempre visa a otimização de um ou mais objetivos. A otimização pode visar por exemplo, a redução do atraso na entrega do item produzido, como proposto por

Tyagi, Tripathi e Chandramouli (2016), ou reduzir o tempo de espera para atendimento de navios, como foi estudado no trabalho de Martin et al. (2015), redução dos níveis de estoque intermediário e também nas áreas de tecnologia, como apresentado por Brandão (2015), que propôs resolver um problema de otimização de redes de comunicação WDM.

De acordo com Bretas (2013), o problema de sequenciamento, consiste em determinar a ordem de processamento dos pedidos, ou *jobs*. O objetivo principal está relacionado, normalmente à redução dos custos ou a maximização das receitas. Uma outra exigência para todas as organizações é a entrega dos produtos no prazo, pois o não atendimento deste requisito pode impactar em, além da redução da credibilidade da empresa, multas contratuais, perda de credenciamento na cadeia de fornecedores homologados, perdas financeiras associadas à eficiência produtiva e perda da competitividade. Por estes motivos muitos estudos já foram e estão sendo realizados nesta área de sequenciamento da produção. As diferenças entre estas pesquisas estão nos seus objetivos, características e também nas diferentes restrições específicas aplicadas em cada uma delas.

Conforme apresentado por Ravetti (2003), na grande maioria dos casos, os problemas de sequenciamento caracterizam-se por possuírem três conjuntos, sendo eles o conjunto de tarefas (T_j , onde j consiste na quantidade de tarefas), o conjunto de máquinas (P_m , onde m representa a quantidade de máquinas) e um conjunto de recursos adicionais (R_s). As máquinas envolvidas podem ser classificadas como paralelas ou dedicadas. Outro ponto a ser destacado sobre as máquinas é a sua classificação em função da velocidade, podem ser de velocidades iguais ou não e máquinas que a velocidade está diretamente relacionada com o *job* a ser executado.

Em seu livro, Pinedo (2008) explica que a definição de recursos e tarefas é dependente de cada caso ou organização a ser estudado. Alguns exemplos de recursos são: máquinas em um centro de serviços de usinagem e seu sequenciamento, como no trabalho elaborado por Rodrigues (2012), pistas de um aeroporto, entre outros. Exemplos de tarefas: peças a serem fabricadas, decolagens ou pousos no aeroporto, ou alocação de berços para carga e descarga de navios, conforme estudo realizado por Martin et al. (2015).

Segundo Baker e Trietsch (2009), os problemas de alocação começaram a ser estudados antes da Segunda Guerra Mundial. As primeiras publicações especializadas reconhecidas sobre sequenciamento aconteceram entre os anos 1950 e 1960. Muth e Thompson em 1963 e Conway Maxwell e Miller em 1967 publicaram algumas pesquisas sobre o tema.

As classificações dos problemas variam de acordo com a quantidade de máquinas, os fluxos envolvidos, número de etapas ou estágios e máquinas de cada estágio. Entretanto,

segundo Pinedo (2008), algumas características são comuns à maioria das tarefas dos problemas, independente da classificação. São elas:

- Campo contendo o tempo de processamento da tarefa de acordo com a máquina (p_{ij}) selecionada ou independente da máquina (p_j).
- Momento de solicitação ou disponibilização do pedido na linha de produção (r_j), que significa o momento onde a tarefa j poderá ser iniciada.
- Momento de entrega (ou data compromisso) (d_j). É a data de entrega solicitada pelo cliente. Entregas após esta data são consideradas atrasadas e em alguns casos entregas antes desta data são classificadas como antecipadas.

A notação tripla $\alpha | \beta | \gamma$ apresentada por Brucker (2006), Pinedo (2008) e Fuchigami (2016) é frequentemente utilizada para classificar os problemas de sequenciamento e cada um dos campos possui um significado bem definido, a saber: o campo α caracteriza o tipo de máquina envolvida no problema e possui somente um valor, o campo β apresenta os detalhes de processamento e as restrições associadas. Pode estar vazio, com um ou vários valores. Por último tem-se o campo γ , que contém o objetivo a ser alcançado e normalmente é composto por somente um valor.

Em seu trabalho (PINEDO, 2008) apresenta alguns exemplos de aplicações onde foram utilizadas técnicas de otimização para resolução dos problemas. Entre eles destacam-se a fábrica de embalagens de papel e a indústria de semicondutores.

No trabalho de Fuchigami (2016) foi apresentado um resumo dos tipos de problemas de sequenciamento destacando o número de máquinas, rotas e estágios de cada um deles. O problema é considerado *Flow Shop* quando possui múltiplos estágios de produção sequenciados, uma única máquina por estágio e os *jobs* podem ser processados em ordens diferentes em cada estágio. O problema do tipo *Flow Shop* Permutacional difere do *Flow Shop* comum, exigindo a mesma ordem de processamento de *Jobs* seja mantida em todos os estágios. Já em problemas do tipo *Job Shop* cada *job* passa pelos estágios em uma ordem específica, indicando dependência entre estágios. Em problemas do tipo *Open Shop* não há dependência entre estágios e, portanto, os *jobs* não possuem uma ordem pré-definida de estágios a ser seguida. Estas características podem ser vistas na Figura 2.2.

Nos estudos realizados por Yen e Wan (2003), Adamu e Adewumi (2014) podem ser encontradas uma gama considerável de livros e estudos das diversas etapas do desen-

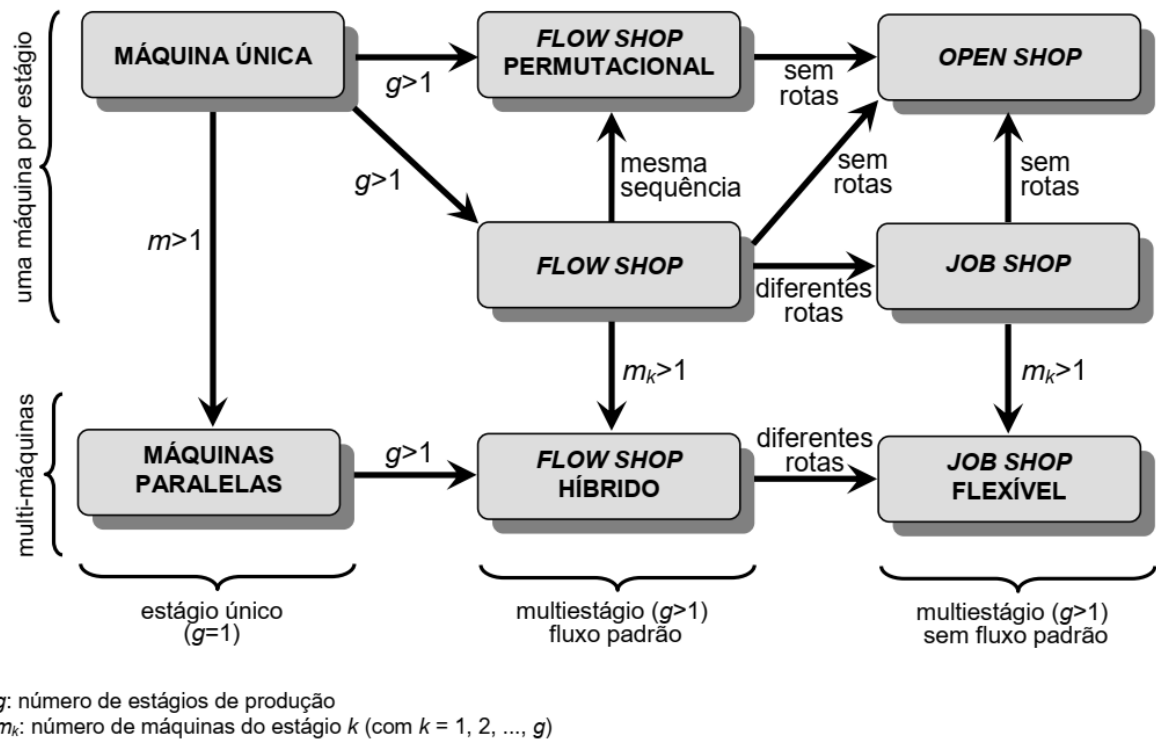


Figura 2.2: Problemas de sequenciamento e suas características.

Fonte: Fuchigami (2016)

volvimento dos mecanismos de sequenciamento. Por exemplo, existem trabalhos focados nos aspectos computacionais do sequenciamento, aspectos estocásticos de filas com prioridades, complexidade computacional de sequenciamentos determinísticos, sequenciamento determinístico multi-objetivos com restrições de recursos, foi feita uma análise de muitas heurísticas para sequenciamento de problemas do tipo *job shop*, análises de algoritmos de sequenciamento determinístico e programação por restrições. Os autores também citam estudos que abrangeram inteligência artificial e programação com restrições para elaboração de sequenciamentos.

Os autores Ravetti (2003) e Pinedo (2008) apresentam algumas funções clássicas estudadas e amplamente utilizadas. O momento da conclusão de um *job* j em uma determinada máquina i é definido por C_{ij} e o tempo total para finalização deste mesmo *job* j em todo o sistema é definido como C_j . Uma função objetivo relacionada aos momentos de conclusão é chamada de *lateness*, sendo calculada como na Equação (2.1) onde L_j é o resultado da diferença entre o momento de conclusão do pedido C_j e a data desejada d_j

$$L_j = C_j - d_j, \quad (2.1)$$

quando for positiva significa atraso na conclusão do *job*, e caso contrário representa um *job* concluído antes da data desejada. Deste cálculo surgem outras possíveis funções objetivo, a T_j - *tardiness* que é calculada como podem ser visto nas Equações (2.2) e (2.3)

$$T_j = \max(C_j - d_j, 0) \quad (2.2)$$

ou

$$T_j = \max(L_j, 0) \quad (2.3)$$

e a outra função *Penalidade*, que indica se o *job* j está atrasado ou não é calculada na Equação (2.4).

$$U_j = \begin{cases} 1 & \text{se } C_j > d_j (\rightarrow \text{atrasado}) \\ 0 & \text{caso contrário} (\rightarrow \text{dentro do prazo}) \end{cases} \quad (2.4)$$

Outras variações permitidas para os campos α | β | γ podem ser encontradas nos trabalhos citados de Pinedo (2008) e Fuchigami (2016).

A seguir tem-se exemplos de funções objetivo frequentemente aplicados nas pesquisas sobre sequenciamento.

- *Makespan* (C_{max}): momento de conclusão do último pedido ou duração do sequenciamento.
- *Lateness* (L_{max}): maior diferença entre a data de conclusão e a data desejo de cada um dos *jobs* do sequenciamento.
- Total de antecipação e atraso, como expressado na Equação (2.5), onde E representa a antecipação e T o atraso do pedido j .

$$\sum_{j=1}^n E_j + \sum_{j=1}^n T_j \quad (2.5)$$

O trabalho desenvolvido por Jr. e Tseng (2001) aborda a minimização do *Makespan* e também do tempo médio de fluxo em um problema de *Flow Shop* com tempo de *setup* distintos para as máquinas. Duas novas implementações para o problema em programação linear inteira mista foram propostas. Na primeira foi aplicado o modelo de atribuição de posição dos *jobs* e a segunda modelagem foi utilizado um conjunto de restrições que avaliava par a par os *jobs* garantindo que um trabalho só poderia iniciar em uma máquina m se todos os *jobs* anteriores a ele, em qualquer posição da fila de execução, tivessem

finalizado o seu processamento nesta mesma máquina m e o respectivo *setup* já estivesse finalizado. Este foi um artigo de uma série de estudos que os autores fizeram sobre os problemas com tempos de *setup* independentes da sequência.

Nas pesquisas citadas por Pinedo (2008) são apresentadas outras formas de avaliação da performance do sequenciamento e uma delas trata as penalidades por antecipação e atraso com pesos (custos) diferentes. Para isso, calcula-se a antecipação ou *earliness* como $E_j = \max(d_j - C_j, 0)$ e o atraso, como apresentado na Equação (2.2) e em seguida faz-se a multiplicação destes valores pelos seus respectivos pesos w , conforme demonstrado na Equação (2.6). Estes pesos podem representar, num cenário industrial, os custos de estocagem da produção antecipada ou também multas contratuais por atraso na entrega dos produtos aos clientes ou até o custo de perda da janela de tempo para envio de materiais por via marítima. Esta abordagem foi utilizada no trabalho de Bustamante e Carvalho (2006).

$$\sum_{j=1}^n w_1 E_j + \sum_{j=1}^n w_2 T_j \quad (2.6)$$

Segundo Pinedo (2008), o problema exposto na Equação (2.6) é classificado como \mathcal{NP} -Difícil. Nesta dissertação são aplicados os conceitos do sequenciamento de máquina única. Para outros tipos de problemas de sequenciamento, recomenda-se a leitura dos trabalhos de *Flow Shop* elaborados por Jr. e Tseng (2001) e Parveen e Ullah (2011). Para *job shop* recomenda-se o trabalho de Lopes et al. (2018) e para sequenciamento de máquinas idênticas paralelas sugere-se o estudo realizado por Bretas (2013).

2.3.1 Sequenciamento de Máquina Única

Foi apresentada no trabalho de Yousefi e Yusuff (2012) uma definição a respeito do Problema de Sequenciamento de Máquina Única (PSMU): consiste no processo de alocação de um grupo de *jobs* a um recurso ou uma máquina. A ordenação das tarefas deve ser tal que algum indicador de performance possa ser avaliado.

De acordo com Baker e Trietsch (2009), o modelo de problema de sequenciamento mais simples é onde existe somente um equipamento (ou máquina) e todos os tempos de processamento dos trabalhos (*jobs*) são determinísticos. Eventualmente pode-se resolver o sequenciamento de máquina única incorporando seus resultados nos problemas maiores. Em problemas de múltiplos estágios, podem aparecer etapas consideradas gargalos

(*bottlenecks*) e o sequenciamento de máquina única pode resolver ou auxiliar na resolução fornecendo informações ou propriedades deste gargalo para o sistema de sequenciamento com maior complexidade. Um problema com esta característica foi estudado por Mainieri (2014), onde foi necessário um tratamento para a etapa gargalo antes das demais etapas do sequenciamento.

Com o desenvolvimento da filosofia de produção *Just in Time (JIT)* ocorreram diversas mudanças estratégicas nas linhas de produção, passando a pensar na eliminação do desperdício de tempo durante as atividades de produção. Deste modo iniciou-se a avaliação dos pedidos produzidos com atraso e também antes da hora. Esta preocupação com os atrasos e, agora, com as antecipações estimulou diversas pesquisas sobre sequenciamento eficiente Rodrigues (2012).

Kim e Yano (1992) também citam alguns exemplos que justificam o esforço de minimização dos atrasos e também das antecipações. Como exemplo prático ele apresenta uma situação onde a data de embarque dos produtos já está definida antes do processamento deles na linha de produção e, com isso, tem-se uma data alvo a ser seguida pela produção. Caso sejam produzidos antes desta data terão acréscimo do custo de estocagem, e por outro lado, se houver atraso, incorrerão custos extras de transporte emergencial e multas contratuais, por exemplo. Somente o fato de a empresa correr o risco de perder credibilidade com os clientes em função de atrasos já justifica o trabalho, porém, se forem considerados também as premissas da produção JIT, o desperdício ocasionado pela estocagem de produtos produzidos antes da data compromisso precisa ser minimizado ou eliminado quando possível. No trabalho de Yousefi e Yusuff (2012) são apresentadas pesquisas alinhadas à produção *JIT* desde 1977. A partir daí, outros trabalhos de pesquisa foram elaborados com diversos indicadores de performance.

No trabalho realizado por Yousefi e Yusuff (2012) foi proposto o uso de algoritmo genético para a resolução do problema de sequenciamento de máquina única com penalidades por antecipação e atraso com data compromisso comum. Este tipo de problema é classificado como \mathcal{NP} -Difícil. O trabalho foi realizado em um problema real com 16 *jobs* e buscava a melhor sequência de execução. Algumas características foram consideradas em seu algoritmo, como por exemplo a inserção de um tempo de espera para o primeiro pedido, não foram permitidos tempos entre os *jobs* e não havia tempo de *setup*. Para Yousefi e Yusuff (2012) os resultados foram satisfatórios, encontrando a melhor solução em menos de vinte gerações do algoritmo.

Para a continuação deste trabalho faz-se necessário recordar algumas definições que

Pinedo (2008) e Baker e Trietsch (2009) apresentaram como condições mínimas para um problema de sequenciamento de máquina única:

1. Todos os n jobs estão disponíveis no instante inicial.
2. A máquina processa somente um job por vez.
3. O tempo de *setup* é considerado no tempo de processamento e não depende da sequência de execução adotada.
4. O tempo de processamento é conhecido e bem definido.
5. A máquina não possui momentos de indisponibilidade.
6. Sempre que existirem trabalhos aguardando, serão processados, não existindo máquina ociosa com jobs aguardando processamento.
7. Não há preempção, isto é, a execução de nenhum job poderá ser interrompida antes do seu término.

Sendo estas condições atendidas tem-se um conjunto de soluções gerado a partir da permutação dos n jobs ou seja $n!$ sequências dos n jobs.

Os trabalhos de Baker e Trietsch (2009) e Fuchigami (2016) explicam que para a resolução dos problemas de sequenciamento são necessários alguns dados no início do problema, que chamam de dados de entrada. Nesta dissertação, os dados de entrada são escritos em letras minúsculas, por exemplo: p_j - tempo de processamento da tarefa j , d_j - data compromisso da tarefa j ou r_j - momento de chegada do pedido na linha de produção. As informações geradas como resultado do sequenciamento são expressas em letras maiúsculas, por exemplo: C_j - momento de conclusão da tarefa j . Nos trabalhos citados constam diversos exemplos de dados de entrada, dados de saída e funções aplicáveis aos tipos de problemas de sequenciamento. A avaliação dos resultados alcançados com o sequenciamento normalmente é feita observando o momento de conclusão da tarefa C_j . A seguir estão alguns exemplos:

- F_j : tempo que a tarefa j permaneceu no sistema. Calculada como $C_j - r_j$.
- L_j : atraso na conclusão da tarefa j . Quanto tempo após a data compromisso d_j .
- $F = \sum_{j=1}^n F_j$: Tempo total de processamento.

- $T = \sum_{j=1}^n T_j$: Tempo total de atrasos.
- **Makespan**: calculado como $C_{max} = MAX(C_J)$ ou $F_{max} = MAX(F_j)$.

Diversas abordagens podem ser encontradas na literatura para os PSMU. Kim e Yano (1992) apresentam um resumo das abordagens e seus objetivos. Uma delas consiste na minimização dos custos associados à antecipação e atraso com mesmo valor de penalidade. Um outro problema relatado apresenta uma característica peculiar, a mesma data de compromisso para todos os pedidos. Nesta abordagem o autor apresenta alguns trabalhos com diferentes funções objetivo, como por exemplo a minimização do desvio médio quadrático entre a data de conclusão e data desejada, ou então a minimização da diferença absoluta entre a data de conclusão e data desejada. Nos dois exemplos foram utilizadas penalidades para antecipação e atraso iguais para todos os *jobs*. Também são expostos trabalhos que consideraram pesos distintos para os *jobs* e até penalidades proporcionais ao tempo de processamento deles. Também foram analisados outros trabalhos que consideravam datas de compromisso distintas entre os *jobs*. Em seu trabalho foi considerada a inserção de tempos de espera entre os *jobs*. Esta característica não é aceita nesta dissertação pois, como se trata do sequenciamento de uma linha de produção contínua, a existência de tempos de espera entre os *jobs* seria equivalente a parar a linha para aguardar o momento de início do processamento do *job* seguinte.

Rajabzadeh, Vahdani e Arabasadi (2014) propuseram o PSMU com uma abordagem incomum. Consideraram, além do sequenciamento, a entrega dos produtos aos clientes. A modelagem foi desenvolvida para executar os *jobs* e decidir entre: (a) enviar diretamente para os clientes ou (b) agrupar os *jobs* prontos em um conjunto para posterior envio aos clientes. O envio dos *jobs* é feito utilizando diversos veículos de capacidades distintas. Cada um dos *jobs* agrupados ocupa uma parte da capacidade dos veículos. O objetivo do trabalho foi desenvolver uma solução usando programação linear inteira mista que escolhesse a melhor combinação dos custos de transporte, atraso e armazenamento. Por se tratar de um problema complexo, foi proposto o uso de uma heurística para resolver as instâncias de maior porte. Nas pesquisas realizadas para o trabalho relataram que não encontraram literatura que considerasse os dois problemas conjuntamente na mesma modelagem. Os autores avaliaram que a heurística proposta conseguiu chegar em resultados aceitáveis em um tempo de execução de poucos segundos.

São apresentados no *survey* elaborado por Adamu e Adewumi (2014) muitas abordagens para o PSMU. O primeiro caso estudado é o problema do tipo $1||\sum U_j$ ou $1||\sum w_j U_j$, que consiste na minimização do número de *jobs* atrasados e na segunda expressão, con-

siderando o peso (ou importância) w_j de cada *job*. Para este problema o autor elencou algumas pesquisas com abordagens distintas, como por exemplo programação inteira com a estrutura do *tableau Simplex*. A pesquisa comprovou que se trata de um problema do tipo \mathcal{NP} -Difícil. Um dos trabalhos estudados foi capaz de resolver instâncias com 85 *jobs* em um tempo razoável utilizando *Branch and Bound* e regras de dominância entre os *jobs*. Outro trabalho analisado propôs uma heurística combinada com um algoritmo exato. Ele utilizava *Branch and Bound* com os limites obtidos do algoritmo *knapsack* substituto (*surrogate knapsack*). Com esta combinação ele conseguiu chegar em valores ótimos com poucos segundos para instancias com 2500 *jobs*. Para a solução exata desta mesma instância foram necessários aproximadamente 6 minutos. Para problemas do tipo $1|r_j|\sum U_j$ e $1|r_j|\sum w_j U_j$, que consiste na minimização do número de *jobs* atrasados com datas de liberação distintas (com peso w_j para cada *job*) foi comprovado que esta variação do PSMU também é \mathcal{NP} -Difícil. Um dos estudos apresentou um algoritmo *branch and check* onde uma programação inteira com relaxação foi resolvida com *Branch and Bound*. Conseguiu sucesso em instâncias com até 140 *jobs*.

Em Pinedo (2008) e Baker e Trietsch (2009) os autores citaram diversas abordagens para os problemas de sequenciamento de máquina única. Apresentam-se nas Seções 2.3.2 e 2.3.3 alguns detalhes das abordagens orientadas a tempo de processamento e a data de conclusão.

