



UNIVERSIDADE
ESTADUAL DE LONDRINA

POLIANE CRISTINA DE FARIAS

**PROBLEMA DE CORTE DE ESTOQUE UNIDIMENSIONAL
COM REAPROVEITAMENTO DE SOBRAS:
ABORDAGEM DE RESOLUÇÃO POR MEIO DE UMA TÉCNICA
DE GERAÇÃO DE COLUNAS**

Londrina
2011

POLIANE CRISTINA DE FARIAS

**PROBLEMA DE CORTE DE ESTOQUE UNIDIMENSIONAL
COM REAPROVEITAMENTO DE SOBRAS:
ABORDAGEM DE RESOLUÇÃO POR MEIO DE UMA TÉCNICA
DE GERAÇÃO DE COLUNAS¹**

Dissertação de mestrado apresentada ao Departamento de Matemática da Universidade Estadual de Londrina, como requisito parcial para a obtenção do Título de MESTRE em Matemática Aplicada e Computacional.

Orientador: Prof. Dr. Robinson Samuel Vieira Hoto

Londrina
2011

¹ Trabalho realizado com o apoio financeiro da CAPES

**Catálogo elaborado pela Divisão de Processos Técnicos da Biblioteca Central da
Universidade Estadual de Londrina**

Dados Internacionais de Catalogação -na-Publicação (CIP)

F224p Farias, Poliane Cristina de.

Problema de corte de estoque unidimensional com reaproveitamento de sobras
: abordagem de resolução por meio de uma técnica de geração de colunas /
Poliane Cristina de Farias. – Londrina, 2011.
93 f. : il.

Orientador: Robinson Samuel Vieira Hoto.

Dissertação (Mestrado em Matemática aplicada e Computacional) – Universidade
Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós -Graduação em
Matemática Aplicada e Computacional, 2011.

Inclui bibliografia.

1. Programação linear – Teses. 2. Reaproveitamento (Sobras, refugos, etc.)
– Teses. 3. Problema do corte de estoque – Teses. 5. Otimização matemática – Teses.
6. Pesquisa operacional – Teses. I. Hoto, Robinson Samuel Vieira. II. Universidade
Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em
Matemática Aplicada e Computacional. III. Título.

CDU 519.85

POLIANE CRISTINA DE FARIAS

**PROBLEMA DE CORTE DE ESTOQUE UNIDIMENSIONAL COM
REAPROVEITAMENTO DE SOBRAS:
ABORDAGEM DE RESOLUÇÃO POR MEIO DE UMA TÉCNICA DE
GERAÇÃO DE COLUNAS**

Dissertação de mestrado apresentada ao Departamento de Matemática da Universidade Estadual de Londrina, como requisito parcial para a obtenção do Título de MESTRE em Matemática Aplicada e Computacional.

BANCA EXAMINADORA

Prof. Dr. Robinson Samuel Vieira Hoto
UEL – Londrina – PR

Profa. Dra. Narefh Kumar Sharma
UEL – Londrina – PR

Profa. Dra. Maria Angélica de Oliveira Camargo
Brunetto
UEL – Londrina – PR

Londrina, 18 de abril de 2011.

AGRADECIMENTOS

Agradeço a Deus por estar sempre me auxiliando e abençoando as atividades que me proponho a realizar.

Ao meu esposo pelo amor, paciência e compreensão.

Aos meus amigos do PGMAC, Camila pela amizade e companhia ao NAFI, a Cibele, ao Geovani, a Renata, ao Rodrigo, Daniela e Vinícius pelas distrações nos momentos tensos e pelo compartilhamento de conhecimento.

Ao Maurício pela sua cumplicidade, pelo compartilhamento de conhecimento e amizade nos momentos difíceis do curso e da vida.

Ao Everton por sempre estar disposto a ajudar e por me ensinar a vencer os desafios que os computadores insistem em nos lançar quando programamos.