2.3.2 Problemas Associados a Tempo de Processamento e Estoque

O tempo gasto para processar um pedido j é o F_j . Se for o menor tempo possível, significa que foi alcançado o melhor resultado para o problema. A função objetivo neste caso é a minimização do *makespan*. Um outro possível objetivo pode ser a minimização do estoque, que indiretamente está associado ao tempo de processamento dos *jobs*. Quanto maior for o número de entregas menor será o tamanho do estoque em processo (RODRIGUES, 2012).

Baker e Trietsch (2009) apresentam a função $J(t)$ que representa o total de *jobs* no sistema no instante t . Nos problemas de máquina única, no instante $t = 0$ tem-se um total de n *jobs* aguardando processamento. Já no instante $t = p(1)$ o total de *jobs* aguardando será diminuído em um, pois o primeiro *job*, $j = 1$, acabou de ser processado com tempo de execução $p(1)$. Seguindo este raciocínio, a função $J(t)$ representa o comportamento degrau, sendo diminuído à medida que os *jobs* são concluídos. Ela chegará em zero quando

$$o\ t = \sum_{j=1}^n p_j.$$

Mesmo com o tempo total de processamento dos pedidos não sendo influenciado pela ordem de execução dos *jobs*, pode-se encontrar uma melhor sequência que objetive a diminuição da quantidade de *jobs* aguardando processamento. Esta abordagem é descrita na literatura como minimização do estoque em processamento. A estratégia adotada consiste em processar primeiro os *jobs* cujo tempo p_j sejam os menores e com isso mais *jobs* são concluídos no início do ciclo (RODRIGUES, 2012).

Esta estratégia de minimização do estoque em processo pode ser aplicada em sistemas dinâmicos onde novos *jobs* aparecem com o passar do tempo, em sistemas probabilísticos onde o tempo de processamento é incerto, ou em sistemas onde o custo de estocagem dos materiais aguardando processamento seja muito elevado (BAKER; TRIETSCH, 2009).

Uma outra abordagem aplicada a estes problemas é a minimização do tempo de processamento considerando a importância do pedido, ou, conforme notação apresentada por Pinedo (2008), $1||\sum_{j=1}^n w_j F_j$. A importância do pedido é registrada no vetor w_j , que é multiplicada pelo tempo total de processamento do pedido, o F_j . A Equação (2.7) é a função objetivo sendo o w_j a importância que pode representar a penalidade do atraso na entrega dos pedidos ou o custo de armazenagem dos produtos no estoque (BAKER; TRIETSCH, 2009).

$$F_w = \sum_{j=1}^n w_j F_j \quad (2.7)$$

Outra abordagem associada a data de conclusão do pedido consiste na minimização do número de *jobs* atrasados ou $1||\sum_{j=1}^n U_j$. Neste caso processa-se o trabalho cuja data de compromisso está mais próxima de ser alcançada. Esta função está representada na Equação (2.8).

$$U = \sum_{j=1}^n U_j \quad (2.8)$$

As abordagens apresentadas sobre *makespan* ou tempo de fluxo visam a minimização do tempo total gasto para concluir o *job* ou a minimização do tempo em que o produto ficou no(s) estoque(s) intermediário(s) aguardando para ser processado. Vale notar que não há qualquer preocupação com a data de entrega nestas formulações. Na Seção 2.3.3 tem-se outras abordagens que consideram a data de compromisso do pedido ou *due date*.

2.3.3 Problemas Associados a Data de Conclusão

No trabalho de Pinedo (2008) é citado outro grupo de funções objetivo que estão relacionadas com a proximidade da data de conclusão do pedido e da data desejada informada pelo cliente. Uma destas funções já foi apresentada na Equação (2.4), sendo comumente chamada de *lateness*. Ela calcula a precisão do sequenciamento em concluir o trabalho na data informada pelo cliente. A Equação (2.4) tendo um resultado positivo significa conclusão atrasada do *job*, em caso de valor negativo conclui-se que o *job* foi finalizado antes da data requisitada. Nos casos de valores zerados tem-se o melhor resultado, que consiste no acerto entre as datas de conclusão e desejada. Nos problemas de sequenciamento com este objetivo é possível utilizar a técnica conhecida na literatura como *shortest processing time (SPT)* ou *Menor tempo de processamento*. Ela ordena os *jobs* de acordo com o tempo restante até a data desejada. Caso existam pesos w_j ou penalidades associados aos pedidos, a função objetivo sofre modificação passando a considerar estes pesos. O tema foi abordado no *survey* elaborado por Adamu e Adewumi (2014). A Equação (2.8) modificada passa a ser escrita como a Equação (2.9).

$$U = \sum_{j=1}^n w_j U_j. \quad (2.9)$$

Pode ser necessário avaliar somente o atraso das entregas. Nestes problemas a função objetivo visa minimizar o tempo total de atraso que pela notação inicial definida pode ser escrito como $1||\sum_{j=1}^n T_j$. Os problemas deste tipo apresentam como característica relevante o fato de o tempo de atraso do *job* j não ser função linear do seu respectivo tempo de conclusão. Sendo assim, para resolução deles, normalmente usam-se as técnicas de otimização combinatória ou heurísticas (PINEDO, 2008; BAKER; TRIETSCH, 2009). Como exemplo desta abordagem podem ser citados os trabalhos de Valente et al. (2011) e Tyagi, Tripathi e Chandramouli (2016). Apresenta-se na Equação (2.10) o problema na forma de atrasos ponderados.

$$T_w = \sum_{j=1}^n w T_j \quad (2.10)$$

Outro exemplo de função objetivo é a $1||\sum_{j=1}^n w_j C_j$ ou Soma ponderada do tempo total de conclusão dos pedidos. Nesta função ocorre a soma dos momentos de conclusão C_j dos pedidos multiplicados pelo seu respectivo peso w_j . A Equação (2.11) apresenta este cálculo. Estes tipos de problemas são resolvidos pelo método *WSPT* - (*Weighted*

Shortest Processing Time) ou Menor tempo de processamento ponderado, e a ponderação é calculada como w_j/p_j . A classificação ocorre de forma decrescente baseada nos resultados da ponderação e do momento de conclusão C_j (PINEDO, 2008).

$$C_w = \sum_{j=1}^n w_j C_j \quad (2.11)$$

2.3.4 Problemas de Minimização de Atrasos e Estoque

No trabalho de Bustamante e Carvalho (2006) são apresentados dois modelos matemáticos para o problema de sequenciamento de máquina única. Nele são consideradas penalidades distintas para a antecipação e para o atraso dependentes dos *jobs*, também são permitidos tempos ociosos entre os *jobs*. Considera-se a existência de tempo de preparação (*setup*) entre os *jobs*. A notação adotada pelos autores está descrita a seguir.

- i : *job* que será sequenciado.
- P_i : tempo de processamento do *job* i .
- T_i^{due} : data desejada para término da produção do *job* i (*due date*).
- S_{ij} : tempo de preparação para produzir o *job* j depois do *job* i .
- I : conjunto dos *jobs* a serem sequenciados.
- t_i : variável que determina a data de início da produção do *job* i .
- l : posições possíveis na sequência de produção.
- F^1 : custo por antecipação (estoque) por unidade de tempo.
- F^2 : custo por atraso no atendimento (produção) por unidade de tempo.
- L : conjunto das posições possíveis dentro da sequência de produção.
- Z_{il} : variável binária que determina, quando igual a 1, se o *job* i foi alocado na posição l na sequência de produção, e caso contrário quando igual a 0.
- t_l : variável que determina a data de início da produção da posição l .
- h_l^1 : variável que determina o tempo de antecipação da produção da posição l .
- h_l^2 : variável que determina o tempo de atraso da produção da posição l .
- W_{ijl} : variável binária que determina se os *jobs* i e j foram alocados em posições subsequentes, sendo igual a 1 se o *job* i foi alocado na posição l e o *job* j foi alocado na posição $l+1$ e 0, se não.

Uma representação dos valores de antecipação h_i^1 e atraso h_i^2 pode ser vista na Figura 2.3.

Observa-se na Figura 2.3 que o *Job* i foi concluído antes da data desejo, ou seja, com

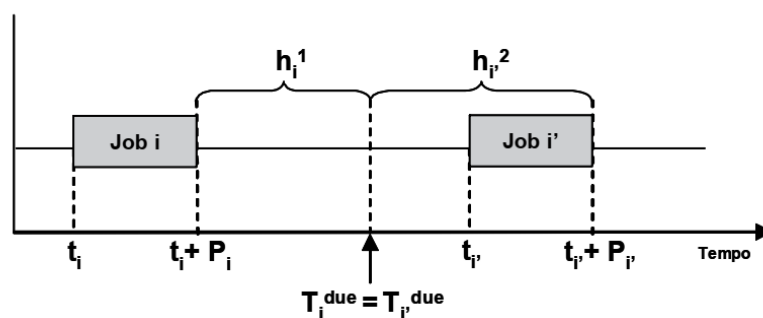


Figura 2.3: Valor do atraso e da antecipação.

Fonte: Bustamante e Carvalho (2006)

antecipação h_i^1 e o *Job i'* foi finalizado com atraso $h_{i'}^2$. Neste exemplo, ambos possuíam a mesma data desejo T_i^{due} .

Para simplificação didática, a notação e modelagem utilizadas como base desse trabalho sofreram algumas modificações. A listagem a seguir destaca todas estas modificações.

Modelo	Novo	Descrição
P_i	p_i	- tempo de processamento do <i>job i</i> .
T_i^{due}	d_i	- data desejada para término da produção do <i>job i</i> .
I	P	- conjunto dos <i>jobs</i> a serem sequenciados.
S_{ij}	ST_{ij}	- tempo de preparação para produzir o <i>job j</i> após o <i>job i</i> .
F^1	α	- custo por antecipação (estoque) por unidade de tempo.
F^2	β	- custo por atraso na produção por unidade de tempo.
L	F	- conjunto das posições dentro da sequência de produção.
t_l	S_l	- variável com a data de início da produção da posição l .
h_l^1	E_l	- tempo de antecipação da produção da posição l .
h_l^2	T_l	- tempo de atraso da produção da posição l .
—	—	- Variáveis mantidas com a mesma identificação.
i	i	- <i>job</i> que será sequenciado.
Z_{il}	Z_{il}	- variável binária que determina, quando igual a 1, se o <i>job i</i> foi alocado na posição l na sequência de produção e 0 caso contrário.
W_{ijl}	W_{ijl}	- variável binária que determina se os <i>jobs i</i> e j foram alocados em posições subsequentes, sendo igual a 1 se o <i>job i</i> foi alocado na posição l e o <i>job j</i> foi alocado na posição $l+1$ e 0, se não.

A formulação proposta pelos autores Bustamante e Carvalho (2006) e adaptada con-

forme citado previamente pode ser vista a partir da Equação (2.12). Esta formulação foi utilizada como base desta dissertação.

$$\text{Min } Z = \sum_{l \in F} \alpha E_l + \sum_{l \in F} \beta T_l \quad (2.12)$$

S.a.

$$S_{l+1} - S_l - \sum_{i \in P} p_i Z_{il} - \sum_{i, j \in P, i \neq j} ST_{ij} W_{ijl} \geq 0, \forall l \in F, l < |F| \quad (2.13)$$

$$\sum_{i \in P} Z_{il} = 1, \forall l \in F, \quad (2.14)$$

$$\sum_{l \in F} Z_{il} = 1, \forall i \in P \quad (2.15)$$

$$W_{ijl} - Z_{il} - Z_{j(l+1)} \geq -1, \forall i, j \in P, i \neq j, l \in F, l < |F| \quad (2.16)$$

$$\sum_{i, j \in P, i \neq j} \sum_{l \in F, l < |F|} W_{ijl} = |F| - 1 \quad (2.17)$$

$$S_l + \sum_{i \in P} p_i Z_{il} + E_l - T_l - \sum_{i \in P} d_i Z_{il} = 0, \forall l \in F \quad (2.18)$$

$$S_l \geq 0, \forall l \in F \quad (2.19)$$

$$E_l \geq 0, \forall l \in F \quad (2.20)$$

$$T_l \geq 0, \forall l \in F \quad (2.21)$$

$$W_{ijl} \in \{0, 1\}, \forall i, j \in P, l \in F \quad (2.22)$$

$$Z_{il} \in \{0, 1\}, \forall i \in P, l \in F \quad (2.23)$$

Pode ser observado que a função objetivo apresentada em (2.12) busca minimização das antecipações, ou como os autores apresentam, minimização dos custos de estoque e também a redução dos atrasos. O conjunto de restrições (2.13) exige que o início (S_{l+1}) do *job* da posição $l + 1$ aconteça após a conclusão do *job* anterior e da execução do *setup* ST_{ij} . As Restrições (2.14) e (2.15) garantem que cada posição l seja ocupada por somente um *job* i e que cada *job* ocupe somente uma posição l da fila. As posições dos *jobs* são identificadas através das Equações (2.16) e (2.17). A medição do atraso ou antecipação do *job* que ocupa a posição l da fila de execução F é feita na equação (2.18). O domínio das variáveis é definido nas Restrições (2.19), (2.20), (2.21), (2.22) e (2.23).

Esta formulação proposta por Bustamante e Carvalho (2006) é linear com variáveis inteiras. Dada a importância deste tipo formulação para este trabalho, na Seção 2.4 são apresentados alguns dos principais conceitos deste tipo de formulação e algumas das

estratégias usadas para sua resolução.

2.4 Programação Linear Mista

O autor Winston (2004) define em seu trabalho o que é um problema de programação linear (PPL). Ele caracteriza o PPL como um problema de otimização que:

- Busca alcançar um valor máximo (ou mínimo) para uma função linear (chamada de função objetivo) composta por variáveis de decisão.
- O valor das variáveis de decisão deve satisfazer todas as restrições do problema. Cada uma das restrições podem ser equações ou inequações lineares.
- Os valores válidos das variáveis de decisão devem ser especificados nas restrições. Podem ser valores não nulos, positivos ou em casos específicos, toda a faixa de valores positivos, nulo ou negativos.

A programação linear pode ser dividida em três classes de problemas. Cada classe agrupa os problemas que apresentam as seguintes características:

1. Todas as variáveis contínuas: problemas de programação linear PPL.
2. Todas as variáveis inteiras: problemas de programação inteira PPI.
3. Variáveis contínuas e inteiras: problemas de programação mista.

Para a resolução dos problemas de programação linear PPL pode-se utilizar o algoritmo *simplex*. Este método teve origem no ano de 1947, nos Estados Unidos, e atribui-se a George Bernard Dantzig Ourisson a sua primeira formulação. O método consiste em um procedimento iterativo que, a partir de uma solução básica factível, faz uma busca na vizinhança por soluções básicas factíveis melhores até encontrar o valor ótimo para a função objetivo (BELFIORE; FÁVERO, 2013). Pode ser visto na Figura 2.4 a forma tabular deste método com a solução inicial de um problema.

Para a solução de problemas inteiros ou mistos existem múltiplas abordagens. Brucker (2006) e Santos (2016) destacam os algoritmos *Branch-and-bound*, Plano de Cortes e *Branch-and-cut*.

1. *Branch-and-bound* - elabora uma enumeração implícita das soluções do problema. Percorre o espaço de busca usando como estrutura auxiliar uma árvore montada

Variável básica	nº da equação	Coeficientes					Constante
		z	x_1	x_2	x_3	x_4	
z	0	1	-3	-2	0	0	0
x_3	1	0	1	1	1	0	6
x_4	2	0	5	2	0	1	20

Figura 2.4: Forma tabular inicial do método simplex.

Fonte: Adaptado de Belfiore e Fávero (2013).

dinamicamente, onde o nó raiz caracteriza o problema original a ser solucionado. As folhas representam soluções viáveis inteiras ou inviáveis. Já os nós internos correspondem aos problemas de solução fracionária derivados do problema original. O algoritmo mantém a melhor solução inteira obtida até o momento e um conjunto de nós correspondentes aos problemas ainda não explorados.

O primeiro passo do algoritmo é remover as restrições de integralidade do problema original e inserir o problema resultante desta remoção no conjunto de nós. A partir deste ponto o algoritmo repete as seguintes operações até o conjunto estar vazio:

- (a) remover um problema do conjunto.
- (b) resolver o problema escolhido.
- (c) decidir entre fazer ramificação ou poda.

A decisão entre ramificação ou poda é dependente da solução encontrada para o problema. A ramificação é feita sempre que três condições são satisfeitas:

- (a) a solução encontrada é viável.
- (b) a solução encontrada contém variáveis fracionárias.
- (c) a solução encontrada tem função objetivo melhor do que a melhor solução inteira obtida até o momento. .

Neste caso, a partir do problema escolhido, denotado por p , são criados outros dois problemas derivados, denotados por $p+$ e $p-$, restringindo o valor de uma variável fracionária na solução. Suponha que uma variável x da solução tenha um valor fracionário v . Assim o problema $p+$ terá a mesma função objetivo e o mesmo conjunto de restrições de p , acrescido de uma restrição adicional da forma $x \geq \lceil v \rceil$. Já o problema $p-$ terá a mesma função objetivo e conjunto de restrições de p , acrescido de uma restrição adicional da forma $x \leq \lfloor v \rfloor$. Com isso, $p+$ e $p-$ são inseridos no conjunto de problemas e p é removido. Note que ao executar uma operação de

ramificação, o algoritmo retira um problema do conjunto, mas insere outros dois. O processo de poda é feito nos demais casos. Quando uma solução inteira é encontrada, ela é comparada com a melhor solução inteira obtida até o momento. Caso a nova solução inteira seja melhor que a armazenada, esta última é atualizada, de modo que a melhor solução sempre fique armazenada. Quando a solução encontrada para um problema for inviável, não é necessário fazer a ramificação, pois inserir mais restrições ao problema não fará com que ele se torne viável. Quando encontra-se uma solução fracionária com função objetivo pior que a melhor solução encontrada até o momento, não se executa o processo de ramificação, devido ao fato de que a inserção de restrições sempre mantém ou causa piora na função objetivo. Neste sentido, nenhuma solução encontrada nos problemas derivados irá encontrar uma melhor que a já obtida.

2. Plano de Cortes - Neste algoritmo, cortes (restrições) são inseridos iterativamente, por meio de restrições individuais no problema original. Nesta estratégia o problema inteiro é resolvido como um PPL, retirando as restrições de integralidade, e com a solução encontrada faz-se um novo problema, porém com novas restrições, forçando a remoção da solução encontrada do espaço de busca sem que nenhuma solução inteira seja removida. O novo problema é resolvido e novas restrições são inseridas, ciclicamente, excluindo parte do espaço de busca. O processo é repetido até alcançar uma solução inteira.
3. *Branch-and-cut* - esta estratégia une o *Branch-and-bound* com Plano de Cortes visando reduzir o número de problemas resolvidos pelo *branch-and-bound*. Ele incorpora cortes válidos em cada problema resolvido, objetivando reduzir o número de ramificações. Esta ação visa diminuir o tempo total de busca.

Kawamura e Ronconi (2006) estudaram em seu trabalho o PSMU com data de entrega comum e penalidades por antecipação e atraso. Nele foi proposta a resolução exata através de um algoritmo *branch-and-bound*. Normalmente, para resolução dos problemas cuja classificação é *NP-Difícil* utilizam-se heurísticas ou meta-heurísticas. A realização do estudo comprovou a eficiência do modelo proposto em instâncias de pequeno porte. Compararam a performance do algoritmo desenvolvido com softwares comerciais de otimização conseguindo melhores tempos com o seu trabalho.

No estudo realizado por Tang e Gao (2009) foi abordado o PSMU de uma linha de galvanização contínua, cujo objetivo era elaborar um sequenciamento da produção que respeitasse as restrições impostas pelas características do processo e dos produtos produzidos.

Neste problema haviam dois tipos de bobinas, A e B , que apresentavam, respectivamente, qualidade superficial elevada e regular. A primeira exigência estava vinculada ao número máximo de bobinas do tipo A produzidas sequencialmente, outra característica considerada foi a inserção de bobinas do tipo B entre ciclos de bobinas do tipo A , pois algumas destas bobinas possuíam incompatibilidade (de processamento sequencial) entre si. Neste problema havia a necessidade de sequenciar dois tipos de produtos (bobinas de qualidades diferentes) considerando os custos associados as mudanças de *jobs* e também os custos de inserção das bobinas de transição entre os elementos de sequenciamento incompatíveis. Inicialmente os autores fizeram o modelamento matemático utilizando programação inteira mas não foi possível resolver o problema pois, como ele é \mathcal{NP} -Difícil e as instâncias do problema real eram grandes, o tempo de resolução seria inviável. Os autores resolveram o problema utilizando a heurística *Tabu Search*. Foram feitas melhorias no algoritmo incluindo *Variable Depth Search (VDS)* para intensificar as buscas, e também, uma estratégia de diversificação das buscas que passou a utilizar um número não definido de movimentos de troca e inserções durante a sua execução. Os autores relatam que para todas as instâncias que o modelo exato encontrou o resultado ótimo, o seu algoritmo também o fez, e nas instâncias maiores onde o modelo exato não conseguiu o resultado ótimo, a heurística alcançou melhores resultados.

Na Seção 2.5 são descritas as heurísticas adotadas para a resolução dos problemas de sequenciamento de máquina única.

2.5 Heurísticas para a Resolução do PSMU

No trabalho de Santos (2016) são apresentados alguns dos motivos pelos quais utilizam-se heurísticas para a resolução dos problemas na área de pesquisa operacional. O principal deles é quando se precisa resolver os problemas classificados como \mathcal{NP} -Difícil. Nestes casos, soluções boas são aceitas pois encontrar a solução ótima pode ser uma tarefa muito complicada ou até impossível. As heurísticas normalmente não garantem resultados ótimos, e para o seu desenvolvimento, não são exigidas rigorosas análises formais do problema entretanto, os conhecimentos específicos sobre o problema a ser tratado são fundamentais para a elaboração da heurística (MELIÁN; PÉREZ; VEGA, 2003).

Em seu trabalho, Fuchigami (2016) destaca algumas características interessantes dos métodos heurísticos, tais como a capacidade de alcançar soluções boas (e eventualmente ótimas) sem a necessidade de percorrer todas as soluções viáveis, normalmente são de fácil

implementação, são pouco exigentes quanto ao ferramental matemático e computacional e frequentemente são rápidos, podendo ser aplicados em problemas reais. Ele também salienta algumas das situações onde a aplicação da heurística é adequada:

- Quando não se dispõe de um método exato ou quando ele demanda um tempo elevado para sua execução.
- Nas situações onde soluções razoáveis são suficientes.
- Como etapas de busca local em algoritmos mais complexos, ou nas meta-heurísticas.