Aos professores da UEL pelo conhecimento compartilhado.

Ao professor Robinson, meu orientador, pela paciência à minha conquista de maturidade acadêmica.

À minha família por acreditarem no meu potencial.

Agradeço especialmente a minha mãe que com muito amor e orações transmitiu-me força e perseverança.

FARIAS, Poliane Cristina de. **Problema de corte de estoque unidimensional com reaproveitamento com reaproveitamento de sobras**: abordagem de resolução por meio de uma técnica de geração de colunas. 2011. 108 f. Dissertação (Mestrado em Matemática Aplicada e Computacional) – Universidade Estadual de Londrina, Londrina, 2011.

RESUMO

O Problema de Corte de Estoque Unidimensional é um problema de Programação Linear Inteira que consiste em cortar peças menores de objetos em estoque, utilizando o material de forma eficiente, minimizando uma determinada função objetivo. Este tipo de problema de otimização tem uma grande aplicabilidade prática em inúmeros segmentos industriais, sendo de grande importância no planejamento dos sistemas produtivos. Neste contexto, o Problema de Corte de Estoque com Reaproveitamento de Sobras (PCES) enfoca esta questão visando a possibilidade de reutilização de material resultante do processo de corte, reduzindo o descarte de matéria-prima. Diversos modelos matemáticos, bem como diversas técnicas de resolução para este tipo de problema, tem sido estudados. Este trabalho apresenta algumas abordagens de resolução para o PCES utilizando o Método Simplex com Geração de Colunas. São propostos modelos matemáticos que visam dividir a geração de colunas (padrões de corte) em dois tipos de subproblemas: geração de padrões com perdas, em que os retalhos resultantes do processo são descartados, e geração de padrões com sobras, nos quais os retalhos podem ser reaproveitados futuramente. Busca-se minimizar o custo total de produção, que envolve custos relativos ao processo de corte, perda e sobra de material. Os modelos propostos foram implementados utilizando o software XPRESS-MP, em linguagem Mosel, e foram realizadas simulações, cujos resultados são apresentados e comparados com alguns trabalhos existentes na literatura.

Palavras-chave: Problema de Corte de Estoque Unidimensional. Reaproveitamento de Sobras. Geração de Coluna. Método Simplex. Branch and Bound.

FARIAS, Poliane Cristina de. **Problema de corte de estoque unidimensional com reaproveitamento com reaproveitamento de sobras**: abordagem de resolução por meio de uma técnica de geração de colunas. 2011. 108 f. Dissertação (Mestrado em Matemática Aplicada e Computacional) – Universidade Estadual de Londrina, Londrina, 2011.

ABSTRACT

One-Dimensional Cutting Stock Problem is a Linear Programming problem which consists in cutting objects in stock into smaller parts, using the material efficiently and minimizing a certain objective function. This type of optimization problem has practical application in several industrial areas and is too important in planning the production systems. In this context, the Cutting Stock Problem with Usable Leftovers (PCES) focuses on the reuse of the material resulting from the cutting process as such reducing the raw material discard. Several mathematical models and techniques for solving this type of problem have been studied. This work presents some approaches for solving the PCES using the Simplex Method with Column Generation. Mathematical models proposed aim to divide the generation of columns (cutting patterns) into two types of subproblems: the generation of patterns with waste, in which the residual material from the process is discarded, and generation of patterns with leftovers, in which the residual material could be reused later. The goal is to minimize the production cost including those of the cutting process, waste and leftovers. The proposed models were implemented with the XPRESS-MP software, using the Mosel language, and simulations were performed. The results are presented and compared with some papers in the literature.

Keywords: One-Dimensional Cutting Stock Problem. Usable Leftovers. Column Generation. Simplex Method. Branch-and-Bound.

LISTA DE FIGURAS

Figura 5.1 – Fluxograma da abordagem M1P&R	61
Figura 5.2 – Fluxograma da abordagem M2P&R	62
Figura 5.3 – Fluxograma da abordagem M3P&R	63
Figura 5.4 – Fluxograma da abordagem M4P&R	64

LISTA DE TABELAS

Tabela 6.1	– Solução do Exemplo sem limitação.	65
Tabela 6.2	– Solução do Exemplo 1	66
Tabela 6.3	– Solução do Exemplo 2	67
Tabela 6.4	– Solução do Exemplo 3	68
Tabela 6.5	– Solução do Exemplo 4	69
Tabela 6.6	– Solução do Exemplo 5	70
Tabela 6.7	– Solução do Exemplo 6	71
Tabela 6.8	– Solução do Exemplo 7	72
Tabela B.1	– Estoque 1 (sem limitação).....	81
Tabela B.2	– Dados do Exemplo sem limitação.....	81
Tabela B.3	– Estoque do Exemplo 1	82
Tabela B.4	– Dados do Exemplo 1	82
Tabela B.5	– Estoque do Exemplo 2	82
Tabela B.6	– Dados do Exemplo 2	82
Tabela B.7	– Estoque do Exemplo 3	83
Tabela B.8	– Dados do Exemplo 3	83
Tabela B.9	– Estoque do Exemplo 4	83
Tabela B.10	– Dados do Exemplo 4	84
Tabela B.11	– Estoque do Exemplo 5	84
Tabela B.12	– Dados do Exemplo 5	85
Tabela B.13	– Estoque do Exemplo 6	86
Tabela B.14	– Dados do Exemplo 6	86
Tabela B.15	– Estoque do Exemplo 7	86
Tabela B.16	– Dados do Exemplo 7	86
Tabela E.1	– Estoque do Exemplo 4	101
Tabela E.2	– Dados do Exemplo 4	101
Tabela E.3	– Solução do Exemplo 4	102

LISTA DE SÍMBOLOS E NOTAÇÕES

Conjunto de índices:

I : conjunto de índices de tipos de itens ($I=\{1, \dots, m\}$)

J : conjunto de índices de tipos de objetos ($J=\{1, \dots, M\}$)

K : conjunto de índices de tipos de compartimentos

\mathfrak{N} : conjunto de índices das colunas da matriz N

$P(j)$: conjunto de índices de padrões possíveis para o objeto de índice j

$P(j)^\alpha$: conjunto de índices para os padrões com perdas

$P(j)^\beta$: conjunto de índices para os padrões com sobras

R : conjunto de índices de tipos de intervalos que definem comprimentos de retalhos

Constantes:

α_{ijp} : quantidade de vezes que o item de índice i é cortado no j -ésimo objeto no p -ésimo padrão de corte, $p \in P(j)^\alpha$

α_{jp} : representa um padrão com perda se $\text{retalho} \leq S$

β_{ijq} : quantidade de vezes que o item de índice i é cortado no j -ésimo objeto no q -ésimo padrão de corte, $q \in P(j)^\beta$

β_{jp} : representa um padrão com sobra se $\text{retalho} \geq S$

π : vetor multiplicador simplex

π_i : componente do vetor multiplicador simplex

a_{ijp} : número de itens de índice i no padrão de corte p no j -ésimo objeto

a_{jp} : padrão de corte de índice p para o j -ésimo objeto

A : matriz de restrições

B : matriz básica

c : vetor de custos

c^{comp} : valor (custo) associado à movimentação de compartimentos

c_{jp} : valor (custo) associado ao objeto de índice j no p -ésimo padrão

c_k : custo associado a variável x_k da estratégia simplex

c^{perda} : valor (custo) associado à perda

$c^{processo}$: valor (custo) associado ao processo de corte

$c^{retalho}$: valor (custo) associado ao retalho

c^{sobra} : valor (custo) associado à sobra

d : vetor de demanda

d_i : demanda do item l_i

l_i : comprimento do item de índice i

L_j : comprimento do objeto de índice j

m : número de tipos de itens

M : número de tipos de objetos em estoque

N : matriz não básica

r_k : comprimentos da sobra de índice k

S : comprimento mínimo aceito como sobra

s_{jk} : quantidade disponível do j -ésimo objeto no compartimento móvel de índice k