As heurísticas podem ser classificadas em dois tipos: heurísticas construtivas, que requerem um conhecimento significativo do problema pois normalmente são específicas para cada caso e as heurísticas de melhoria ou busca local, que, a partir de uma solução viável, buscam outras soluções melhores que a fornecida inicialmente. Esta busca pode acontecer segundo a estratégia adotada, que pode considerar somente procurar uma melhoria ou buscar em todo o conjunto de soluções vizinhas, retornando a melhor encontrada. O primeiro tipo, normalmente, apresenta respostas mais rápidas porém, eventualmente, menos aprimoradas e no segundo tipo, garante-se o alcance da melhor solução local possível, entretanto gasta-se mais tempo para esta busca(SANTOS, 2016).

Segundo Talbi (2009), Santos (2016) as heurísticas são específicas para a resolução de determinados problemas. As meta-heurísticas podem ser aplicadas a qualquer tipo de problema de otimização, mesmo nos casos mais complexos e com grandes instâncias. Normalmente possuem desenvolvimento e implementação simples e são muito versáteis. As áreas de aplicação das meta-heurísticas são virtualmente ilimitadas, podendo ser usadas por exemplo em inteligência artificial, programação matemática e pesquisa operacional. Elas podem ser divididas em dois grupos:

- Meta-heurística com população: faz a evolução de toda a população de soluções. Assim consegue-se uma diversificação maior em todo o espaço de busca.
- Meta-heurística de solução única: nesta ocorre a manipulação de uma única solução durante a busca. Esta estratégia permite uma intensificação da busca em determinadas regiões.

São exemplos de meta-heurística com população os algoritmos evolucionários e para os casos de solução única tem-se alguns exemplos como *Simulated Annealing*, *Iterated Local Search* e *Variable Neighborhood Search* (TALBI, 2009).

Uma das meta-heurísticas de grande destaque na literatura são os algoritmos genéticos (TALBI, 2009). Eles foram desenvolvidos pelo professor John Holland, da universidade

de Michigan - EUA sendo inspirados nos processos de evolução da natureza. Segundo Junior (2006), estes algoritmos devem obedecer a algumas etapas mínimas:

1. Inicialização da população.
2. Avaliação de cada indivíduo da população.
3. Geração de novos membros da população a partir dos elementos existentes, aplicando operações de cruzamento e mutação, substituindo os indivíduos da população anterior.
4. Finalização do ciclo evolutivo quando for alcançado o critério de parada, caso contrário, continuação da evolução da população a partir da etapa 2.

Algumas definições sobre os elementos dos algoritmos genéticos devem ser apresentadas para a sua melhor compreensão. São eles:

- **População:** Trata-se da forma como as diversas soluções do problema são armazenadas e manipuladas. Normalmente um cromossomo representa uma solução completa do problema. Eles são compostos por genes e os seus valores são chamados de alelos.
- **Função objetivo:** Caracteriza-se como mecanismos de medição da *qualidade* de um cromossomo. Ela também chamada de função *fitness*. Ela está diretamente associada ao problema e às suas restrições.
- **Método de seleção:** São os critérios utilizados na heurística para identificação e seleção dos cromossomos que serão utilizados nas operações de evolução da população.
- **Método de reprodução:** As operações de evolução da população normalmente presentes nos algoritmos são cruzamento (*crossover*) e mutação. No cruzamento dois ou mais elementos da população são escolhidos e seus genes são selecionados e inseridos em um novo cromossomo, chamado *filho*. Na mutação, normalmente uma parte dos genes de um cromossomo é alterada visando introduzir modificações na população para permitir uma busca ampla do domínio do problema.
- **Substituição de elementos:** Os novos *filhos* ou *mutantes* substituem os cromossomos existentes de acordo com as regras que forem estabelecidas, podendo haver substituição parcial da população ou total.

A evolução dos elementos da população do algoritmo genético ocorre por cruzamento ou mutação. O operador de cruzamento normalmente seleciona dois elementos aleatoriamente na população, e faz a combinação dos genes destes cromossomos criando um novo membro para a população da geração seguinte. A mutação consiste, normalmente, na modificação aleatória e pontual de um cromossomo qualquer. Ela ocorre segundo alguma probabilidade e visa evitar a convergência prematura da população, ou seja, escapar de um ótimo local (JUNIOR, 2006).

Foram estudados vários trabalhos para buscar um melhor entendimento das soluções heurísticas ou meta-heurísticas aplicadas e também foram verificados os critérios de escolha dos autores pelos métodos empregados. Foi identificado em alguns trabalhos a aplicação de um framework chamado *BRKGA* - Algoritmo Genético de Chaves Aleatórias Viciadas, porém não foi encontrado qualquer trabalho de sequenciamento de linha de produção contínua que tenha utilizado esse *framework*. Devido a isso decidiu-se por utilizar nesta dissertação o *BRKGA* para identificar a sua aplicabilidade nesse tipo de problema. A seguir tem-se uma lista de trabalhos e pesquisa operacional que utilizaram essa meta-heurística.

- Desenvolvimento de uma Metaheurística BRKGA com Infecção Viral para o Problema de Atribuição de Localidades a Anéis em Redes SONET/SDH - Rebouças (2018)
- Metaheurísticas GRASP e BRKGA aplicadas ao Problema da Diversidade Máxima - Moro (2017)
- Otimização de Redes de Distribuição de Energia Elétrica por BRKGA - Cavalheiro e Filho (2017)
- Metaheurística BRKGA aplicada a um problema de programação de tarefas no ambiente *Flowshop* híbrido - Mainieri (2014)
- Metaheurística Híbrida Aplicada ao Problema de *Job Shop* - Rizzi, Chaves e Lorena (2015)
- Métodos Exatos e uma Abordagem Heurística para o Problema de Alocação de Berços - Martin et al. (2015)

A meta-heurística baseada no algoritmo de chaves aleatórias viciadas, ou do inglês *Biased Random-Key Genetic Algorithm*, também identificado pela abreviação *BRKGA*

foi derivado do algoritmo de chaves aleatórias ou *RKGA* estudado e proposto por Bean (1994). Os seus detalhes podem ser vistos na Seção 2.6.

2.6 Meta-heurística *BRKGA*- Algoritmo Genético de Chaves Aleatórias Viciadas

O *RKGA* introduzido por Bean (1994), foi aplicado em problemas de sequenciamento e otimização. Nele, diferente de muitos algoritmos genéticos tradicionais, a representação cromossômica da solução é composta de chaves numéricas aleatórias, e esta característica permite que o operador de cruzamento produza novos cromossomos válidos, enquanto as representações tradicionais apresentam um ponto de fragilidade nesta operação. O resultado do cruzamento pode ser um cromossomo inválido. Como exemplo tem-se um problema de sequenciamento de máquina única onde os cromossomos representam a ordem de execução dos pedidos e, após uma operação de cruzamento de 1 ponto, foram obtidos dois *filhos* inválidos com valores duplicados, como pode ser visto na Figura 2.5 . No *RKGA* o resultado é sempre uma sequência de números aleatórios. Para a interpretação destes valores utiliza-se uma função de decodificação. Ela é responsável por interpretar os valores aleatórios das chaves que compõem o cromossomo identificando a solução do problema. No algoritmo *RKGA* proposto, a identificação do cromossomo é feita ordenando os seus valores de forma crescente e, em seguida identificando as respectivas posições no cromossomo original. Destaca-se na Figura 2.6 um exemplo de operação de cruzamento de 1 ponto feita pelo algoritmo, mostrando os dois possíveis cromossomos de chaves aleatórias e a representação decodificada. A avaliação do cromossomo quantificando o seu resultado é feita através de uma função *fitness*, que pode ou não estar acoplada à função decoder.

No *RKGA* a seleção dos cromossomos para o cruzamento é feita aleatoriamente em toda a população e o cruzamento entre os elementos é feito baseado na estratégia de crossover uniforme parametrizado estudado por Spears e De Jong (1991), onde a cada gene,

Cromossomo A	1	2	3	4	5
Cromossomo B	5	3	4	2	1
Cruzamento de 1 ponto					
Resultado 1	1	2	4	2	1
Resultado 2	5	3	3	4	5

Figura 2.5: Cruzamento de 1 ponto - Inválido
Fonte: Autoria própria baseado em Bean (1994)

Cromossomo A	0,45	0,87	0,15	0,68	0,77	Ordem de execução: 3-1-4-5-2
Cromossomo B	0,1	0,25	0,89	0,44	0,99	Ordem de execução: 1-2-4-3-5
Cruzamento de 1 ponto						
Resultado 1	0,45	0,87	0,89	0,44	0,99	Ordem de execução: 4-1-2-3-5
Resultado 2	0,1	0,25	0,15	0,68	0,77	Ordem de execução: 1-3-2-4-5

Figura 2.6: Cruzamento de 1 ponto - Válido
 Fonte: Autoria própria baseado em Bean (1994)

uma moeda viciada é jogada, e de acordo com o resultado cada cromossomo contribui com o seu gene. Considerando que o lado *cara* da moeda seja viciado, com uma probabilidade de 70%, e esteja associado ao cromossomo A e o lado *coroa*, com probabilidade de 30%, esteja associado ao cromossomo B, a probabilidade de que um gene do cromossomo A seja passado ao filho é maior. Este mecanismo pode ser usado para priorizar genes de boas soluções e sua intensidade pode ser ajustada conforme a necessidade. Outra característica deste algoritmo está no modo como a mutação ocorre. Um cromossomo da população é substituído por um novo cromossomo de valores aleatórios. Esta estratégia permite sair de ótimos locais. A geração seguinte recebe os melhores cromossomos da geração atual, que são chamados de cromossomos elite.(BEAN, 1994; GONÇALVES; RESENDE, 2011).

Eventualmente, em problemas com um conjunto maior de restrições em que o decodificador é mais complexo, faz-se necessário ajustar os cromossomos gerados no *RKGA* para que não violem as restrições do problema em estudo. Pode se verificar na Figura 2.7 um exemplo de problema da mochila onde o cromossomo, após decodificado, produziu uma solução que violou a restrição de capacidade máxima da mochila. No exemplo foi gerada uma chave aleatória com os valores $[0, 82; 0, 12; 0, 54; 0, 89; 0, 26]$ e foram selecionados todos os genes cujos alelos são $> 0, 5$. Isso resultou na seleção dos objetos $[1; 3; 4]$, cuja soma dos seus pesos ultrapassa a capacidade da mochila violando assim um conjunto de restrições. Neste caso, pode-se descartar o cromossomo ou ajustá-lo para atender a todas as restrições.(CHAGAS, 2017)

Em seu trabalho, Bean (1994) aplicou o *RKGA* em problemas de Alocação de recursos, problema do caixeiro viajante e problemas de associação quadráticas, tendo sucesso em quase todas as instâncias e problemas testados. Os autores relatam que nos problemas de associação quadrática, onde se busca associar m agentes a m localidades minimizando uma função objetivo quadrática, não conseguiram bons resultados, quando comparados com outras heurísticas citadas em seu trabalho. Em suas conclusões afirmam que o algoritmo apresentado foi muito eficiente alcançando valores ótimos em muitos casos.

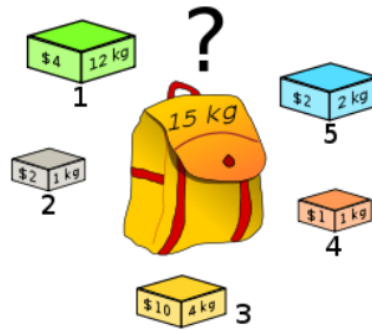


Figura 2.7: Problema da mochila com violação da restrição.
Fonte: Adaptado de Chagas (2017)

Gonçalves e Resende (2011) apresentaram um *framework* para resolução de problemas de otimização combinatória que chamaram de *Biased Random-Key Genetic Algorithms* - *BRKGA*. Como principal característica desta ferramenta está a capacidade de separação entre a arquitetura do *framework* e as características do problema em estudo. Esta separação permite a utilização do *BRKGA* para qualquer problema, restando apenas ao pesquisador o trabalho de elaborar a função de decodificação dos cromossomos.

No *BRKGA* a seleção dos cromossomos que serão cruzados difere do *RKGA* pois o cruzamento sempre acontece com um cromossomo do grupo *elite* e com outro do restante da população. Em algumas implementações o segundo cromossomo pode ser selecionado do grupo Elite. O cruzamento ocorre da mesma forma como no algoritmo proposto por Bean (1994), isto é, utilizando a estratégia de crossover uniforme parametrizado (SPEARS; De Jong, 1991). Para que esta característica esteja presente no algoritmo é necessário que $p_e > 0,5$. Pode se verificar na Figura 2.8 uma operação de cruzamento deste *framework*, onde observa-se que em todos os casos onde o número randômico sorteado foi inferior ao limite $p_e = 0,7$ o cromossomo *elite* forneceu o seu gene.

A evolução das populações ocorre migrando os n melhores cromossomos, chamados de *elites*, de uma geração para a outra, gerando uma parcela de mutantes e cruzando o restante da população. Na Figura 2.9 pode-se observar o processo de evolução das gerações da população. Vale destacar que a mutação é simplesmente a inclusão de novos elementos gerados aleatoriamente, da mesma forma como a população inicial foi gerada.

A sequência das etapas descritas para este *framework* podem ser visualizadas na Figura 2.10. Nela também está destacada a única etapa dependente do problema, que é decodificação de cada um dos cromossomos e a sua avaliação.

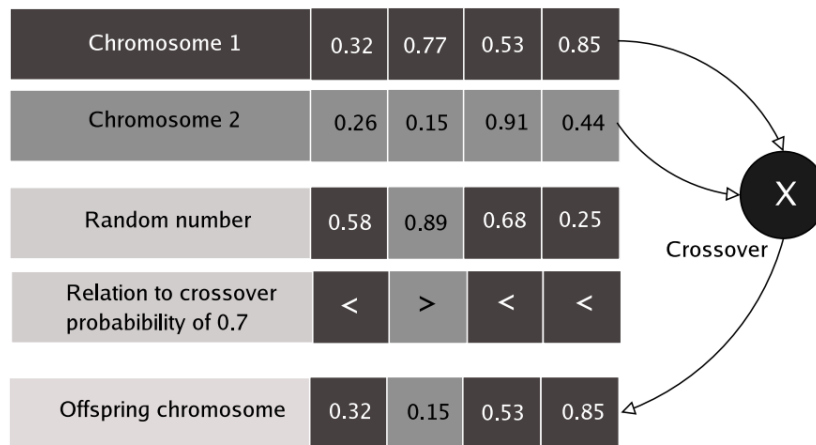


Figura 2.8: Cruzamento no *BRKGA*
 Fonte: Adaptado de Gonçalves e Resende (2011)

No trabalho desenvolvido por Mainieri (2014) foi estudado o *flowshop* híbrido, utilizando uma solução exata em programação linear inteira mista e algumas variações do *BRKGA*. Todas as variações propostas foram implementadas na função de decodificação mantendo o *framework* original. O autor testou algumas heurísticas construtivas, tais como ordem inversa e direta de programação das tarefas, identificação do estágio gargalo e algumas outras. Estas variações foram possíveis pois, conforme apresentado na Figura 2.10, somente o *decoder* é dependente do problema. Neste trabalho foi verificado que o *BRKGA* encontrou o ótimo global em 61% dos casos avaliados.

Moro (2017) estudou a aplicação de duas meta-heurísticas, *Greedy Randomized Adaptive Search Procedure (GRASP)* e *Biased Random-Key Genetic Algorithm (BRKGA)* em um problema de diversidade máxima. O problema, classificado como \mathcal{NP} -Difícil, consiste na seleção de um subconjunto de elementos que sejam os mais diversos possíveis dentro de um conjunto. Ele utilizou o *path relinking* como estratégia de busca local para o *GRASP*. O autor destacou que o *GRASP* obteve melhores resultados em instâncias grandes e o *BRKGA* alcançou melhores resultados em instâncias pequenas e médias.

No trabalho de Gonçalves e Resende (2011) foram apresentados alguns problemas onde o *BRKGA* foi aplicado com sucesso. O primeiro trabalho citado foi de formação de células de fabricação onde o *BRKGA* apresentou soluções melhores, em média, de 2,88%. No problema de empacotamento ortogonal foram comparados os algoritmos *SGA* e *SAGA* com o *BRKGA*, e este último apresentou reduções na função objetivo de 0,24% e 0,36% comparados com os anteriores. Para o problema de sequenciamento *job shop* o *BRKGA* foi comparado com algoritmos genéticos, *simulated annealing*, e busca tabu alcançando o ótimo local em 72% das 43 instâncias testadas. Foi também avaliado um PSMU com pe-

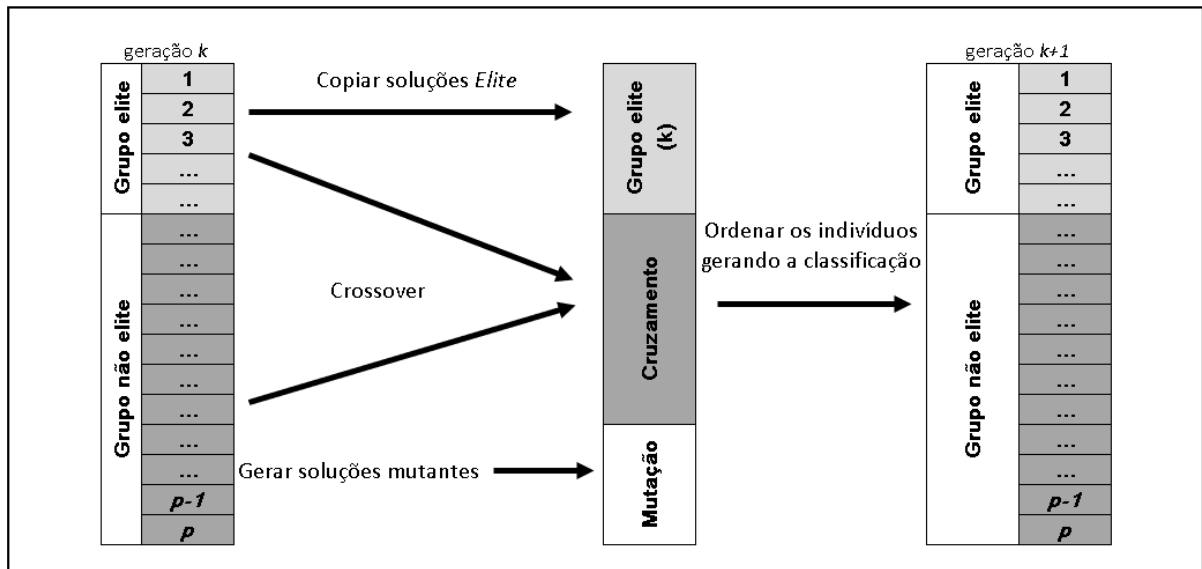


Figura 2.9: Evolução da População no *BRKGA*
 Fonte: Adaptado de Mainieri (2014)

nalidade linear para antecipação e quadrática para atrasos. Foram utilizados três métodos de busca local para aprimoramento dos cromossomos, *API*, *3SW* e *LCI*, permitindo que o *BRKGA* alcançasse o ótimo global em mais de 90% das instâncias testadas, demonstrando a sua superioridade quando comparado com os demais métodos utilizados.

Prasetyo, Qoyyimah e Fauza (2018) analisaram o problema de roteamento de veículos fechados capacitados com janelas de tempo (*Capacitated Closed Vehicle Routing Problem with Time Windows - CCVRPTW*). Os autores afirmaram que o *BKRGGA* apresentou melhores resultados que a heurística *MTVRP* proposta anteriormente, além disto, testaram uma variação do *framework* incluindo o cruzamento com mais de dois cromossomos ou *Multiple Parent*. Os autores apresentaram na Figura 2.11 uma comparação entre o *framework* original, a versão modificada utilizando três cromossomos para o cruzamento (um elite + dois da população, podendo ser outro elite) e a última versão com somente um elite e dois não elites no cruzamento. A implementação que utilizou somente um *elite* apresentou uma convergência mais rápida que as demais, mostrando que a modificação trouxe benefícios ao *BRKGA*, tornando a versão com múltiplos cromossomos 7,4% melhor que o *BRKGA* padrão.

Rizzi, Chaves e Lorena (2015) aplicaram a meta-heurística *BRKGA* híbrido em um problema de *job shop*. Foi adicionado ao algoritmo original uma estratégia de *Clustering Search* que mapeia as regiões com maior número de soluções concentradas e após esta avaliação efetua-se uma busca local no centro desta região para encontrar possíveis solu-

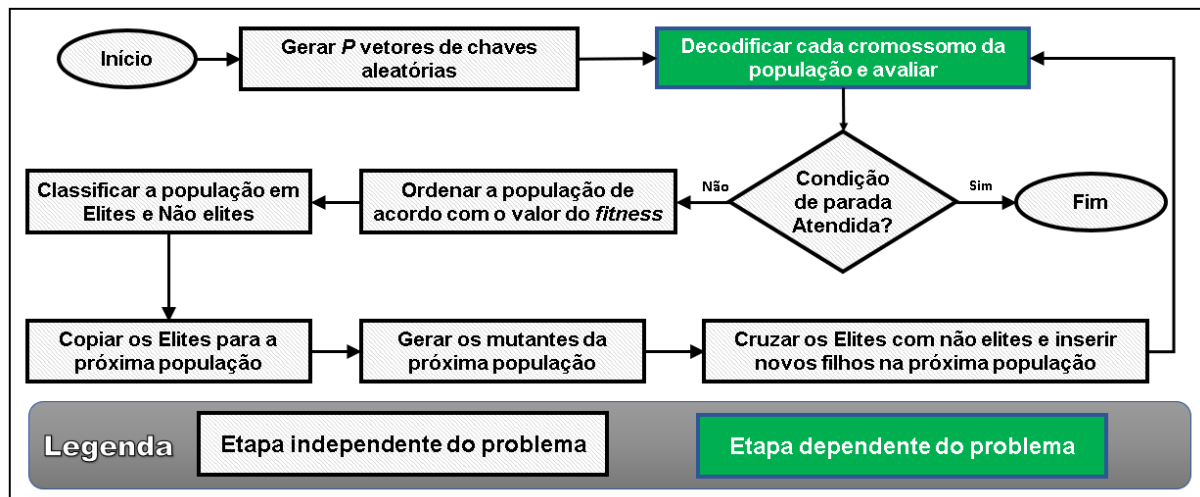


Figura 2.10: *Framework* do *BRKGA*
 Fonte: Adaptado de Resende (2012) e Chagas (2017)

ções melhores neste *cluster*. Caso encontre, o centro do *cluster* é atualizado com o melhor cromossomo e o processo de avaliação do *cluster* é repetido. O autor destaca que o algoritmo híbrido proposto alcançou solução ótima em aproximadamente 58% das instâncias testadas e a diferença média em relação aos melhores resultados foi de 1,12%. Concluíram que o método implementado é eficiente e trouxe resultados em tempos baixos.

Rebouças (2018) aplicou o *BRKGA* para resolução do Problema de Atribuição de Localidades a Anéis SONET (*Synchronous Optical Network*) (PALAS). Neste problema existe a necessidade de dividir as localidades em anéis que satisfaçam as exigências e apresentem o menor custo possível. O autor relata que se trata de um problema classificado como *NP-Difícil* e por isso faz-se necessário o uso de meta-heurísticas. Os anéis locais devem atender às demandas especificadas e se comunicarem com um anel principal, chamado anel federal. Através dele que os anéis locais podem comunicar entre si. Também são descritos no trabalho os métodos normalmente utilizados para resolução do problema, que são os algoritmos genéticos, *simulated annealing*, *GRASP*, *Variable Neighbourhood Search*, colônia de formigas e busca tabu. Durante as pesquisas foi encontrada uma aplicação do *BRKGA* para o PALAS, entretanto, neste trabalho foi aplicado um método de mutação na população elite visando uma melhoria dos melhores cromossomos encontrados. A nova mutação foi chamada pelo autor de infecção viral e durante o seu processo os cromossomos infectados são avaliados e os melhores são inseridos na geração seguinte da população. Os autores apresentaram uma tabela avaliando a eficiência do método proposto, encontrando uma melhora considerável na assertividade do algoritmo.

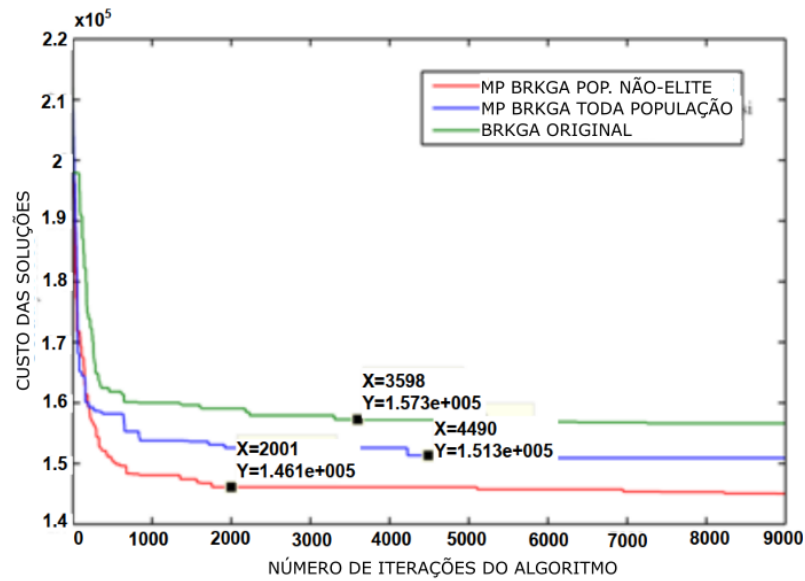


Figura 2.11: Avaliação do algoritmo *BRKGA*.
 Fonte: Adaptado de Prasetyo, Qoyyimah e Fauza (2018)

Em três instâncias C1, C2 e C3 testadas o *BRKGA* original alcançou respectivamente, 92,5%, 61,3% e 50% das soluções ótimas conhecidas e a versão com infecção viral alcançou 99,2%, 97,8% e 100% dos valores ótimos.

Os autores Martínez et al. (2011) aplicaram o *BRKGA* no *Capacitated Arc Routing Problem (CARP)*. No trabalho adotaram a chave randômica com números inteiros entre dois limites $[MinU, MaxU]$. O decodificador classifica os genes dos cromossomos em ordem crescente e os envia para uma heurística *Iterated Tour Partitioning* que faz a divisão das rotas como soluções que respeitam as restrições do problema (soluções viáveis). Após esta etapa é aplicada uma busca local no cromossomo e em seguida, faz-se o ajuste das chaves no cromossomo para que representem todas as modificações realizadas nas melhorias e, por último, o cromossomo é devolvido à população. Os autores adotaram uma estratégia de mutação que altera alguns alelos aleatoriamente. Outra modificação implantada foi o reinício da população mantendo apenas os cromossomos do conjunto *elite*. Como resultados, os autores obtiveram valores ótimos ou muito próximos em 21 das 25 instâncias testadas, um custo médio de viagens 10,7% melhor que os melhores resultados da literatura, o tempo gasto para encontrar a melhor solução foi menor que 60% do limite definido em 22 instâncias.

Neste trabalho foi aplicado o *BRKGA* em sua modelagem original para a resolução do Problema de Sequenciamento de pedidos em linha de galvanização contínua. Os parâmetros do algoritmo foram ajustados segundo Gonçalves e Resende (2011). Foi implementada

uma heurística construtiva para a inicialização da população baseada na ideia apresentada no trabalho de Santos (2016), entretanto, a função elaborada não inicializa toda a população, gera somente um único cromossomo válido e substitui o primeiro elemento da primeira população.

Na Seção 2.7 estão resumidos alguns dos trabalhos relacionados ao tema de sequenciamento de linhas de produção contínuas de aço e produtos correlatos encontrados durante o levantamento bibliográfico desta dissertação.

2.7 Trabalhos Relacionados

Nesta seção são apresentados alguns dos trabalhos mais recentes encontrados durante a pesquisa para esta dissertação, que abordam o tema sequenciamento de produção e que estejam relacionados à fabricação de aço e seus derivados.

Os autores Tan et al. (2017) estudaram em seu trabalho a otimização do sequenciamento da produção de um laminador de tiras a quente visando a redução do consumo de energia elétrica principalmente nos horários de pico. Neste trabalho foram consideradas tarifas de energia distintas e a divisão do dia em quatro horários. A principal abordagem do método proposto foi a equalização do planejamento de produção com a diminuição da demanda energética nos horários de maior custo por Kwh. O algoritmo considerava, além da redução de consumo energético, a minimização das penalidades associadas às mudanças entre as placas sequenciadas. Os autores relatam que o algoritmo proposto gerou resultados melhores que o método de deslocamento de demanda com produção fixa.

O trabalho apresentado pelos autores GAO e QU (2018) propõe o desenvolvimento de uma modelagem para resolução do problema de sequenciamento de linhas de galvanização contínua paralelas utilizando uma combinação entre programação linear inteira mista e programação por restrições (MILP/CP). O objetivo principal é a redução dos custos de produção das linhas. Uma característica destacada pelos autores é que neste problema as regras de produção são complexas. Devido a grande complexidade do problema e o elevado tempo de processamento, os autores propuseram o uso de uma heurística que visa identificar as inviabilidades dos sequenciamentos gerados e com isso gerar cortes (restrições) de maneira mais eficiente. Eles utilizaram 60 instâncias criadas aleatoriamente para representar os dados de produção. Os resultados encontrados pelos autores comprovam que o método híbrido proposto resolveu todas as instâncias mais rapidamente que o método tradicional, principalmente nas instâncias de grande porte.

O trabalho apresentado pelos autores Cohen, Foxx e Alul (2018) utilizou algoritmo genético para construção de sequências de produção para uma linha de galvanização contínua cujos objetivos principais contemplavam a minimização das transições entre bobinas sequenciadas. Neste problema há a inserção de bobinas de transição quando não é alcançada a compatibilidade entre dois itens sequenciados. São consideradas quatro características na função *fitness*: espessura, largura, especificação do aço e revestimento. Todas as características foram normalizadas entre valores mínimos e máximos e a cada uma delas foi associado um peso na função objetivo. Os autores relatam que houve redução das mudanças das características entre as bobinas sequenciadas. Outro ponto destacado é a capacidade da alteração dos pesos associados às características consideradas facilitando a priorização de alguma delas em detrimento às demais.

Foram estudadas algumas técnicas aplicadas ao sequenciamento da produção no *survey* elaborado pelos autores Lei e Cai (2020). Neste trabalho foram pesquisados os usos das meta-heurísticas com multi-populações colônia artificial de abelhas (ABC), algoritmo competitivo imperialista (ICA) e "sapo pulando embaralhado" (SFLA) nos problemas de sequenciamento de máquina única, sequenciamento de máquinas paralelas, *job shop* e *job shop* flexível. São apresentadas algumas características das soluções implementadas nos trabalhos estudados pelos autores, tais como o tipo de função objetivo, se apresentam ou não restrições especiais como definição de lotes ou estoques intermediários limitados. Os autores destacam em suas conclusões os trabalhos que foram aplicados visando sequenciamento distribuído. Destacam também os trabalhos que trouxeram novos mecanismos de otimização dos algoritmos citados (ABC, ICA e SFLA) e além disso, apresentam os trabalhos que foram aplicados em problemas de sequenciamento de empresas sustentáveis como por exemplo sequenciamentos de produção considerando o consumo energético em horários de pico.

No trabalho dos autores Dewa e Nleya (2020) são estudados algoritmos para resolução do problema de sequenciamento de máquinas paralelas de uma planta de galvanização em batelada visando a minimização do *makespan*. São avaliados quatro algoritmos de sequenciamento neste trabalho, sendo eles a programação linear inteira, *SPT* ou menor tempo de processamento, *LPT* ou maior tempo de processamento e por último um algoritmo genético guloso. Foi utilizado o *Matlab* neste estudo e os autores constataram experimentalmente que o algoritmo genético guloso apresentou o melhor resultado entre eles. Na instância testada, com 50 *jobs*, o pior *makespan* encontrado foi utilizando o *SPT*, com 192 minutos e o melhor foi obtido pelo algoritmo genético, com 176 minutos.

Os autores Armellini et al. (2020) propoem um modelo geral para o problema de sequenciamento da produção de máquinas de corrida contínua utilizadas na fabricação de aço, bem como nos processos de fabricação de lingotes. Para resolver o problema foi utilizada uma heurística baseada no recozimento simulado (SA). Foi também elaborado um procedimento construtivo que atribui aos *jobs* o seu tempo de processamento e os recursos necessários. A busca por soluções melhores na vizinhança foi baseada em mudanças de ordem e mudanças de grupos de *jobs*. Os autores relatam que a técnica desenvolvida foi capaz de encontrar boas soluções para uma ampla gama de configurações de máquinas e horizontes de produção diferentes. Também destacaram que o algoritmo superou os resultados do método baseado em restrições em quase todas as instâncias testadas.

As principais características dos trabalhos citados foram resumidas no Quadro 2.1, onde a primeira coluna contém a referencia do trabalho, a segunda caracteriza o tipo de processo estudado, a terceira e quarta colunas apresentam os métodos aplicados pelos autores na resolução de suas pesquisas.

Quadro 2.1: Resumo dos Trabalhos

Autores	Processo	Método	Método
		Exato	Aproximado
Tan et al. (2017)	Batelada	-	A.G.
GAO e QU (2018)	Contínuo	PLIM-CP	-
Cohen, Foxx e Alul (2018)	Contínuo	-	A.G.
Lei e Cai (2020)	Ambos	-	ABC, ICA, SFLA
Dewa e Nleya (2020)	Batelada	PLIM	A.G.
Armellini et al. (2020)	Contínuo	-	S.A.
Essa pesquisa	Contínuo	PLIM	A.G.

Legenda

- A.G. : Algoritmo Genético.
- ABC : Colônia Artificial de Abelhas
- ICA : Algoritmo Competitivo Imperialista.
- PLIM : Programação Linear Inteira Mista.
- PLIM-CP : Programação Linear Inteira Mista + Programação por Restrições.
- SFLA : Sapo Pulando no Pântano.
- S.A. : Recozimento Simulado.

Comparando esta pesquisa com os principais trabalhos apresentados no Quadro 2.1 podem ser destacadas as seguintes diferenças:

- Tan et al. (2017) : Sequenciamento de um laminador de tiras a quente visando redução do gasto com energia nos horários de maior demanda (maior custo associado)

e a minimização das mudanças de largura.

- GAO e QU (2018) : Sequenciamento de linhas paralelas de galvanização contínua com dois tipos de bobinas (maior e menor qualidade superficial) buscando a minimização do custo de produção que é calculado considerando as mudanças de largura, a inclusão de bobinas intermediárias, a existência de atrasos na produção, entre outros fatores. Há restrições limitando a quantidade de bobinas de maior qualidade em uma mesma sequência, também considera a ocupação de todas as linhas de produção disponíveis. Há outro conjunto de restrições associados ao tipo de bobina e a ordem de produção delas, como por exemplo, no grupo de bobinas de maior qualidade não são permitidas sequências cuja próxima bobina possua largura maior que a atual.
- Cohen, Foxx e Alul (2018) : Sequenciamento de uma linha de galvanização contínua utilizando algoritmo genético considerando na função objetivo quatro características das bobinas, a largura, espessura, composição do aço e espessura do revestimento. Associados a estas características foram definidas penalidades para permitir ajustes da "importância" de cada uma delas. Não foi abordado nessa pesquisa a minimização de atrasos ou antecipações.
- Lei e Cai (2020) : Este *survey* aborda algumas técnicas de sequenciamento baseadas em meta-heurísticas com multi-população. São analisados diversos trabalhos e suas técnicas de resolução, bem como as restrições aplicadas em cada um deles.
- Dewa e Nleya (2020) : Sequenciamento de linhas paralelas de galvanização em tanques buscando a minimização do *makespan*. Nesse trabalho os autores aplicaram PLIM, aplicaram o sequenciamento ordenando pelo maior tempo de processamento (LPT), também sequenciaram pelo menor tempo de processamento (SPT) e por último utilizaram um algoritmo genético guloso para fazer o sequenciamento. Não são consideradas quaisquer características dos pedidos além da data de entrega.
- Armellini et al. (2020) : Sequenciamento da etapa de produção do aço iniciando no forno elétrico até a máquina de corrida contínua utilizando a meta-heurística reconhecimento simulado (S.A.). Nessa pesquisa são consideradas as restrições associadas a cada uma das etapas do processo de fabricação e os tempos de transporte entre os equipamentos envolvidos. A vida útil dos componentes foi avaliada nessa modelagem pois algumas corridas especiais dependem de componentes no início de sua vida útil. Os autores analisam para a construção do sequenciamento o momento de

início da produção, os recursos necessários para a produção pedido a pedido. Além das restrições associadas aos pedidos também são consideradas restrições operacionais dos equipamentos de toda a cadeia produtiva, como por exemplo vida útil, velocidades máxima e mínima, dimensional de cada etapa entre outros fatores.

Comparando as necessidades desta pesquisa com os trabalhos apresentados no Quadro 2.1 percebe-se que os autores GAO e QU (2018) e Cohen, Foxx e Alul (2018) se aproximam parcialmente das necessidades identificadas. O primeiro trabalho utilizou PLIM-CP e considerou a minimização das mudanças de largura nos sequenciamentos e o segundo trabalho utilizou A.G. buscando a minimização das mudanças de largura, espessura, composição do aço e revestimento, entretanto não considerou a data de entrega dos pedidos. Como observado, não há na literatura, segundo o conhecimento dos autores, nenhum trabalho que atenda todas as necessidades dessa pesquisa e por isso o presente estudo se mostra relevante para o meio acadêmico. Outra colaboração desse trabalho consiste na aplicação da meta-heurística *BRKGA* na resolução do problema de sequenciamento da linha de galvanização contínua visto que durante as pesquisas também não foram encontrados trabalhos que aplicaram essa meta-heurística em problemas semelhantes. Nesse trabalho foram consideradas as características dimensionais, térmicas, uso do tratamento químico, uso do laminador de encruamento e também buscou-se a minimização das antecipações e atrasos dos pedidos sequenciados.

No Capítulo 3 apresenta-se a metodologia utilizada para a resolução dos modelos desenvolvidos, bem como a avaliação da performance desses modelos e a influência das dimensões da instância no tempo de resolução.

Capítulo 3

Metodologia

Neste capítulo são descritas as implementações das modelagens utilizadas para resolução do problema apresentado. Constam na Seção 3.1 os requisitos do sequenciamento da linha de galvanização contínua, bem como os problemas que um sequenciamento inadequado pode provocar. Ainda nesta seção apresenta-se a notação adotada neste trabalho. Na Seção 3.1.1 é desenvolvida a modelagem matemática com o foco na minimização dos atrasos e antecipações. Na Seção 3.1.2 são realizadas algumas modificações do modelo matemático para contemplar as características dimensionais. Na Seção 3.1.3 são expostas as considerações a respeito do ciclo térmico, laminador de encruamento e do TQ. Nesta seção apresenta-se a modelagem matemática completa e como ocorreu a implementação do algoritmo exato. Apresentam-se na Seção 3.2 as etapas seguidas para a construção de uma heurística para a resolução deste problema e por fim, na Seção 3.3 são apresentadas as características do banco de dados e instâncias utilizadas neste trabalho.

3.1 Formulação para o Problema de Sequenciamento da Linha de Galvanização Contínua

Esta seção aborda o problema de sequenciamento de máquina única utilizado em uma linha de galvanização contínua, aplicando a programação linear mista como ferramenta de resolução do problema apresentado.

O problema em estudo caracteriza-se pela necessidade de sequenciar diversos pedidos distintos em uma linha de galvanização contínua visando a entrega na data solicitada, na especificação metalúrgica exigida e com a qualidade contratada pelo cliente. O sequenciamento não pode ser feito de qualquer forma, pois muitas características dos pedidos e da

linha de produção devem ser consideradas para a minimização do risco de aparecimento de defeitos de qualidade nos pedidos. Na Seção 2.2 foram apresentadas algumas destas características e os possíveis defeitos.

Antes da apresentação do modelo matemático faz-se necessário recordar os requisitos do problema e também apresentar a associação das variáveis utilizadas com as características do problema estudado. Trata-se de um PSMU estático, isto é, todos os pedidos estão disponíveis no tempo $t = 0$ e qualquer um deles pode iniciar a sequência (BUSTAMANTE; CARVALHO, 2006).

Possui um conjunto de P pedidos ou *jobs* com n elementos ($1..n$) a serem processados. Cada um dos pedidos possui dia desejo d_j e tempo de processamento p_j definidos, ambos em dias. Com estes dados, quando efetua-se o cálculo da antecipação ou atraso encontra-se um valor contínuo. Cada unidade deste número representa um dia. Por exemplo, se a antecipação for calculada com o resultado 2,5 significa dizer que o pedido será entregue dois dias e meio antes do desejado.

Outra característica do processo em estudo é não permitir a preempção, ou seja, não é permitida a interrupção do processamento de um pedido para a troca por outro. Também não há indisponibilidade, uma vez iniciado o processamento de um pedido, ele permanece até a sua conclusão. No caso prático existe a indisponibilidade, mas é planejada com antecedência considerável, de modo que não afete a entrega dos pedidos e a sua produção, e quanto a preempção, ela é evitada ao máximo pois sempre que ocorre uma interrupção no processamento de um pedido aumenta-se a chance de aparecimento de alguns dos problemas descritos a seguir.

- **Perda da sequência de produção:** Alguns materiais são agrupados, mesmo sendo destinados a clientes distintos, para serem produzidos sequencialmente, como é o caso dos pedidos com tratamento químico. Em caso de interrupção desta sequência, pode acontecer de o restante do material não processado ser deslocado para o estoque e ficar aguardando a próxima oportunidade de formação de novo agrupamento compatível. Isto implica em atrasos na entrega dos pedidos.
- **Material incompatível com os demais pedidos:** As sequências de produção são elaboradas para garantir o atendimento dos critérios de qualidade, das regras de negócio e dos limites técnicos do equipamento, e uma interrupção com inclusão de pedidos diferentes pode implicar no não atendimento destes critérios. Eventualmente, quando surge uma demanda especial de atendimento emergencial de algum

cliente os engenheiros de processo e a equipe de planejamento fazem uma análise da sequência que está em execução para identificar o ponto onde esta solicitação especial pode ser inserida com o menor prejuízo ao sequenciamento previamente estabelecido. Como observado, trata-se de uma exceção que não é tratada no modelo proposto.

Por se tratar de uma linha de produção contínua não são permitidos tempos ociosos entre os pedidos. Toda a fila F de execução é preenchida com os pedidos aguardando processamento na ordem em que serão executados. O tempo ocioso, na linha em estudo, significa a parada da produção e todas as vezes que ocorre uma interrupção da operação normal da linha tem-se a geração de materiais que são classificados como qualidade inferior ou até descartados. Na fase de retorno do processamento (após paradas inesperadas ou interrupção planejada) também ocorre o desvio de material, que cessa quando o equipamento alcança a estabilidade operacional. Por estes motivos a inclusão dos tempos ociosos foi desconsiderada na modelagem deste trabalho.

Outra característica das linhas contínuas é que ao final da produção de cada um dos pedidos ocorre uma união com o início do pedido seguinte, processo que é chamado de soldagem dos materiais. Há uma pequena preparação para ocorrer esta união, mas é feita durante a execução do pedido anterior à solda, e com isto não há tempo de preparação (ou *setup*). Outro fato relevante é que a linha possui capacidade para a execução de apenas um pedido por vez, e como citado anteriormente, ele é concluído para que um novo pedido tenha o seu processamento iniciado.

Foram consideradas penalidades por antecipação ou atraso distintas entre si, que no modelo são representadas por α e β . Elas também são independentes dos *jobs*, ou seja, não existe uma matriz ou vetor para as penalidades, somente duas constantes. Segue a notação adotada no trabalho.

P	:	Conjunto dos pedidos aguardando sequenciamento na fila de execução.
n	:	Número total de pedidos aguardando sequenciamento.
F	:	Fila contendo os pedidos ordenados a serem executados.
s_f	:	Comprimento da fila F .
i e j	:	números de pedidos aguardando sequenciamento.
p_i	:	Tempo de processamento do pedido i .
d_i	:	Data compromisso ou desejo do pedido i .
$ciclo_i$:	Número do ciclo térmico do pedido i .
$Peso_Min$:	Peso mínimo para processamento de uma sequência de materiais com TQ.
$peso_pedido_i$:	Peso do pedido i .
l	:	Posição da fila de execução dos pedidos.
α	:	Fator de penalização por antecipação.
β	:	Fator de penalização por atraso.
E_l	:	Valor de antecipação do pedido que está na posição l da fila de execução.
E_{max}	:	Limite máximo para as antecipações.
T_l	:	Valor de atraso do pedido que está na posição l da fila de execução.
T_{max}	:	Limite máximo para os atrasos.
S_l	:	Momento de início da execução do pedido que está na posição l da fila de produção.
Z_{il}	:	Variável de decisão binária, que em 1, indica a ocupação da posição l da fila de execução pelo <i>job</i> i . Caso contrário permanecerá zerada.
W_{ijl}	:	Variável de decisão binária, que em 1, indica que o pedido i está ocupando a posição l e o pedido j está ocupando a posição $l + 1$ na fila de execução. Caso contrário conterá o valor 0.