S_{jq} : perda no objeto de índice j no padrão de índice p

t_{jp} : comprimento do retalho no p -ésimo padrão de corte aplicado no objeto de índice j

T_{jp} : perda no objeto de índice j no padrão de índice p

v_j : valor associado ao objeto de índice j

x_B^0 : vetor de variáveis básicas inicial

x_N^0 : vetor de variáveis não básicas inicial

w_k : valor associado à sobra de índice k

Variáveis:

x : vetor de variáveis

x_B : vetor das variáveis básicas

x_{jp}^α : número de objetos de comprimento L_j cortados segundo o p -ésimo padrão de corte, $p \in P(j)^\alpha$, produzindo perdas

x_{jq}^β : número de objetos de comprimento L_j cortados segundo o q -ésimo padrão de corte, $q \in P(j)^\beta$, produzindo sobras

x_{jp} : número de objetos de índice j cortados conforme o padrão de corte p

x_p : número de objetos cortados no p -ésimo padrão

x_N : vetor das variáveis não básicas

Parâmetros:

L_{perda_j} : comprimento máximo de retalho para perda aceitável para o objeto de índice j

L_{sobra_j} : comprimento mínimo de retalho aceito como sobra para o objeto de comprimento L_j

$L_{sobra_j}^{max}$: comprimento máximo de retalho aceito como sobra para o objeto de comprimento L_j

$qntObj_j$: quantidade de objetos de comprimento L_j disponíveis em estoque

LISTA DE ACRÔNIMOS

PCE	Problema de Corte de Estoque Unidimensional
PCEC	Problema de Corte Unidimensional Clássico
PCER	Problema de Corte de Estoque Unidimensional Restrito
PCES	Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras
PLI	Programação Linear Inteira
PM	Problema Mestre
PMR	Problema Mestre Restrito
PMRR	Problema Mestre Restrito Relaxado

SUMÁRIO

CAPÍTULO 1 – INTRODUÇÃO	13
CAPÍTULO 2 – PROBLEMA DE CORTE DE ESTOQUE UNIDIMENSIONAL (PCE)	16
2.1 FORMULAÇÃO MATEMÁTICA DO PCE	17
2.2 O MÉTODO SIMPLEX APLICADO AO PCE	20
2.2.1 Descrição do método	20
2.3 A TÉCNICA DE GERAÇÃO DE COLUNAS	23
2.3.1 Descrição da técnica	23
2.4 MÉTODO SIMPLEX COM GERAÇÃO DE COLUNAS	24
2.5 O MÉTODO BRANCH-AND-BOUND	24
CAPÍTULO 3 – PCE COM REAPROVEITAMENTO DE SOBRAS (PCES)	26
3.1 MODELO PROPOSTO POR SCHEITHAUER <i>ET AL.</i> (1991)	31
3.2 APLICAÇÃO EM UMA INDÚSTRIA DE PROCESSAMENTO DE MADEIRA APRESENTADA POR KOCH <i>ET AL.</i> (2008)	34
3.3 ESTRATÉGIAS DE RESOLUÇÃO POR MEIO DE DOIS OBJETIVOS PROPOSTAS POR PINTO (2008)	38
CAPÍTULO 4 – ESTRATÉGIAS DE RESOLUÇÃO PARA O PCES	49
4.1 ABORDAGEM M1P&R	50
4.1.1 Modelagem Matemática Proposta I	50
4.2 ABORDAGEM M2P&R	52
4.2.1 Modelagem Matemática Proposta II	52
4.3 ABORDAGEM M3P&R	53
4.3.1 Modelagem Matemática Proposta III	54
4.4 ABORDAGEM M4P&R	56
4.4.1 Modelagem Matemática Proposta IV	56
CAPÍTULO 5 – SIMULAÇÕES	58
5.1 ALGORITMOS IMPLEMENTADOS	61