No problema em estudo são necessárias outras considerações, além da data de conclusão, para a elaboração da sequência de produção. Nas Seções 3.1.1, 3.1.2 e 3.1.3 são apresentadas todas as características que foram consideradas e os motivos que justificaram a sua consideração.

3.1.1 Formulação para a Minimização dos Atrasos e das Antecipações

De acordo com a definição da Equação (2.5), que calcula o atraso e a antecipação, foi elaborada a formulação matemática do PSMU considerando as datas de conclusão e desejo de cada um dos pedidos ou *jobs*. Utilizou-se a formulação proposta por Bustamante e Carvalho (2006) com algumas modificações. A modelagem matemática desenvolvida para resolver o Problema de Sequenciamento de Máquina Única com Antecipações e Atrasos (PSMUAA) pode ser vista a partir da Expressão (3.1).

$$\text{Min } Z = \sum_{l \in F} \alpha E_l + \sum_{l \in F} \beta T_l \quad (3.1)$$

S.a.

$$\sum_{i \in P} Z_{il} = 1, \forall l \in F, \quad (3.2)$$

$$\sum_{l \in F} Z_{il} \leq 1, \forall i \in P \quad (3.3)$$

$$S_l + \sum_{i \in P} (Z_{il}(p_i - d_i)) + E_l - T_l = 0, \forall l \in F \quad (3.4)$$

$$S_{l+1} - S_l - \sum_{i \in P} p_i Z_{il} = 0, \forall l < s_f \quad (3.5)$$

$$W_{ijl} - Z_{il} - Z_{j(l+1)} \geq -1, \forall i \neq j, \forall i, j \in P, \forall l < s_f \quad (3.6)$$

$$\sum_{i, j \in P, i \neq j} \sum_{l \in F, l < s_f} W_{ijl} = s_f - 1 \quad (3.7)$$

$$Z_{il} \in \{0, 1\}, \forall i \in P, \forall l \in F \quad (3.8)$$

$$W_{ijl} \in \{0, 1\}, \forall i, j \in P, \forall l \in F \quad (3.9)$$

$$S_l \geq 0, \forall l \in F \quad (3.10)$$

$$0 \leq E_l \leq E_{max}, \forall l \in F \quad (3.11)$$

$$0 \leq T_l \leq T_{max}, \forall l \in F \quad (3.12)$$

A Equação (3.1) representa a função objetivo do problema, que visa minimizar o tempo total de antecipações e atrasos considerando as suas respectivas penalidades α e β . Aqui tem-se uma diferença entre o que foi implementado neste trabalho, que considera penalidades independentes dos pedidos, e o que foi proposto por Bustamante e Carvalho (2006), onde considerava-se penalidades dependentes dos pedidos. Contudo, estas penalidades dependentes são facilmente incorporáveis ao modelo.

Os conjuntos de restrições (3.2) e (3.3) garantem respectivamente, a ocupação de todas as posições da fila de execução F preenchendo cada uma das l posições com somente um pedido i e, também garantem que o pedido i será sequenciado no máximo uma vez em toda a fila F .

As restrições (3.4) calculam o valor de antecipação E_l ou atraso T_l do pedido que está ocupando a posição l da fila de execução. O cálculo é feito considerando o início S_l definido pelo modelo, a dia desejo d_i e o tempo de processamento p_i .

Nas restrições (3.5) tem-se a garantia de não haver tempo ocioso entre a conclusão de um pedido na posição l da fila e o início do próximo na posição $l + 1$, visto que se trata de um equipamento de produção contínua. Esta é uma modificação em relação ao modelo proposto por Bustamante e Carvalho (2006), que permitia a existência de ociosidade entre os pedidos para redução das penalidades por antecipação.

No conjunto de restrições (3.6) e (3.7) ocorre a identificação do pedido i que está ocupando a posição l da fila e do pedido j que está na sequência (posição $l + 1$ da fila). Estes conjuntos de restrições garantem a ordenação dos pedidos na fila F . Por fim, delimitam-se os valores válidos para as variáveis de decisão através das restrições (3.8), (3.9), (3.10), (3.11) e (3.12). Vale destacar que em (3.11) e (3.12) ocorre a limitação da antecipação e do atraso.

O modelo matemático proposto considera apenas as antecipações e atrasos, e no cenário real são necessárias outras considerações, que estão associadas as características da linha de produção em estudo. Na Seção 3.1.2, estas características são consideradas no modelamento e apresentam-se os motivos de sua inclusão no modelo inicial.

3.1.2 Formulação para o Tratamento das Características Dimensionais

Na Seção 3.1.1 foi apresentado o modelo clássico para solução do PSMUAA considerando apenas antecipações e atrasos. Nesta seção são incluídas novas restrições visando aproximar o modelo matemático do cenário real. As primeiras modificações foram feitas para que o modelo considere as dimensões dos materiais sequenciados. Esta consideração se deve ao fato de que uma mudança brusca destas dimensões pode provocar o aparecimento de defeitos de qualidade, como por exemplo rugas, ondulações, repuxados, recozimento insuficiente do aço que provoca dureza fora da especificação. Além dos defeitos no material, uma sequência não otimizada provoca alteração no consumo dos gases

durante o processamento dos materiais, sobrecarga nos sistemas mecânicos, entre outros problemas.

Para fazer esta alteração foi necessário disponibilizar ao modelo os dados de largura e espessura de cada um dos pedidos. Inicialmente, esta característica havia sido tratada como uma restrição, ou seja, não poderia haver mudança fora de um limite estabelecido. Porém estabelecer um limite rígido (uma constante máxima) para as diferenças mostrou-se ineficiente devido à variabilidade dos dados. Isto ocorre porque, em alguns casos, os clientes fazem pedidos incompatíveis entre si, cabendo a indústria metalúrgica tomar as devidas providências para resolver o problema. Na prática, opta-se por processar materiais incompatíveis sequencialmente, mesmo arriscando perda de qualidade em parte da bobina. Neste sentido, o modelo foi alterado para que estas divergências fossem inseridas como penalidade na função objetivo da seguinte forma: quanto maior a diferença de largura entre dois pedidos consecutivos maior a penalidade aplicada na função objetivo, o mesmo procedimento foi adotado para a espessura.

Para armazenar estas penalidades associadas às diferenças de espessuras e larguras foram criadas duas matrizes quadradas $n \times n$, sendo n o número total de pedidos aguardando sequenciamento. Estas matrizes, chamadas de MTZ_larg e MTZ_esp foram preenchidas com base nas Equações (3.13) e (3.14).

$$MTZ_larg_{ij} = abs(largura_i - largura_j) \forall i, j \in P \quad (3.13)$$

$$MTZ_esp_{ij} = abs(espessura_i - espessura_j) \forall i, j \in P \quad (3.14)$$

A função objetivo alterada para considerar estas diferenças está representada na Equação (3.15). Neste trabalho, como estas diferenças afetam o valor da função objetivo foi usado o termo penalidades. As novas variáveis DW_l e DT_l representam as penalidades calculadas para, respectivamente, largura e espessura. As Equações (3.16) e (3.17) retratam a expressão utilizada para o cálculo destas penalidades.

$$Min Z = \sum_{l \in F} \alpha E_l + \sum_{l \in F} \beta T_l + \sum_{l \in F} (DW_l + DT_l) \quad (3.15)$$

$$DW_l = \sum_{i,j \in P} (MTZ_larg_{ij} W_{ijl}) \forall l \in F, l < s_f \quad (3.16)$$

$$DT_l = \sum_{i,j \in P} (MTZ_esp_{ij} W_{ijl}) \forall l \in F, l < s_f \quad (3.17)$$

O cálculo destas penalidades ocorre da seguinte forma: buscam-se todos os valores das penalidades calculadas previamente na matriz MTZ_esp_{ij} para todos os pedidos i e j combinados dois a dois, que estão sequenciados nas posições l e $l + 1$ da fila F de execução e em seguida registra-se o valor encontrado no vetor DW_l (Equação (3.16)). Operação semelhante acontece para as diferenças de espessura (Equação (3.17)). Com estas modificações o modelo passou a sequenciar os pedidos com larguras e espessuras próximas umas das outras.

Na Seção 3.1.3 são tratadas as características relacionadas ao ciclo de recozimento, uso do laminador de encruamento e aspectos vinculados à aplicação de produto químico para proteção da chapa.

3.1.3 Formulação para Tratamento do Ciclo Térmico, Laminador de Encruamento e Aplicação de Tratamento Químico

Nas seções 3.1.1 e 3.1.2 foram abordados os aspectos relacionados ao cálculo das antecipações e atrasos e das penalidades associadas às características dimensionais. Nesta seção são tratados outros fatores que influenciam fortemente a qualidade do processamento dos materiais sequenciados. Estes fatores são o ciclo térmico, o laminador de encruamento e a aplicação do tratamento químico (TQ).

Diferentemente da abordagem utilizada para as características dimensionais, o ciclo térmico foi tratado utilizando-se um conjunto de restrições que impedem o sequenciamento inadequado de pedidos. No presente estudo os ciclos térmicos foram enumerados de 1 a 3, para representarem as faixas de temperaturas em que os materiais são processados. O processamento de qualquer material em um ciclo térmico errado provoca defeitos de planicidade e afeta as suas características mecânicas. Os materiais que, eventualmente, são processados em um ciclo térmico inadequado são descartados ou reclassificados, e um novo material é processado, no ciclo correto, para ser enviado ao cliente. Este erro provoca atraso na entrega das encomendas aos clientes e também um aumento no custo de produção da linha pelo desperdício dos insumos que não podem ser reciclados, como por exemplo gases, energia elétrica e produtos químicos.

Foi inserido no modelo matemático o vetor $ciclo_i$ contendo a numeração do ciclo térmico de cada um dos pedidos. Foi estabelecido que pode acontecer sequenciamento de

dois materiais de ciclos distintos desde que a diferença absoluta entre os ciclos seja de no máximo 1. Esta classificação térmica foi definida de forma crescente, tanto nos valores (de 1 até 3) quanto nas faixas de temperatura utilizadas no processo. Um material do ciclo 2 pode ser sequenciado após um material do ciclo 1 ou do ciclo 3. Em ambos os casos a diferença absoluta entre os ciclos foi unitária. Um exemplo de sequenciamento inválido é um pedido do ciclo 3 após um pedido do ciclo 1. Neste caso a diferença absoluta entre ciclos foi superior a 1. Foi aplicada a mesma estratégia de cálculo de utilizando uma matriz com as diferenças de ciclos de todos os pedidos combinados aos pares, e o resultado foi armazenada na matriz MTZ_ciclos_{ij} . Na equação (3.18) temos representado este cálculo. Com esta característica definida foi possível modificar o modelo matemático incluindo o novo conjunto de restrições, que é representado por (3.19).

$$MTZ_ciclos_{ij} = abs(ciclo_i - ciclo_j) \forall i, j \in P \quad (3.18)$$

$$W_{ijl} = 0; \forall i, j \in P : MTZ_ciclos_{ij} > 1 \quad (3.19)$$

As restrições (3.19) garantem o sequenciamento de materiais com ciclos distintos limitando a mudança a, no máximo, um degrau no ciclo térmico. Foi utilizada a variável de decisão W_{ijl} que representa a execução em sequência dos pedidos i e j , com o pedido i ocupando a posição l e o pedido j na posição $l + 1$ da fila de pedidos. Ela assumirá o valor zero em todas as posições onde a mudança registrada na matriz MTZ_ciclos_{ij} for superior a 1.

Além destas restrições foram associadas penalidades às mudanças de ciclos térmicos visando um agrupamento dos materiais. A cada mudança de ciclo ocorre um incremento no valor da penalidade. Utilizou-se a mesma matriz MTZ_ciclos_{ij} e através da Equação (3.20) foi feito o registro do valor total das penalidades de acordo com a mudança dos ciclos conforme a sequência de execução dos pedidos.

$$Cust_Ciclos_l = \sum_{i,j \in P} MTZ_ciclos_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.20)$$

O tratamento químico (TQ) consiste na aplicação de um produto sobre a chapa de aço após a galvanização com o objetivo de aumentar a proteção contra a corrosão branca. Neste trabalho foi considerado que todas as vezes onde ocorrem alternância na exigência do TQ é acrescentada uma penalidade para o sequenciamento, pois tem-se um trecho de

material cuja qualidade é comprometida durante estes instantes de mudança. De forma semelhante às características dimensionais, foi utilizado o modelo de penalidade para o TQ. O objetivo desta penalidade é minimizar os movimentos de ativação e desativação do sistema de aplicação do TQ. Foi gerada a matriz MTZ_TQ_{ij} para armazenar as penalidades associadas aos pedidos com TQ. Ela está representada na Equação (3.21) e a Equação (3.22) foi incluída no modelo matemático para considerar na função objetivo os valores registrados na matriz MTZ_TQ_{ij} .

$$MTZ_TQ_{ij} = abs(TQ_i - TQ_j) \forall i, j \in P \quad (3.21)$$

$$Cust_TQ_l = \sum_{i,j \in P} MTZ_TQ_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.22)$$

O laminador de encruamento caracteriza-se por ser um equipamento cujos objetivos principais do seu uso são a aplicação de rugosidade na tira, eliminação do patamar de escoamento e melhoria da planicidade da tira. No modelo matemático foi considerada uma pequena penalidade visando o agrupamento dos materiais que exigem o uso do laminador. A penalidade é calculada todas as vezes que houver a inclusão de um pedido sem o laminador e para isso foi criada a matriz MTZ_LE e apresenta-se na Equação (3.23) o cálculo utilizado para o seu preenchimento. É importante destacar que em todas as equações onde foram utilizadas matrizes foi feito o seu carregamento na etapa de pré-processamento, ou seja, antes da execução propriamente dita do modelo matemático. Os resultados dos cálculos são armazenados no vetor $Cust_LE_l$ e ele foi acrescentado na função objetivo. A Equação (3.24) demonstra o cálculo da penalidade incluída na função objetivo.

$$MTZ_LE_{ij} = MAX(0, (LE_i - LE_j)) \forall i, j \in P \quad (3.23)$$

$$Cust_LE_l = \sum_{i \in P} MTZ_LE_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.24)$$

A função objetivo foi revisada para contemplar todas as novas penalidades apresentadas, sendo elas o custo da mudança de ciclo térmico $Cust_Ciclos_l$, custo do TQ ($Cust_TQ_l$) e o custo do laminador ($Cust_LE_l$) ficando da forma como exposto na Equação (3.25).

$$Min Z = \sum_{l \in F} \alpha E_l + \sum_{l \in F} \beta T_l + \sum_{l \in F} (DW_l + DT_l + Cust_Ciclos_l + Cust_TQ_l + Cust_LE_l) \quad (3.25)$$

Para a produção de materiais com TQ exige-se, conforme já apresentado na Seção 3.1.3, uma preparação do equipamento de aplicação. Esta atividade de preparação inclui a substituição de algumas peças do sistema para que ele possa fazer a aplicação correta do produto químico. Ao final do ciclo estas peças não podem ser reutilizadas sem que haja uma nova substituição. Por este motivo existe uma restrição que estabelece um volume mínimo de material com TQ para que se possa iniciar o uso do sistema de aplicação. Sendo assim, para que o modelo matemático pudesse representar adequadamente o PSMU em estudo foi necessário incluir esta restrição. Ela garante que se não for sequenciado pelo menos o volume mínimo estabelecido, nenhum item com TQ será considerado.

Para este tratamento foi definido o conjunto de pedidos com TQ, chamado de CTQ , como sendo todos os pedidos do conjunto P que necessitam desta etapa de processamento adicional. Sendo assim o conjunto $CTQ \subset P$. A definição deste conjunto pode ser avaliada na Equação (3.26). Na Equação (3.27) está representado o conjunto de restrições associadas ao peso mínimo considerando o conjunto CTQ .

$$CTQ = \{P_i : P_i \in P, TQ_i = 1\} \quad (3.26)$$

$$\left(\sum_{i \in CTQ, l \in F} peso_pedido_i Z_{il} \geq Peso_Min \right) or \quad (3.27)$$

$$\left(\sum_{i \in CTQ, l \in F} peso_pedido_i Z_{il} = 0 \right)$$

O conjunto de restrições representado pela Equação (3.27) possui o seguinte comportamento: Se houver algum elemento do conjunto de pedidos com TQ sequenciado na fila é realizada a soma de todos os pesos destes pedidos inseridos na fila com TQ e, em seguida, verificado se o peso total é superior ao mínimo especificado ($Peso_Min$). Se esta condição não for atendida então a segunda parte da equação garante que não ocorra a seleção de pedidos com TQ. Esta equação não linear é convertida em expressões lineares através de um mecanismo interno e automático do *IBM ILOG CPLEX* de conversão, conforme apresentado no seu manual (IBM, 2017, P. 55). Neste mecanismo as expressões que apresentam restrições lógicas sofrem modificações e geram novas restrições e até novas

variáveis de decisão, de forma automática e transparente ao usuário. Podem ser utilizadas as expressões lógicas descritas a seguir:

- $\&\&$ \rightarrow AND: Operador lógico "E".
- $\|\|$ \rightarrow OR: Operador lógico "OU".
- $!$ \rightarrow NOT: Operador lógico "NÃO".

A partir da Equação (3.28) tem-se a modelagem matemática completa proposta neste trabalho. Esta modelagem inclui as considerações feitas por Bustamante e Carvalho (2006) a respeito do atraso e antecipação, bem como as ponderações discutidas neste trabalho referentes às características dimensionais, ao uso do laminador, à aplicação do tratamento Químico e à variação do ciclo Térmico.

$$\begin{aligned} \text{Min } Z &= \sum_{l \in F} \alpha E_l + \sum_{l \in F} \beta T_l + \\ &\sum_{l \in F} (DW_l + DT_l + Cust_Ciclos_l + Cust_TQ_l + Cust_LE_l) \end{aligned} \quad (3.28)$$

S.a.

$$\sum_{i \in P} Z_{il} = 1, \forall l \in F, \quad (3.29)$$

$$\sum_{l \in F} Z_{il} \leq 1, \forall i \in P \quad (3.30)$$

$$S_l + \sum_{i \in P} (Z_{il}(p_i - d_i)) + E_l - T_l = 0, \forall l \in F \quad (3.31)$$

$$S_{l+1} - S_l - \sum_{i \in P} p_i Z_{il} = 0, \forall l < s_f \quad (3.32)$$

$$W_{ijl} - Z_{il} - Z_{j(l+1)} \geq -1, \forall i \neq j, \forall i, j \in P, \forall l < s_f \quad (3.33)$$

$$\sum_{i, j \in P, i \neq j} \sum_{l \in F, l < s_f} W_{ijl} = s_f - 1 \quad (3.34)$$

$$DW_l = \sum_{i, j \in P} (MTZ_larg_{ij} W_{ijl}) \forall l \in F, l < s_f \quad (3.35)$$

$$DT_l = \sum_{i, j \in P} (MTZ_esp_{ij} W_{ijl}) \forall l \in F, l < s_f \quad (3.36)$$

$$\sum_{i, j \in P} \sum_{l \in F, l < s_f} MTZ_ciclos_{ij} W_{ijl} \leq 1 \quad (3.37)$$

$$Cust_Ciclos_l = \sum_{i, j \in P} MTZ_Alter_Ciclos_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.38)$$

$$Cust_TQ_l = \sum_{i, j \in P} MTZ_TQ_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.39)$$

$$Cust_LE_l = \sum_{i \in P} MTZ_LE_{ij} W_{ijl} \forall l \in F, l < s_f \quad (3.40)$$

$$\left(\sum_{i \in CTQ, l \in F} peso_pedido_i Z_{il} \geq \text{Peso_Min} \right) \text{ or}$$

$$\left(\sum_{i \in CTQ, l \in F} peso_pedido_i Z_{il} = 0 \right) \quad (3.41)$$

$$Z_{il} \in \{0, 1\}, \forall i \in P, \forall l \in F \quad (3.42)$$

$$W_{ijl} \in \{0, 1\}, \forall i, j \in P, \forall l \in F \quad (3.43)$$

$$S_l \geq 0, \forall l \in F \quad (3.44)$$

$$0 \leq E_l \leq E_{max}, \forall l \in F \quad (3.45)$$

$$0 \leq T_l \leq T_{max}, \forall l \in F \quad (3.46)$$

Por razões didáticas, a função objetivo e restrições da formulação matemática exposta de (3.28) à (3.46) são descritas novamente. A função objetivo (3.28) minimiza as antecipações, atrasos e as parcelas relacionadas aos critérios de qualidade considerados, sendo eles, na ordem em que aparecem na função objetivo: diferenças de largura, espessura, custo das mudanças de ciclo térmico, custo de inclusão dos pedidos com e sem tratamento químico e por último o custo associado ao fim das sequências de materiais sem laminador de encruamento. As restrições (3.29) garantem o preenchimento de todas as posições da fila de produção por algum pedido e as restrições apresentadas em (3.30) garantem que, se um pedido for sequenciado, ocupará apenas uma posição da fila de produção. Na equação (3.31) são calculadas as antecipações e os atrasos dos pedidos que estão na fila de produção. Em (3.32) tem-se o bloqueio de intervalos entre as produções dos pedidos, isto é, o próximo pedido sempre inicia imediatamente após a conclusão do anterior. Em (3.33) define-se o conjunto de restrições de enlace. Na equação (3.34) limita-se o número de posições preenchidas na variável de decisão W_{ijl} ao tamanho da fila -1. Este limite se deve ao fato de uma posição desta variável preenchida com o valor 1 representar um sequenciamento imediato de dois pedidos e, seguindo este raciocínio, na penúltima posição preenchida tem-se também o número do último pedido da sequência. As restrições compreendidas entre (3.35) e (3.41) estão associadas as exigências de qualidade do sequenciamento. As duas primeiras, (3.35) e (3.36), calculam as penalidades vinculadas as mudanças dimensionais entre os pedidos sequenciados. Na equação (3.37) há o bloqueio do sequenciamento de pedidos de ciclos térmicos incompatíveis. Em (3.38) ocorre o cálculo das penalidades oriundas das mudanças de ciclos térmicos. Ocorre na equação (3.39) o cálculo do valor da penalidade referente a inclusão dos pedidos com TQ. Em (3.40) são calculadas as penalidades originadas nas finalizações das sequências de materiais com laminador de encruamento. As restrições representadas na equação (3.41) impõem uma quantidade mínima de materiais com TQ a ser incluída em um sequenciamento. A partir da equação (3.42) definem-se os limites das variáveis de decisão.

A programação do modelo exato deste trabalho de pesquisa foi desenvolvida no software *IBM ILOG CPLEX Optimization Studio* versão 12.6.0.0 em plataforma 64bits. O computador utilizado foi um notebook Acer Aspire VX 15, com processador Intel I5 de 7ª geração, 8GB de memória DDR4 de 2400Mhz com sistema operacional Windows 10 64Bits.

Para a implementação do modelo matemático apresentado a partir da Equação (3.28) no *solver IBM ILOG CPLEX Optimization Studio* foi necessário utilizar o recurso de pré-processamento disponível na ferramenta. O uso desse recurso permitiu o preenchimento

de todas as matrizes associadas as características de qualidade dos pedidos antes da execução do processo de busca do resultado ótimo. Sem o uso do pré-processamento, o tempo de execução do *solver* havia sofrido um aumento considerável, pois todas as operações matemáticas de cálculos eram executadas a cada etapa do processo de busca do ótimo global. Com a execução do cálculo fora do processo iterativo, o *solver* conseguiu resolver o modelo gastando tempo apenas na busca pela melhor solução.

Todas as matrizes apresentadas nas Equações (3.13), (3.14), (3.18), (3.21) e (3.23) foram preenchidas na seção de pré-processamento do *solver*.

Para a entrada de dados do modelo exato utilizou-se uma planilha com os dados dos pedidos já normalizados seguindo a função descrita na Equação (3.47).

Na Seção 3.2 são descritas as etapas necessárias para a construção da heurística *BRKGA* e sua aplicação no PSMUAA estudado nesta dissertação.

3.2 Abordagem Heurística

Adotou-se o *framework* proposto por Gonçalves e Resende (2011) e Resende (2012) como base para o desenvolvimento desta meta-heurística. A sua implementação foi realizada conforme exposto no Algoritmo 1. O pseudo-código do *framework* BRKGA busca a minimização da função $f(x)$ e utiliza alguns parâmetros de entrada, sendo eles:

- $|P|$: Tamanho da população completa.
- $|P_e|$: Número de cromossomos elite que são transferidos entre as gerações.
- $|P_m|$: Número de cromossomos mutantes inseridos em cada geração.
- n : Quantidade de genes de um cromossomo.
- p_a : Probabilidade do filho herdar um alelo de um cromossomo elite.
- NP : Número de populações que evoluem independentemente.
- $Interv_troca$: Intervalo de gerações para ocorrer o intercâmbio de cromossomos elite entre as populações.
- $Quant_troca$: Número de cromossomos elite replicados entre as populações.

O cromossomo foi definido como um vetor cujo comprimento é idêntico ao número de pedidos do banco de dados acrescido de uma posição. Essa posição extra é utilizada para registrar o valor da função objetivo do cromossomo.

Como exposto na Seção 2.5, existe uma parte deste *framework* que é dependente do

Algoritmo 1 *Framework* do *BRKGA* adaptado de Gonçalves e Resende (2011).

```

1: BRKGA( $|P|, |P_e|, |P_m|, n, p_a$ )
2: Inicializar o valor da melhor solução encontrada:  $f^* \leftarrow \infty$ ;
3: Gerar  $NP$  populações com  $n$  vetores de chaves aleatórias;
4: Preencher o primeiro elemento da primeira população com o cromossomo resultante
   da função Init_pop();
5: enquanto Critério de parada não alcançado faça
6:   para  $indPop \leftarrow 1$  até  $NP$  faça
7:     Selecionar população  $indPop$ ;
8:     se  $numGerações = Interv\_troca$  então
9:       Replicar os  $Quant\_troca$  melhores cromossomos em todas as populações;
10:    fim se
11:    Calcular o valor da FO de cada cromossomo  $\in P$ ;
12:    Dividir a população  $P$  em dois conjuntos:  $P_e$  e  $P_{\bar{e}}$ ;
13:    Gerar o conjunto  $P_m$  de mutantes, cada um com  $n$  chaves aleatórias;
14:    Acrescentar  $P_m$  na população da nova geração:  $P^+ \leftarrow P^+ \cup P_m$ ;
15:    para  $i \leftarrow 1$  até  $|P| - |P_e| - |P_m|$  faça
16:      Selecionar aleatoriamente um cromossomo  $a$  do conjunto  $P_e$ ;
17:      Selecionar aleatoriamente um cromossomo  $b$  do conjunto  $P_{\bar{e}}$ ;
18:      para  $j \leftarrow 1$  até  $n$  faça
19:        Sortear uma moeda viciada com probabilidade  $p_a > 0.5$  de resultar cara;
20:        se resultado for cara então
21:           $c[j] \leftarrow a[j]$ ;
22:        senão
23:           $c[j] \leftarrow b[j]$ ;
24:        fim se
25:      fim para
26:      Acrescentar o filho  $c$  à população da nova geração:  $P^+ \leftarrow P^+ \cup \{c\}$ ;
27:    fim para
28:    Atualizar população:  $P \leftarrow P^+$ 
29:    Encontrar melhor solução  $X^+$  em  $P$ :  $x^+ \leftarrow \operatorname{argmin} \{f(x) | x \in P\}$ ;
30:    se  $f(x^+) < f^*$  então
31:       $x^* \leftarrow x^+$ ;
32:       $f^* \leftarrow f(x^+)$ ;
33:    fim se
34:  fim para
35: fim enquanto
36: retornar  $x^*$ ;

```

problema em estudo. Esta parte, no Algoritmo 1, está representada na linha 5, onde se faz o cálculo do valor da função objetivo de cada um dos elementos da população (ou cromossomos). Neste trabalho esta etapa foi dividida em duas: uma chamada **Decoder** e outra chamada **Métrica** que fazem, respectivamente, a conversão das chaves aleatórias em identificação dos pedidos e a segunda função faz a medição do valor do cromossomo (valor da função objetivo) decodificado previamente.

A função **Decoder** faz o mapeamento das chaves aleatórias com os respectivos pedidos da instância. Ela faz a ordenação crescente dos valores das chaves registradas nos genes de cada um dos cromossomos e retorna a identificação dos pedidos que são representadas pelas *tamFila* posições do cromossomo. Para melhor entendimento esta função está representada no Algoritmo 2.

Algoritmo 2 Função Decoder utilizada no *BRKGA*.

- 1: **Decoder**(*cromos*, *tamFila*)
 - 2: $n \leftarrow \text{comprimento}(\textit{cromos})$; ▷ Identificando o tamanho do cromossomo.
 - 3: $\textit{VetAux} \leftarrow \text{indice}(\text{ordenar}(\textit{cromos}))$; ▷ Ordenando genes e identificando as suas posições.
 - 4: $\textit{Decodificado} \leftarrow \text{subVetor}(\textit{VetAux}, \textit{tamFila}) \cup \textit{cromos}(n)$; ▷ Separando os *tamFila* primeiros elementos de *VetAux* e acrescentando o valor da função objetivo na última posição.
 - 5: **Retornar** *Decodificado*;
-

Após decodificar o cromossomo de chaves aleatórias faz-se necessário avaliar a sua aptidão ou o resultado da função objetivo desse cromossomo e para isso foi elaborada a função **Métrica**, que recebe como parâmetros de entrada o cromossomo decodificado, os dados da instância e o tamanho da fila de pedidos. O valor de saída é um número real que representa o cálculo da função objetivo. O Algoritmo 3 exibe o pseudo-código desta função.

Além destas duas funções dependentes do problema foi também implementada uma heurística construtiva que faz geração de um único cromossomo válido para a ocupação da primeira posição da população 1. O nome dado a função foi *Init_pop()*. O pseudo-código da implementação desta função pode ser visto no Algoritmo 4.

Com a implementação dos Algoritmos 1, 2 e 3 foi possível resolver o PSMUAA proposto nesta pesquisa. Os resultados dos testes do modelo aproximado podem ser vistos na Seção 4.2.

Especifica-se na Seção 3.3 a construção do banco de dados aplicado neste estudo, bem como definem-se as instâncias utilizadas nas simulações.

Algoritmo 3 Função Métrica para o *BRKGA*

```

1: Metrica( $\alpha, \beta, cromos, peso\_min$ )
2:  $MomInicio \leftarrow CalcularInicio(cromos)$ ;
3:  $Antec \leftarrow calcAntecip(cromos, MomInicio)$ ;
4:  $Atras \leftarrow calcAtrasos(cromos, MomInicio)$ ;
5:  $TL \leftarrow calcDifLarguras(cromos)$ ;
6:  $TE \leftarrow calcDifEspessura(cromos)$ ;
7:  $DifCiclos \leftarrow calcDifCiclos(cromos)$ ;
8: se  $max(DifCiclos) > 1$  então
9:   retornar  $\infty$ ;
10: fim se
11:  $PesoPedTQ \leftarrow CalculaPesoTQ(cromos)$ ;
12: se ( $PesoPedTQ < peso\_min$ ) então
13:   retornar  $\infty$ ;
14: fim se
15:  $QtdPedTQ \leftarrow ContarPedidosTQ(cromos)$ ;
16:  $QtdPedLamin \leftarrow ContarPedidosLamin(cromos)$ ;
17:  $Metrica = \alpha Antec + \beta Atras + (TL + TE + QtdPedTQ + DifCiclos + QtdPedLamin)$ ;
18: retornar  $Metrica$ ;

```

Algoritmo 4 Função *Init_pop()* usada no *BRKGA*.

```

1: Init_pop( $pedIni, bancoDados, MaxAntecip, MaxAtraso$ )
2: Calcular a data de conclusão do Pedido  $pedIni$  considerando o  $DiaDesejo$  somado com o seu  $TempoProcessamento (dtProx)$ .
3: Gerar um vetor de chaves aleatórias em ordem crescente( $Chaves$ );
4: Definir o ciclo térmico que pode ser sequenciado na posição 2 ( $cicloCompat$ );
5: Ordenar o  $bancoDados$  pelos campos  $cicloTermico, DiaDesejo$  e  $tratamentoQuimico$ ;
6: Identificar no  $bancoDados$  os pedidos que atendem aos seguintes requisitos ( $RAT2$ ):
  (( $DiaDesejo \geq dtProx$ ) E ( $DiaDesejo < dtProx + MaxAntecip$ )) E (Ciclo térmico compatível ( $cicloCompat$ ))
  OU
  (( $DiaDesejo < dtProx$ ) E ( $DiaDesejo > dtProx - MaxAtraso$ ))
7: Identificar no  $bancoDados$  os pedidos que não atenderam aos requisitos anteriores porém possuem ciclo térmico compatível ( $RAT1$ );
8: Montar a primeira parte do cromossomo resultado com os pedidos identificados em  $RAT2$  e atribuir valores às suas chaves a partir do vetor  $Chaves$ ;
9: Montar a segunda parte do cromossomo resultado com os pedidos identificados em  $RAT1$  e atribuir valores às suas chaves a partir do vetor  $Chaves$ ;
10: Montar a terceira parte do cromossomo resultado com os pedidos que não foram identificados em  $RAT1$  e  $RAT2$ , atribuindo os valores às suas chaves a partir do vetor  $Chaves$ ;
11: Unir as três partes em um único vetor ( $VetResposta$ );
12: Retornar  $VetResposta$ ;

```

3.3 Banco de Dados e Instâncias

Para a criação do banco de dados utilizado no trabalho foram feitas entrevistas com especialistas em linhas de galvanização. Com base nas informações obtidas nestas entrevistas foi possível identificar as características dos dados, os seus limites e outras informações relevantes. Com esse levantamento foi possível criar o banco de dados com valores simulados baseados nas características identificadas. A seguir tem-se uma listagem com as definições adotadas neste estudo. O banco de dados foi criado com 300 pedidos identificados sequencialmente do **PED.01** até **PED.300**.

As características dos campos e as etapas de elaboração do banco de dados podem ser avaliadas no Apêndice A.

Foram definidas 5 instâncias para testes e as suas características podem ser vistas no Quadro 3.1.

Quadro 3.1: Definição das instâncias para as simulações

Identificação	Número de pedidos	Composição
INST.A	35	Pedidos do 1 ao 35
INST.B	60	Pedidos do 1 ao 60
INST.C	100	Pedidos do 1 ao 100
INST.D	200	Pedidos do 1 ao 200
INST.E	300	Todos os pedidos

Foram definidos 4 tamanhos de fila de execução para os testes. A fila de execução representa a programação do sequenciamento da produção de um dia inteiro, e normalmente esta programação é executada no início do dia, sendo refeita no dia seguinte.

- **Fila com 6 posições:** para simular um ambiente com pedidos que apresentam tempos de execução de aproximadamente 4 horas.
- **Fila com 8 posições:** para simular um ambiente com pedidos que apresentam tempos de execução até 3 horas.
- **Fila com 12 posições:** para simular um ambiente com pedidos que apresentam tempos de execução até 2 horas.
- **Fila com 20 posições:** para simular um ambiente com pedidos que apresentam tempos de execução curtos, com aproximadamente 1 hora.

Todos os testes realizados foram com um pedido inicial definido, pois trata-se de um

cenário onde existe a necessidade de programar a nova fila de execução considerando o último pedido produzido, simulando uma fila de execução contínua.

Foram adotados 6 e 8, respectivamente, para os valores de α e β em todos os testes. Estes valores foram calibrados experimentalmente para atendimento da expectativa do especialista. Todos os testes foram realizados no mesmo computador, com somente o software *IBM ILOG CPLEX Optimization Studio* em execução.

Seguindo as características apresentadas, foram gerados dois bancos de dados, onde o primeiro consistiu nos resultados obtidos a partir das regras apresentadas, e baseado nele foi gerado o segundo banco de dados, porém com os campos *espessura*, *largura* e *dia desejo* normalizados linearmente conforme proposto por Fonseca (2017). O campo *tempo de processamento* foi dividido por 365 para representar o tempo em dias do ano necessário para o processamento deste pedido. A formulação adotada para esta normalização pode ser vista na Equação (3.47).

$$X_{norm} = \frac{X - X_{Min}}{X_{Max} - X_{Min}} \quad (3.47)$$

Onde X representa o valor a ser normalizado, X_{norm} é o valor normalizado, X_{Min} e X_{Max} representam os valores mínimo e máximo do conjunto de dados original.

Apresentam-se nas Figuras 3.1 e 3.2 as quantidades de pedidos do banco de dados agrupados por espessura e largura utilizado em todos os testes.



Figura 3.1: Distribuição dos pedidos por espessura.

Fonte: Próprio autor.

Pode ser visto na Figura 3.2 uma distribuição uniforme de pedidos em todas as faixas de larguras.

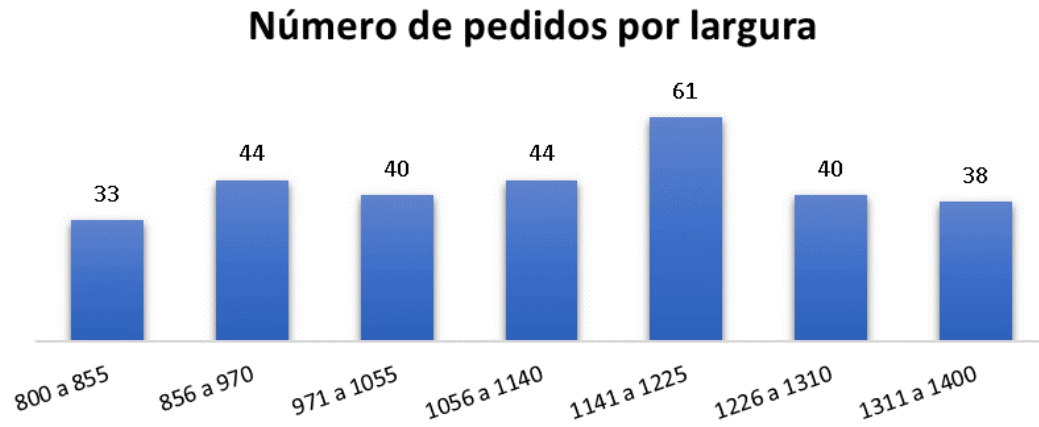


Figura 3.2: Distribuição dos pedidos por largura.
Fonte: Próprio autor.

Apresentam-se no Capítulo 4 os resultados encontrados nas simulações realizadas nesta pesquisa.

Capítulo 4

Resultados

São apresentados neste capítulo os resultados obtidos nos testes utilizando os modelos exato e heurístico. Descrevem-se na Seção 4.1 os resultados alcançados pelo modelo matemático e na Seção 4.2 os valores encontrados nas execuções do algoritmo genético *BRKGA*.

Para permitir a comparação entre os resultados alcançados pelos métodos exato e a heurística foi definido um indicador chamado **DIST** que representa a distância percentual comparando as respostas obtidas pelos modelos implementados, sendo que quanto menor este valor mais próximo do ótimo global o método aproximado chegou. A sua fórmula pode ser vista na Equação (4.1).

$$DIST(\%) = \frac{\text{Valor FO aproximado} - \text{Valor FO exato}}{\text{Valor FO exato}} * 100 \quad (4.1)$$

Um resultado de 20% significa que o modelo aproximado conseguiu alcançar uma sequência cujo valor da função objetivo foi 20% pior que o algoritmo exato. Se este resultado for 0% indica que o modelo aproximado chegou ao ótimo global.

4.1 Resultados do Modelo Matemático

Apresentam-se nessa seção os principais resultados obtidos com o modelo matemático exato desenvolvido. Estes resultados estão separados em subseções de acordo com o pedido inicial. Em cada uma delas avaliam-se os testes feitos nas instâncias, separados por tamanho da fila de execução.

A dimensão das instâncias testadas pode ser recordada no Quadro 4.1 e no Quadro

4.2 estão registrados os pedidos utilizados na primeira posição do sequenciamento de cada um dos testes, bem como algumas de suas características. Vale recordar que todas as simulações foram realizadas também variando o número de pedidos sequenciados. Os valores adotados foram 6, 8, 12 e 20 pedidos.

Quadro 4.1: Composição das instâncias testadas

Identificação	Número de pedidos	Composição
INST.A	35	Pedidos do 1 ao 35
INST.B	60	Pedidos do 1 ao 60
INST.C	100	Pedidos do 1 ao 100
INST.D	200	Pedidos do 1 ao 200
INST.E	300	Todos os pedidos

Quadro 4.2: Características dos pedidos testados

Pedido	TQ	Laminador	Ciclo térmico	Dia desejo
1	N	S	1	217
2	N	S	2	322
5	S	N	1	357
10	N	N	3	238

Todos os resultados gerados foram divididos em dois quadros com o primeiro contendo os dados associados ao calendário de produção e o segundo, os dados relacionados a qualidade dos pedidos sequenciados. São expostos na Seção 4.1.1 alguns dos resultados obtidos com o Pedido 1 na primeira posição da fila de execução.

4.1.1 Resultados - Pedido 1

Podem ser analisados nesta seção os resultados encontrados nas simulações realizadas com o Pedido PED.1 na primeira posição. Apresentam-se no Quadro 4.3 as principais características dos resultados encontrados pelo método exato implementado.

Observa-se no Quadro 4.3 que as sequências com 6 pedidos foram geradas em todas as instâncias e com o aumento do tamanho das instâncias o valor da função objetivo foi melhorando. A Instância INST.A não produziu uma sequência válida com fila composta por 8 pedidos pois violou o conjunto de restrições de atraso máximo.

As filas com 12 pedidos foram geradas completamente apenas nas Instâncias INST.B e INST.C. Não foi possível obter resultado na INST.A pois ela não possui pedidos suficientes para a criação de uma sequência válida e o processamento das Instâncias INST.D e INST.E foi finalizado antes de sua conclusão devido a ocupação de toda a memória disponível no computador.

Quadro 4.3: Características dos resultados para o Pedido PED.1

Fila	Inst.	Antecip.	Atraso	Produção	Tempo Gasto	Valor F.O.	Atenção
Pedidos	-	dias	dias	horas	segundos	-	-
6	A	21	7	13,0	0,08	6,82	Sim
	B	35	7	12,5	0,42	3,42	
	C	14	7	12,5	0,70	2,40	
	D	7	7	11,7	3,59	1,66	Sim
	E	0	7	10,8	6,40	1,21	
8	B	35	7	18,0	2,59	5,00	
	C	28	7	18,3	4,22	3,72	
	D	14	7	12,8	35,90	2,20	
	E	0	7	14,8	76,00	1,63	
12	B	35	7	28,5	171,80	3,42	Sim
	C	35	7	29,8	1348,00	2,40	

As três sequências que apresentaram pontos de atenção tiveram mudanças dimensionais superiores ao considerado adequado do ponto de vista prático. Os valores tidos como seguros são: mudanças de largura até 250mm e espessura até 0,30mm. Estes três sequenciamentos possuem mudanças de largura superiores aos 250mm e, se forem colocados em produção demandarão acompanhamento dos engenheiros e técnicos de operação para que sejam garantidos todos os requisitos de qualidade do produto e também sejam minimizados os riscos de defeitos nos materiais.

O sequenciamento obtido na Instância INST.A com fila composta por 6 pedidos pode ser visto nos Quadros 4.4 e 4.5.

Quadro 4.4: Sequenciamento com Pedido inicial 1 e 6 posições - INST.A

Posição	Dia	Hora	Duração	Antecipação	Atraso	Dia desejo	Identificação
1	216	23:14	0:46	-	-	217	PED. 1
2	217	0:00	2:19	-	7 dias	210	PED. 30
3	217	2:19	2:08	-	7 dias	210	PED. 8
4	217	4:27	2:07	21 dias	-	238	PED. 11
5	217	6:34	4:33	7 dias	-	224	PED. 13
6	217	11:07	0:50	21 dias	-	238	PED. 10

Os dados associados aos critérios de qualidade podem ser vistos no Quadro 4.5.

Pode ser visto no Quadro 4.5 a mudança de largura superior a 250mm entre os pedidos PED.11 e PED.13. Neste exemplo os engenheiros e técnicos de operação acompanharão o processamento para minimizar os riscos de aparecimento de defeitos de qualidade.

Os testes realizados com o Pedido 2 na primeira posição da fila de execução podem ser vistos na Seção 4.1.2.

Quadro 4.5: Atributos dimensionais, térmicos, TQ e laminador. Fila 6 - PED.1

Posição	Identificação	Largura	Espessura	Ciclo térmico	Com TQ	Laminador
1	PED. 1	961	0,69	1	N	S
2	PED. 30	1057	0,94	1	N	S
3	PED. 8	1221	0,76	1	N	S
4	PED. 11	1219	0,61	2	N	S
5	PED. 13	907	0,58	2	N	S
6	PED. 10	1124	0,81	3	N	N

4.1.2 Resultados - Pedido 2

São avaliados nesta seção os resultados obtidos considerando o Pedido 2 na primeira posição dos sequenciamentos. No Quadro 4.6 estão resumidas as características dos resultados gerados considerando o Pedido 2 na primeira posição. Merece destaque o fato de que quase todas as sequências geradas serem adequadas ao processamento sem qualquer preocupação do ponto de vista operacional.

Quadro 4.6: Características dos resultados para o Pedido PED.2

Fila	Inst.	Antecip.	Atraso	Produção	Tempo Gasto	Valor F.O.	Atenção
Pedidos	-	dias	dias	horas	segundos	-	-
6	A	14	1	9,5	0,2	2,87	
	B	14	7	9,0	0,4	2,50	
	C	14	7	12,0	2,1	2,30	
	D	7	7	9,7	5,9	0,96	
	E	7	7	9,7	17,0	0,96	
8	A	14	1	15,0	0,6	3,98	Sim
	B	27	7	14,0	3,2	3,51	
	C	14	7	15,8	12,3	3,00	
	D	21	7	13,0	135,0	1,79	
	E	21	7	13,0	558,7	1,79	
12	A	28	1	21,0	30,5	7,68	Sim
	B	35	7	24,0	176,3	6,04	Sim
	C	28	7	22,6	2.441,0	4,63	

Pode ser identificado no Quadro 4.6 que a maior antecipação foi de 35 dias e o menor atraso foi de apenas 1 dia. Verifica-se que o tempo de produção ficou entre 9 e 22,6 horas, e o maior tempo gasto para obtenção da sequência de produção foi na fila com 12 posições e Instância INST.C, com 2441 segundos, ou quase 41 minutos. As simulações realizadas com 12 pedidos nas Instâncias INST.D e INST.E não foram processadas até a sua finalização pois toda a memória disponível foi ocupada antes do final do processo.

As três sequências que apresentaram pontos de atenção tiveram mudanças de largura superiores aos 250mm considerados adequados do ponto de vista operacional. Um exemplo

pode ser visto nos Quadros 4.7 e 4.8. Neles são expostos os resultados do teste realizado na Instância INST.A com fila composta por 8 pedidos.

Quadro 4.7: Sequenciamento com Pedido inicial 2 e 8 posições - INST.A

Posição	Dia	Hora	Duração	Antecipação	Atraso	Dia desejo	Identificação
1	321	23:38	0:22	-	-	322	PED. 2
2	322	0:00	2:49	7 dias	-	329	PED. 33
3	322	2:49	2:29	-	-	322	PED. 15
4	322	5:18	0:44	14 dias	-	336	PED. 4
5	322	6:02	1:37	7 dias	-	329	PED. 26
6	322	7:39	1:32	14 dias	-	336	PED. 18
7	322	9:11	1:33	-	-	322	PED. 32
8	322	10:44	4:01	14 dias	-	336	PED. 29

Percebe-se no Quadro 4.7 a inexistência de atrasos e que a maior antecipação foi limitada a 14 dias. Também identifica-se que houve 3 pedidos sendo processados e entregues no prazo desejado.

Quadro 4.8: Atributos dimensionais, térmicos, TQ e laminador. Fila 8 - PED.2

Posição	Identificação	Largura	Espessura	Ciclo térmico	Com TQ	Laminador
1	PED. 2	1054	0,68	2	N	S
2	PED. 33	1154	0,68	2	N	S
3	PED. 15	1400	0,70	2	N	S
4	PED. 4	1366	0,73	1	N	S
5	PED. 26	1314	0,71	1	N	S
6	PED. 18	1137	0,74	1	N	S
7	PED. 32	872	0,77	1	N	S
8	PED. 29	914	0,59	1	N	S

Constata-se no Quadro 4.8 que as mudanças de espessura foram adequadas ao processamento porém houve uma mudança de largura superior a 250mm, entre os pedidos da 6^a e 7^a posições cujo valor foi 265mm. Neste exemplo haveria a necessidade de uma atenção especial da equipe operacional no início do processamento do Pedido PED.32 para que houvesse a minimização dos riscos de defeitos de qualidade.

Os resultados obtidos considerando o Pedido PED.5 na primeira posição podem ser avaliados na Seção 4.1.3.

4.1.3 Resultados - Pedido 5

Nesta seção são avaliados os resultados obtidos considerando o Pedido **PED.05** na primeira posição da fila de execução. O Quadro 4.9 contém as principais características das sequências obtidas pelo modelo matemático implementado.

Quadro 4.9: Características dos resultados para o Pedido PED.5

Fila	Inst.	Atraso	Produção	Tempo Gasto	Valor F.O.	Atenção
pedidos	-	dias	horas	segundos	-	-
6	B	7	12,0	0,3	7,87	Sim
	C	7	10,7	0,9	5,83	Sim
	D	7	12,7	1,4	1,56	Sim
	E	7	12,7	3,4	1,56	Sim
8	B	7	17,0	0,2	9,88	Sim
	C	7	14,8	0,6	6,95	Sim
	D	7	13,8	5,8	5,68	Sim
	E	7	15,7	7,7	3,12	Sim
12	D	7	17,65	177,8	7,10	
	E	7	18,6	902,8	6,21	

O Pedido PED.5 possui como característica principal a aplicação do tratamento químico. Esta característica faz com que o modelo matemático tenha que atender a mais um conjunto de restrições, conforme apresentado na Equação (3.27). Estas restrições garantem uma quantidade mínima de materiais com TQ a ser incluído no sequenciamento. Esta inclusão obrigatória de pedidos fez com que quase todas as sequências apresentassem pontos de atenção. Todos estes pontos de atenção estão relacionados com a mudança de largura entre pedidos superior ao valor 250mm, considerado adequado do ponto de vista operacional. Pode ser visto no Quadro 4.9 que não foram gerados sequenciamentos com 20 pedidos e isso se deve ao fato de não haver no banco de dados pedidos que atendam a todas as restrições impostas na modelagem e que sejam suficientes para a fila com 20 pedidos.

As únicas sequências que não apresentaram pontos de atenção foram aquelas geradas nas Instâncias INST.D e INST.E com fila composta por 12 pedidos. A sequência elaborada na INST.D pode ser vista nos Quadros 4.10 e 4.11. No Quadro 4.10 estão as suas características cronológicas. Percebe-se que existem apenas 3 atrasos de 7 dias.

No Quadro 4.11 verifica-se que apenas os primeiros 4 pedidos exigem a aplicação de TQ. Observa-se também que houve pedidos com e sem laminador, entretanto foram agrupados. As mudanças dimensionais deste sequenciamento foram inferiores aos limites práticos existentes.

Quadro 4.10: Sequência com 12 pedidos iniciando pelo PED.5 - INST.D

Posição	Dia	Hora	Duração	Antecipação	Atraso	Dia desejo	Identificação
1	356	22:47	1:13	-	-	357	PED. 5
2	357	0:00	1:33	-	-	357	PED.129
3	357	1:33	2:37	-	7 dias	350	PED. 56
4	357	4:10	0:50	-	-	357	PED. 79
5	357	4:60	0:37	-	7 dias	350	PED. 65
6	357	5:36	2:01	-	-	357	PED.130
7	357	7:37	0:50	-	-	357	PED.135
8	357	8:27	1:38	-	-	357	PED.105
9	357	10:05	1:03	-	-	357	PED.148
10	357	11:08	0:45	-	-	357	PED. 49
11	357	11:53	1:11	-	-	357	PED.154
12	357	13:04	3:22	-	7 dias	350	PED. 42

Quadro 4.11: Atributos dimensionais, térmicos TQ e laminador - PED.5

Posição	Identificação	Largura	Espessura	Ciclo térmico	Com TQ	Laminador
1	PED. 5	1051	0,61	1	S	N
2	PED.129	913	0,68	1	S	N
3	PED. 56	907	0,72	1	S	N
4	PED. 79	854	0,68	1	S	S
5	PED. 65	841	0,73	1	N	S
6	PED.130	972	0,78	1	N	S
7	PED.135	1055	0,74	1	N	S
8	PED.105	1161	0,62	1	N	S
9	PED.148	1197	0,83	1	N	S
10	PED. 49	1291	0,72	1	N	S
11	PED.154	1300	0,70	1	N	S
12	PED. 42	1390	0,73	1	N	S

4.1.4 Resultados - Pedido 10

Nesta seção avaliam-se os resultados encontrados nas simulações com o Pedido PED.10 na primeira posição do sequenciamento. Uma compilação de todas as simulações pode ser vista no Quadro 4.12.

Quadro 4.12: Compilação dos resultados com PED.10 na primeira posição

Fila	Inst.	Antecipação	Atraso	Produção	Tempo Gasto	Valor F.O.	Atenção
-	-	dias	dias	horas	segundos	-	-
6	A	35	0	10,28	0,17	5,57	Sim
	B	21	7	13,75	0,30	4,16	
	C	21	7	15,40	0,89	3,68	
	D	21	0	8,30	4,50	2,88	
	E	35	7	14,70	8,14	2,34	
8	A	35	0	14,76	0,25	8,61	Sim
	B	35	7	18,50	1,06	5,14	
	C	21	7	20,50	7,50	4,51	
	D	35	0	10,90	113,00	4,01	
	E	21	7	19,10	336,00	3,57	
12	B	35	7	24,20	160,00	8,36	

Constata-se no Quadro 4.12 a existência de dois resultados com pontos de atenção. Em ambas o limite prático de largura foi ultrapassado. Na sequência com 6 pedidos a mudança foi de 270mm e na simulação com 8 pedidos foi encontrado o valor 300mm. Nestes dois casos seria necessária a atenção da equipe de técnicos de operação no processamento da sequência e nos demais casos todas as programações poderiam ser executadas sem qualquer preocupação.

Outra característica que deve ser destacada no Quadro 4.12 é o aumento do tempo de processamento associado a mudança do número de pedidos na fila de produção.

Como se observa, uma mudança no tamanho da fila de pedidos provoca um aumento considerável no tempo de execução. Da fila com 6 para 8 posições houve um aumento no tempo de processamento de aproximadamente 3,67 vezes e saindo de 8 para 12 pedidos na fila de execuções encontra-se um aumento de aproximadamente 145,5 vezes.

Resumindo as análises feitas dos resultados obtidos pode-se afirmar que o algoritmo conseguiu encontrar boas e ótimas sequências de produção do ponto de vista prático, e em poucos casos apresentou mudanças dimensionais elevadas e que seriam acompanhadas pelos especialistas da linha de produção visando a redução dos riscos de aparecimento dos defeitos de qualidade.

Apresentam-se na Seção 4.2 as análises dos resultados encontrados para o PSMUAA

através da meta-heurística *BRKGA*.

4.2 Resultados do algoritmo *BRKGA*

Os resultados das simulações realizadas utilizando o *framework BRKGA* com os Pedidos PED.1, PED.2, PED.5 e PED.10 na primeira posição são apresentados nesta seção. Também foram feitas algumas comparações entre os algoritmos elaborados analisando o tempo de execução e a qualidade da resposta encontrada. O tempo de execução do *BRKGA* indicado em todos os quadros foi calculado pela média de tempo obtida em 10 execuções do algoritmo.

A definição dos parâmetros do *framework* foi feita seguindo as recomendações apresentadas por Chagas (2017), Brandão (2015) e com uma série de testes chegou-se ao seguinte conjunto de valores:

- $|P|$: $12 \times$ numero de pedidos da instância.
- $|P_e|$: 15% da população.
- $|P_m|$: 30% da população.
- n : número de pedidos da instância +1 para registrar o valor da função objetivo.
- p_a : 0,7.
- NP : 2.
- $Interv_troca$: 15.
- $Quant_troca$: 2.

Apresenta-se na Seção 4.2.1 a análise feita nos resultados considerando o Pedido PED.1 na primeira posição da fila de execução.

4.2.1 Resultados da heurística para o Pedido 1

Nesta seção são analisados os sequenciamentos obtidos pela heurística *BRKGA* com o pedido 1 na primeira posição. Além dessa análise também são realizadas comparações entre os algoritmos implementados.

Apresenta-se no Quadro 4.13 uma compilação dos resultados alcançados pelo algoritmo genético *BRKGA* em todas as instâncias mantendo-se o PED.1 na primeira posição. Alguns casos estão sinalizados com um (*) indicando que não foi possível chegar aos resultados completos do *CPLEX*. Isso se deve ao fato de ter ocorrido a ocupação de toda

a memória disponível antes da finalização do processo de busca.

Quadro 4.13: Comparativo entre o *BRKGA* e o método exato - PED.1

Fila	Inst.	Tempo	Distância	Tempo	Ótimo
Pedidos	-	BRKGA	Ótimo	Algor. exato	global?
6	A	244	0,5%	0,08	Sim
	B	477	0,0%	0,4	Sim
	C	774	0,0%	0,7	Sim
	D	1.703	0,7%	3,6	Sim
	E	2.396	0,4%	6,4	Sim
8	B	559	1,1%	2,6	Sim
	C	955	0,9%	4,2	Sim
	D	1.891	2,6%	35,9	Sim
	E	2.570	0,0%	76,0	Sim
12	B	683	0,6%	170,8	Não
	C	1.091	2,1%	1.348,0	Não
	D	2.094	-6,3%(*)	485,0(*)	N/D
	E	3.209	-11,7%(*)	906,0(*)	N/D
20	D	2.908	-22,4%(*)	423,0(*)	N/D
	E	4.460	-56,5%(*)	130,0(*)	N/D

Pode-se observar no Quadro 4.13 que o tempo de processamento do algoritmo genético *BRKGA* foi maior que o tempo do método exato em quase todas as simulações, ficando menor apenas na Instância INST.C com fila composta por 12 pedidos. Um dos principais fatores que afetaram a velocidade do *BRKGA* foi a sua implementação ter sido feita utilizando linguagem M e ter sido utilizado um interpretador para sua execução.

Na última coluna do Quadro 4.13 tem-se a sinalização das simulações onde o método aproximado conseguiu encontrar, no mínimo uma vez o valor ótimo global. O valor N/D significa que não foi possível obter o resultado completo do método exato entretanto o algoritmo *BRKGA* alcançou resultados em todos os testes permitindo a comparação entre os dois algoritmos.

Na coluna **Distância Ótimo** tem-se um valor percentual que representa o quão distante do ótimo global está a média dos resultados obtidos pelo *BRKGA*. Na primeira sequência o resultado alcançado pelo *BRKGA* foi 0,5% maior que o valor obtido pelo método exato. Esta coluna foi calculada considerando a média dos valores obtidos nas 10 execuções do algoritmo genético. Os resultados negativos indicam que o algoritmo *BRKGA* obteve soluções melhores que o *CPLEX*. Isso foi possível, pois como já apresentado, o processamento do método exato não foi concluído. Nestes casos o *CPLEX* reporta o melhor valor de solução inteira obtido até o momento e o *GAP* que é a diferença percentual entre os limites dual (melhor solução contínua) e primal (melhor solução inteira)

pesquisada. Os valores dos resultados incompletos podem ser vistos no Quadro 4.14. A última coluna foi calculada como $(A - B)/B * 100$, , sendo **A** a melhor solução obtida pelo *BRKGA* e **B** a melhor solução obtida pelo *CPLEX*.

Quadro 4.14: Resultados incompletos do *CPLEX* - PED.1

Fila	Inst.	(A) Valor FO	(B) Valor FO	<i>GAP</i>	Distância
Pedidos	-	<i>BRKGA</i>	Exato	Método Exato	Relativa
12	D	3,73	3,98	56,4%	-6,3%
	E	2,86	3,24	67,6%	-11,7%
20	D	8,51	10,97	83,6%	-22,4%
	E	6,43	14,80	95,3%	-56,5%

Através desse quadro percebe-se que o método exato não foi capaz de atingir o ótimo global nestas instâncias. Outro item a ser destacado foi a capacidade da meta-heurística *BRKGA* manipular as maiores instâncias e obter sequenciamentos válidos.

Percebe-se ainda que para instâncias com até 100 pedidos o método exato conseguiu encontrar resultados para as simulações com até 12 pedidos em tempos muito pequenos. Nos testes feitos nas Instâncias INST.D e INST.E o método exato não conseguiu encontrar soluções para as filas com 12 e 20 pedidos, entretanto o *BRKGA* os encontrou.

A simulação realizada na INST.E com fila de execução composta por 20 pedidos demandou 4460 segundos (aproximadamente 75 minutos). Com esta sequência criada seria possível manter a operação da linha por mais de 35 horas.

A análise dos resultados obtidos com o algoritmo genético *BRKGA* com o PED.2 na primeira posição pode ser vista na Seção 4.2.2.

4.2.2 Resultados da heurística para o Pedido 2

São analisados nesta seção os resultados obtidos pelo *BRKGA* mantendo o PED.2 na primeira posição da fila de execuções. No Quadro 4.15 encontram-se os dados destas simulações.

Quadro 4.15: Comparativo entre o *BRKGA* e o método exato - PED.2

Fila	Inst.	Tempo	Distância	Tempo	Ótimo
Pedidos	-	BRKGA	Ótimo	Algor. exato	global?
6	A	249,7	0,0%	0,2	Sim
	B	471,9	0,0%	0,4	Sim
	C	801,2	0,0%	2,1	Sim
	D	1774,6	36,0%	5,9	Sim
	E	2446,0	55,6%	17,0	Sim
8	A	258,8	0,2%	0,6	Sim
	B	520,1	0,0%	3,2	Sim
	C	869,8	0,5%	12,3	Sim
	D	2130,5	38,4%	135,0	Não
	E	2746,1	9,5	558,7	Não
12	A	342,5	8,9%	30,5	Não
	B	798,5	1,1%	176,3	Sim
	C	1226,8	3,2%	2.441,0	Sim
	D	2372,5	-4,4%(*)	2.641,0(*)	N/D
	E	3347,5	-17,1%(*)	1031,0(*)	N/D
20	C	1428,3	-3,3%(*)	1298,0(*)	N/D
	D	3079,7	-17,1%(*)	2442,0(*)	N/D
	E	4231,3	-77,3%(*)	115,0(*)	N/D

Novamente percebe-se que nas simulações com as instâncias até 100 pedidos o método exato alcançou resultados com tempos pequenos quando comparados ao algoritmo genético, entretanto, ele não conseguiu obter respostas para as simulações nas instâncias maiores. No Quadro 4.16 são detalhados os resultados incompletos obtidos pelo *CPLEX* nessas simulações.

Quadro 4.16: Resultados incompletos do *CPLEX* - PED.2

Fila	Inst.	(A) Valor FO	(B) Valor FO	Espaço	Distância
Pedidos	-	BRKGA	Exato	Restante	Relativa
12	D	3,7	3,8	65,6%	-4,4%
	E	6,5	7,8	85,7%	-17,1%
20	C	10,0	10,4	77,0%	-3,3%
	D	6,5	7,8	85,7%	-17,1%
	E	5,2	22,7	97,9%	-77,3%

Pode ser identificado que o método exato no teste com fila de 12 pedidos e instância INST.D apresentou um valor de função objetivo de 3,8 restando avaliar 65,6% do espaço de buscas. Em todos os testes listado no Quadro 4.16 o valor apresentado pelo *CPLEX* foi pior que o valor do *BRKGA*. É importante ressaltar que a meta-heurística desenvolvida conseguiu apresentar resultados válidos em todos os casos propostos.

Podem ser analisados os resultados do algoritmo genético *BRKGA* considerando o

PED.5 na primeira posição na Seção 4.2.3.

4.2.3 Resultados da heurística para o Pedido 5

Nesta seção são avaliados os resultados da heurística considerando o PED.5 na primeira posição. Os dados registrados no Quadro 4.17 representam os resultados obtidos.

Quadro 4.17: Comparativo entre o *BRKGA* e o método exato - PED.5

Fila	Inst.	Tempo	Distância	Tempo	Ótimo
Pedidos	-	BRKGA	Ótimo	Algor. exato	global?
6	B	582,5	0,0%	0,3	Sim
	C	859,6	0,0%	0,9	Sim
	D	1549,9	0,0%	1,4	Sim
	E	2506,5	0,2%	3,4	Sim
8	B	519,9	2,0%	0,2	Sim
	C	1130,7	4,3%	0,6	Sim
	D	1761,6	0,6%	5,8	Sim
	E	3166,6	38,8%	7,7	Não
12	D	2205,8	1,0%	177,8	Sim
	E	3326,1	3,1%	902,8	Não

Pode ser visto no Quadro 4.17 que não houve resultados para os testes com fila composta por 20 pedidos. Isso se deve ao fato de não haver quantidade suficiente de pedidos nas instâncias testadas que permitissem a construção de sequências de 20 pedidos sem a violação das antecipações ou atrasos máximos. Percebe-se ainda no mesmo quadro que algumas simulações apresentaram o valor 0,0% na coluna **Distância Ótimo**. Isso significa que o valor ótimo global foi alcançado em todas as execuções da heurística desenvolvida. Outro ponto a ser destacado é a ausência da INST.A, pois ela não possui pedidos suficientes para a montagem de uma sequência de produção que atenda todas as restrições impostas na modelagem.

São analisados na Seção 4.2.4 os resultados obtidos pela heurística mantendo o PED.10 na posição inicial da fila de produção.

4.2.4 Resultados da heurística para o Pedido 10

Nesta seção são apresentados os resultados do *BRKGA* com o Pedido PED.10 na posição inicial do sequenciamento. Observa-se no Quadro 4.18 que o algoritmo aproximado conseguiu obter resposta em todas as simulações realizadas. Nos resultados onde o valor **N/D** apareceu o método exato não concluiu o seu processamento, apresentando apenas

o resultado parcial e devido a isso não foi obtido o valor ótimo global. Estes casos foram destacados com (*) e no Quadro 4.19 são apresentados maiores detalhes destes resultados.

Quadro 4.18: Comparativo entre o *BRKGA* e o método exato - PED.10

Fila	Inst.	Tempo	Distância	Tempo	Ótimo
Pedidos	-	BRKGA	Ótimo	Algor. exato	global?
6	A	194,1	0,0%	0,2	Sim
	B	340,6	0,0%	0,3	Sim
	C	566,6	1,0%	0,9	Sim
	D	1158,0	0,0%	4,5	Sim
	E	1712,4	20,5%	8,1	Não
8	A	273,3	0,0%	0,3	Sim
	B	387,1	1,1%	1,1	Sim
	C	723,8	0,3%	7,5	Sim
	D	1232,5	2,1%	113,0	Sim
	E	2179,8	3,3%	336,0	Não
12	B	534,7	2,9%	160,0	Sim
	C	843,6	3,3%(*)	1250(*)	N/D
	D	1609,5	-15,0%(*)	1442(*)	N/D
	E	2912,4	-19,4%(*)	392(*)	N/D
20	C	1148,1	1,1%(*)	1606(*)	N/D
	D	2350,8	-15,0%(*)	1442(*)	N/D
	E	3428,0	-58,9%(*)	114(*)	N/D

Constata-se no Quadro 4.18 que o *BRKGA* foi mais lento em todas as simulações efetuadas, entretanto, alcançou resultados em todos os testes feitos. Novamente, em boa parte das simulações o resultado ótimo global foi alcançado, somente em dois testes que não foi obtido o ótimo global. Nas simulações incompletas não se alcançou o valor ótimo e por isso não houve comparação direta. Algumas características dos resultados incompletos podem ser vistas no Quadro 4.19.

Quadro 4.19: Resultados incompletos do *CPLEX* - PED.10

Fila	Inst.	(A) Valor FO	(B) Valor FO	Espaço	Distância
Pedidos	-	BRKGA	Exato	Restante	Relativa
12	C	7,1	6,9	29,1%	3,1%
	D	5,5	5,7	40,9%	-2,5%
	E	4,9	6,1	80,0%	-19,4%
20	C	14,1	13,9	61,1%	1,0%
	D	9,0	10,6	72,8%	-15,2%
	E	7,8	19,0	94,3%	-58,9%

As simulações do método exato realizadas com filas compostas por 12 e 20 pedidos na instância INST.C, mesmo sendo finalizadas prematuramente apresentaram resultados ligeiramente melhores que os encontrados pela meta-heurística. Nos demais casos o

BRKGA obteve resultados melhores. Nos testes realizados na maior instância com a fila de 20 pedidos, o *BRKGA* obteve uma solução 58,9% melhor que o resultado parcial obtido pelo *CPLEX*.

O algoritmo genético *BRKGA* produziu um sequenciamento válido em todas as simulações realizadas. No Capítulo 5 são apresentadas as conclusões deste trabalho de pesquisa.

Capítulo 5

Conclusões e Trabalhos Futuros

5.1 Conclusões

Este trabalho apresentou o modelamento de um problema real da indústria metalúrgica na fabricação de aços galvanizados em linha de produção contínua. As características que mais influenciam o sequenciamento da linha de produção foram considerados, como por exemplo a largura, espessura e ciclo térmico. Foram destacadas as dificuldades para uma equipe de profissionais produzir um sequenciamento otimizado que considere todas estas características, inclusive do ponto de vista econômico

Foram apresentadas duas abordagens para a resolução do problema, uma formulação matemática baseada no trabalho de Bustamante e Carvalho (2006), descrita em detalhes na Seção 2.3 e uma meta-heurística baseada em *BRKGA*, descrita em detalhes na Seção 2.5.

Constatou-se que o modelamento matemático desenvolvido e implementado utilizando programação linear inteira mista e a meta-heurística *BRKGA* foram capazes de resolver o problema de sequenciamento de máquina única com minimização dos atrasos e antecipações, além de conseguir a redução das diferenças dimensionais entre os pedidos sequenciados. O modelo exato foi capaz de resolver sequências com 12 pedidos em instâncias com até 100 pedidos e o método aproximado foi resolveu todas as simulações propostas alcançando o resultado ótimo em mais de 80% dos casos. As duas abordagens produziram seus resultados em um tempo adequado ao uso diário.

A definição dos parâmetros α e β permite ao decisor focar a busca em aspectos específicos ou encontrar um ponto de equilíbrio entre minimização das antecipações/atrasos e as mudanças de ciclo térmico, inclusão ou não de pedidos com TQ e a redução da alternância

entre pedidos com e sem laminador. Esta flexibilidade é fundamental para as operações cotidianas da linha de galvanização.

O *framework BRKGA*, apesar de produzir resultados válidos em todos os testes apresentou tempo de processamento maior que o método exato.

O método exato não alcançou resultados completos nas instâncias com mais de 100 pedidos e nas simulações com filas de 12 ou 20 posições devido a ocupação de toda a memória disponível. Nesses casos o *BRKGA* conseguiu obter resultados válidos mesmo apresentando tempo de processamento maior.

Foi verificado que o problema de sequenciamento da linha de galvanização contínua pode ser resolvido adequadamente pelas modelagens propostas neste trabalho. Nas filas com 6 e 8 pedidos o método exato demandou tempos inferiores a 11 segundos, o que caracteriza uma solução aplicável ao uso diário e por várias vezes ao dia sem uma demanda demasiada de tempo computacional.

Outra contribuição desta pesquisa consiste na comprovação da capacidade do método exato e do *framework BRKGA* de resolverem satisfatoriamente o problema de sequenciamento da linha de galvanização contínua estudado nesse trabalho, com todas as suas particularidades.

Na grande maioria dos sequenciamentos obtidos como resultado dos modelos desenvolvidos todos os requisitos de qualidade apresentados foram atendidos. Houve a minimização das antecipações e atrasos, as mudanças dimensionais foram reduzidas, houve o agrupamento dos pedidos por ciclo térmico, também foram agrupados os pedidos com e sem tratamento químico além dos pedidos com e sem laminador de encruamento.

Em alguns resultados apresentados houve mudanças dimensionais que mereceram destaque. Isso se deve ao fato de serem superiores aos valores considerados seguros do ponto de vista prático, entretanto, mesmo com esses valores, as sequências construídas seriam produzidas atendendo aos requisitos de qualidade citados na Seção 1.1, cabendo aos operadores de processo, o acompanhamento da transposição durante o seu processamento.

Os modelos foram implementados considerando valores fixos de α e β , pois assim atenderam as necessidades associadas ao problema de sequenciamento da linha de galvanização contínua estudado. Entretanto, com pequenas modificações, é possível a utilização de valores de α e β que podem variar de acordo com os pedidos ou também podem ser fornecidos a partir de algum outro sistema de gestão da empresa, alterando-os dinamicamente em função das demandas do momento.

Os resultados alcançados nessa pesquisa apresentam alguma semelhança com os obtidos no trabalho dos autores Cohen, Foxx e Alul (2018), onde foram consideradas as características largura e espessura, e dos autores GAO e QU (2018), na minimização das mudanças de largura. As demais características abordadas nessa dissertação não foram encontradas nas literaturas estudadas e por isso não se realizou qualquer comparação direta dos resultados pois as abordagens não consideraram os mesmos atributos.

São sugeridos alguns trabalhos futuros na Seção 5.2.

5.2 Sugestões para Trabalhos Futuros

Para melhorar o tempo de convergência do algoritmo *BRKGA* sugere-se a implementação de uma heurística construtiva que forneça uma população inicial válida.

Recomenda-se uma investigação a respeito da performance do algoritmo proposto em um ambiente com processamento paralelo ativo, minimamente das funções *Decoder* e *Métrica*.

Referências

- AÇO BRASIL, I. *Processo Siderúrgico*. 2019. Disponível em: <www.acobrasil.org.br>. Acesso em: 27 jul. 2019.
- ADAMU, M. O.; ADEWUMI, A. O. A survey of single machine scheduling to minimize weighted number of tardy jobs. *Journal of Industrial & Management Optimization*, p. 219, 2014. ISSN 1547-5816.
- ARMELLINI, D. et al. Modeling and solving the steelmaking and casting scheduling problem. *International Transactions in Operational Research*, v. 27, n. 1, p. 57–90, 2020.
- BAKER, K. R.; TRIETSCH, D. *Principles of Sequencing and Scheduling: Theory, algorithms, and systems*. 1. ed. New Jersey: John Wiley Sons, Inc., 2009. 510 p. ISBN 978-0-470-39165-5.
- BEAN, J. C. Genetic algorithms and random keys for sequencing and optimization. *ORSA - Journal on Computing*, p. 154–160, 1994.
- BELFIORE, P.; FÁVERO, L. P. *Pesquisa Operacional Para Cursos de Engenharia*. Rio de Janeiro: Elsevier Brasil, 2013. ISBN 9788535263350.
- BRANDÃO, J. S. *Algoritmos Genéticos com Chaves Aleatórias Tendenciosas para Problemas de Otimização em Redes*. 111 p. Tese (Doutorado) — UNIVERSIDADE FEDERAL FLUMINENSE, Programa de Pós-Graduação em Computação da Universidade Federal Fluminense, Niterói, RJ, 2015.
- BRETAS, A. M. C. *Modelagem do Sequenciamento no Processo de Lingotamento Contínuo por Métodos Exatos e Heurísticos*. 120 p. Dissertação (Dissertação) — Universidade Federal de Minas Gerais, Departamento de Engenharia de Produção / Escola de Engenharia, Julho 2013.
- BRUCKER, P. *Scheduling Algorithms*. 5. ed. Nova York: Springer, 2006. ISBN 978-3-540-69515-8.
- BUSTAMANTE, L. de M.; CARVALHO, C. R. V. de. Minimização do custo de antecipação e atraso para o problema de sequenciamento de uma máquina com tempo de preparação dependente da sequência. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL. Goiânia, GO: SBPO, 2006. p. 12.
- CARVALHO, P. S. L. de et al. Sustentabilidade da siderurgia brasileira: eficiência energética, emissões e competitividade. n. 41, p. 56, abril 2015.
- CAVALHEIRO, E. M. B.; FILHO, C. L. Otimização de redes de distribuição de energia elétrica por brkga. 2017.

- CHAGAS, J. B. C. Biased random-key genetic algorithms: Algoritmos genéticos de chaves aleatórias viciadas. p. 26, Dezembro 2017.
- COHEN, M. W. et al. A decision support flexible scheduling system for continuous galvanization lines using genetic algorithm. *Production Engineering*, p. 43–52, 2018.
- DEWA, M.; NLEYA, B. Parallel processors scheduling algorithms to minimise makespan in a galvanising plant. *PONTE International Scientific Researches Journal*, p. 9, 2020.
- FONSECA, T. L. *Algoritmo Genético com Regressão: Busca direcionada através de aprendizado de máquina*. 108 p. Dissertação (Mestrado) — Instituto de Ciências Exatas, Agosto 2017.
- FUCHIGAMI, H. Y. Introdução ao sequenciamento da produção. 2016.
- GAO, C.; QU, D. A modelling and a new hybrid milp/cp decomposition method for parallel continuous galvanizing line scheduling problem. *ISIJ International*, ISIJ International, v. 58, n. 10, p. 1820–1827, mar 2018.
- GONÇALVES, J. F.; RESENDE, M. G. Random-key genetic algorithm. *J. of Heuristics*, p. 487–525, 2011.
- IBM. *ILOG CPLEX Optimization Studio: Opl language reference manual*. EUA, 2017. 146 p.
- JR., E. F. S.; TSENG, F. T. Two milp models for the ssist flowshop sequencing problem. *DSI National Meetings*, Alabama, USA, p. 11, nov 2001.
- JÚNIOR, J. H. N. *Contribuição ao estudo da cinética de redução de briquetes auto-redutores*. 140 p. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Ciência dos Materiais e Metalurgia., setembro 2002.
- JUNIOR, L. O. de A. *Método de Programação de Sistemas de Manufatura do Tipo JOB SHOP Dinâmico Não Determinístico*. Tese (Doutorado) — USP, 2006.
- KAWAMURA, M. S.; RONCONI, D. P. Aplicação do método branch-and-bound na programação de tarefas em uma única máquina com data de entrega comum sob penalidades de adiantamento e atraso. In: ABEPRO. *XXVI ENEGEP*. Fortaleza, CE Brasil: ABEPRO, 2006.
- KIM, Y.-D.; YANO, C. A. *Minimizing Mean Tardiness and Earliness in Single Machine Scheduling Problems With Unequal Due-Dates*. Korea, 1992.
- LEI, D.; CAI, J. Multi-population meta-heuristics for production scheduling: A survey. *Swarm and Evolutionary Computation*, v. 58, p. 100739, 2020. ISSN 2210-6502.
- LEVI, D. S. et al. *Cadeias de Suprimentos Projeto E Gestão*. São Paulo: Bookman, 2003. ISBN 9788536301198.
- LOPES, M. F. A. et al. Um algoritmo de evolução diferencial para o problema de job shop scheduling bi-objetivo. p. 12, Agosto 2018.

- MAINIERI, G. B. *Meta-heurística BRKGA aplicada a um problema de programação de tarefas no ambiente flowshop híbrido*. 110 p. Tese (Doutorado) — Escola Politécnica da Universidade de São Paulo, 2014.
- MARTIN, M. P. et al. Métodos exatos e uma abordagem heurística para o problema de alocação de berços. In: *Anais do XLVII Simpósio Brasileiro de Pesquisa Operacional - SBPO*. Porto de Galinhas - PE: SBPO, 2015.
- MARTÍNEZ, C. et al. Brkga algorithm for the capacitated arc routing problem. *Electronic Notes in Theoretical Computer Science*, p. 69–83, 12 2011.
- MELIÁN, B. et al. Metaheurísticas: una visión global. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, p. 7–28, 2003.
- MORO, M. A. *Meta-heurísticas GRASP e BRKGA aplicadas ao problema da diversidade máxima*. 55 p. Dissertação (Mestrado) — Instituto de Matemática, Estatística e Computação Científica - Campinas - SP, Julho 2017.
- PARVEEN, S.; ULLAH, H. Review on job-shop and flow-shop scheduling using multi criteria decision making. *Journal of Mechanical Engineering*, v. 41, n. 2, p. 130–146, Abril 2011.
- PIMENTA, L. de B. *Otimização no sequenciamento de produção em uma fábrica de materiais médico-hospitalares*. 48 p. Monografia — Universidade Federal de Juiz de Fora, Departamento de Engenharia de Produção, novembro 2008.
- PINEDO, M. L. *Scheduling: Theory, algorithms, and systems*. 3. ed. Nova York: Springer, 2008. 662 p. ISBN 978-0-387-78934-7.
- PIRES, C. T. de Ávila. *Sistema de Otimização e Adaptação para a Geração de Referências em um Laminador de Tiras a Frio*. 112 p. Tese (Doutorado) — USP, 2007.
- PRASETYO, H. et al. Solving ccvrptw using biased random key genetic algorithm (brkga) with multiple parent. In: . Aceh Province, Indonesia.: ICEAT, 2018.
- PRODANOV, C. C.; FREITAS, E. C. de. *metodologia do trabalho científico: Métodos e técnicas da pesquisa e do trabalho acadêmico*. 2. ed. [S.l.: s.n.], 2013. ISBN 978-85-7717-158-3.
- RAJABZADEH, M. et al. Single machine scheduling with different types of transportation facilities in batch delivery system. In: *Joint International Symposium on the Social Impacts of Developments in Information, Manufacturing and Service Systems 2014*. Istanbul - Turquia: Computers and Industrial Engineering, 2014. p. 1441–1450.
- RAVETTI, M. G. *Problemas de sequenciamento com máquinas paralelas e tempos de preparação dependentes da seqüência*. Dissertação (Mestrado) — UFMG, Dezembro 2003.
- REBOUÇAS, A. D. C. *Desenvolvimento de uma Metaheurística BRKGA com Infecção Viral para o Problema de Atribuição de Localidades a Anéis em Redes SONET/SDH*. 57 p. Monografia — Universidade do Estado do Rio Grande do Norte- Mossoró- RN, Julho 2018.

- RENN, D. et al. Penalty-based sequencing strategy implemented within a knowledge-based system. In: LTD, O. R. S. (Ed.). *Journal of the Operational Research Society*. EUA: Taylor & Francis, 1999. v. 50, n. 3, p. 6.
- RESENDE, M. G. C. Biased random-key genetic algorithms: A tutorial. *XVI CLAIOXLIV SBPO 2012*, SBPO, Rio de Janeiro - RJ, p. 12, Setembro 2012.
- RIZZI, M. M. et al. Metaheurística híbrida aplicada ao problema de job shop. In: *XLVII Simpósio Brasileiro de Pesquisa Operacional - SBPO*. Porto de Galinhas - PE: SOBRAPO, 2015. p. 1735–1744.
- RODRIGUES, G. S. *O problema do sequenciamento em uma única máquina, com tempos de preparação dependentes da sequência e penalidades por antecipação e atraso: Estudo de caso de um processo de fabricação por usinagem*. 120 p. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio de Janeiro - PUC-RIO, Junho 2012.
- RODRIGUES, V. M. et al. Programação linear aplicada a uma microempresa de comunicação visual. In: AEDB. Resende-RJ, Brasil: AEDB, 2014. p. 11.
- Salzgitter AG. *Setup of a third hot-dip galvanizing line*. 2020.
- SANTOS, V. L. A. *Sequenciamento de Tarefas em Máquinas Paralelas com Desgastes Dependentes da Sequência: Resolução heurística*. 92 p. Dissertação (Mestrado) — Universidade Federal de Viçosa, 2016.
- SPEARS, W. M.; De Jong, K. A. On the virtues of parameterized uniform crossover. In: *Fourth International Conference on Genetic Algorithms*. EUA: Springer, 1991. p. 230–236.
- TALBI, E.-G. *Metaheuristics: From Design to Implementation*. : John Wiley and Sons, 2009. (Wiley Series on Parallel and Distributed Computing). ISBN 9780470496909.
- TAN, M. et al. Optimizing production scheduling of steel plate hot rolling for economic load dispatch under time-of-use electricity pricing. *Mathematical Problems in Engineering*, Hindawi, v. 2017, n. 10, p. 13, mar 2017.
- TANG, L.; GAO, C. A modelling and tabu search heuristic for a continuous galvanizing line scheduling problem. *ISIJ International*, ISIJ International, v. 49, n. 3, p. 375–384, mar 2009.
- TYAGI, N. et al. Single machine scheduling model for total weighted tardiness. p. 8, 2016.
- VALENTE, J. M. S. et al. Genetic algorithms for single machine scheduling with quadratic earliness and tardiness costs. p. 251–265, 2011.
- VIANA, F. L. E. Indústria siderúrgica. v. 2, n. 13, p. 13, Agosto 2017.
- VIANA, F. L. E. Indústria siderúrgica: Investir em eficiência. v. 3, n. 39, p. 14, Agosto 2018.
- WIERCINSKI, A. *Galvanoplastia: Melhorias no processo de zincagem eletrolítica*. 112 p. Monografia — UNIVERSIDADE REGIONAL DO NOROESTE DO ESTADO DO RIO GRANDE DO SUL, 2015.

WINSTON, W. L. *Operations Research: Applications and algorithms*. 4. ed. Canada: Thomson Learning, 2004. 1434 p. ISBN 0-534-52020-0.

YEN, B.; WAN, G. Single machine bicriteria scheduling: A survey. *International Journal of Industrial Engineering : Theory Applications and Practice*, p. 222–231, 09 2003.

YOUSEFI, M.; YUSUFF, R. M. Minimizing earliness and tardiness penalties in a single machine scheduling against common due date using genetic algorithm. p. 6, maio 2012.

APÊNDICE A – Construção do Banco de Dados

As características dimensionais, dia desejo e tempo de processamento de todo o banco de dados foram definidas assim:

- **Largura (mm)**: distribuição uniforme entre 800 e 1400.
- **Espessura(mm)**: distribuição normal com média 0,7 e desvio padrão de 0,1.
- **Peso do pedido(Kg)**: distribuição uniforme entre 7.500 e 150.000. Vale salientar que este peso não é de uma bobina, e sim de cada um dos pedidos.
- **Dia desejo**: calculada em função da semana solicitada pelo cliente. Foi utilizada uma distribuição uniforme entre as semanas 30 e 51. Após definida a semana, multiplicou-se por 7 para representar o dia desejo utilizado no modelo elaborado.
- **Tempo de processamento (dias)**: calculado conforme exposto na Equação (A.1). As unidades foram ajustadas de milímetro para metro, a densidade do aço usada foi 7,86t/m³ e o peso foi convertido de kg para t (tonelada). O resultado desta equação representa o tempo em dias necessário para produzir um determinado pedido. Por exemplo: se o resultado for 1,5 significa que será necessário um dia e meio para concluir o processamento do pedido.

$$T_{proc} = \frac{peso}{1000 \cdot largura \div 1000 \cdot espessura \div 1000 \cdot VPD \cdot 60 \cdot 7,86 \cdot 24} \quad (A.1)$$

O campo *grupo de materiais* representa a norma que rege as especificações técnicas daquele material, e foram divididas em 4 grupos como detalhado a seguir. A classificação do grupo de materiais dos pedidos no banco de dados foi executada nesta ordem para garantir a definição correta dos grupos.

1° Pedidos do grupo 1.

2° Pedidos do grupo 4.

3° Pedidos do grupo 2.

4° Pedidos do grupo 3.

Grupo 1:

- Aproximadamente 30% dos pedidos com espessura entre 0,5 e 0,7 mm pertencem a este grupo.
- Ciclo térmico definido com valor 2 para todos os pedidos.
- Nenhum pedido com tratamento químico.
- Laminador utilizado em aproximadamente 90% dos pedidos.
- Velocidade padrão (VPD): 130 mpm para todos os pedidos.

Grupo 2:

- Cerca de 15% dos pedidos que não foram classificados nos grupos 1 e 4 entraram neste grupo.
- Ciclo térmico definido com valor 1 para todos os pedidos.
- Nenhum pedido com tratamento químico.
- Laminador utilizado em aproximadamente 98% dos pedidos.
- Velocidade padrão (VPD): 130 mpm para todos os pedidos.

Grupo 3:

- Ciclo térmico definido com valor 1 para todos os pedidos.
- Pedidos que não foram classificados nos grupos 1, 2 e 4 foram incluídos neste grupo.
- Cerca de 40% dos materiais exigem tratamento químico.
- Cerca de 50% dos materiais com o tratamento químico utilizam o laminador.
- A velocidade padrão (VPD) foi definida conforme regras abaixo:
 - Espessura menor ou igual a 0,65 → 145 mpm.
 - Espessura entre 0,65 e 0,7 → 110 mpm.
 - Espessura entre 0,7 e 0,8 → 90 mpm.
 - Espessura maior que 0,8 → 85 mpm.

Grupo 4:

- Ciclo térmico definido com valor 3 para todos os pedidos.

- Cerca de 8% dos pedidos que possuem espessuras a partir de 0,68 mm com largura superior a 1000mm e que não foram classificados no grupo 1, foram categorizados no grupo de materiais 4.
- Nenhum pedido com tratamento químico.
- Laminador utilizado em todos os pedidos.
- Velocidade padrão (VPD): 80 para todos os pedidos.