CAPÍTULO 6 – RESULTADOS	65
6.1 PROBLEMA SEM LIMITAÇÃO DE ESTOQUE	65
6.2 PROBLEMAS COM LIMITAÇÃO DE ESTOQUE.	66
6.2.1 Exemplo 1.....	66
6.2.2 Exemplo 2.....	67
6.2.3 Exemplo 3.....	68
6.2.4 Exemplo 4.....	69
6.2.5 Exemplo 5.....	70
6.2.6 Exemplo 6.....	71
6.2.7 Exemplo 7.....	71
CONSIDERAÇÕES FINAIS	73
REFERÊNCIAS	76
APÊNDICES	79
APÊNDICE A – Demonstração de equivalência entre minimizar perda e objetos.....	80
APÊNDICE B – Exemplos	81
APÊNDICE C – Problema da Mochila e método de resolução	87
APÊNDICE D – Obtenção das mochilas para as abordagens propostas.....	84
APÊNDICE E – Artigo submetido à TEMA	93
APÊNDICE F – Trabalho submetido à XII DNPCE.....	106

CAPÍTULO 1

INTRODUÇÃO

As indústrias buscam melhorar seus processos produtivos com o objetivo de aumentar a lucratividade e a competitividade. O bom aproveitamento da matéria-prima pode reduzir de forma significativa o custo da produção, o que pode representar uma vantagem decisiva na competição com outras indústrias do setor. A busca por tornar esses processos mais eficientes, juntamente com os avanços computacionais, tem estimulado pesquisas acadêmicas sobre modelos de otimização para controle e planejamento de sistemas produtivos, e tem feito muitos industriais apreciarem a importância prática dos matemáticos.

Em diversos processos industriais, peças menores (itens) solicitadas são produzidas a partir do corte de peças maiores (objetos) em estoque. Itens devem ser cortados de objetos com a finalidade de atender as solicitações de clientes (demandas). Estes processos de corte, em geral, geram perdas indesejáveis de material. Surge então um problema de otimização, que consiste em cortar os objetos para a produção dos itens nas quantidades solicitadas, otimizando uma determinada função objetivo, de modo que a utilização de material seja a mais eficiente possível. Essa função objetivo pode ser: minimizar as perdas referentes ao processo de corte; minimizar o número de objetos cortados; minimizar o custo total do processo; entre outras [8].

Os objetos podem ter uma ou mais dimensões relevantes para o problema de corte: uma dimensão, por exemplo, no processo de corte de bobinas de aço; duas dimensões, nos processos de cortes em indústrias de placas de madeira; três dimensões, em processos com blocos de matéria-prima para colchões [2]; ou ainda mais dimensões [27]. No caso de um problema unidimensional, denominado Problema de Corte de Estoque Unidimensional (PCE), apenas o comprimento dos objetos é relevante.

Dados os comprimentos dos itens encomendados, pode-se ter diversas maneiras de cortar os objetos em estoque. Essas maneiras são denominadas padrões de corte. O que é produzido nesses padrões de corte e que não são itens, são denominados retalhos. Quando estes possuem um comprimento com condições de reutilização, são chamados de sobras, caso contrário, são chamados de perdas. Um melhor aproveitamento de material pode ocorrer na escolha de não se descartar todos os retalhos produzidos, mas sim analisar a possibilidade de reuso, isto é, descartar apenas as perdas e reutilizar as sobras para cortes futuros. Problemas deste tipo são conhecidos na literatura como Problemas de Corte de Estoque Unidimensional com Reaproveitamento de Sobras (PCES). O problema de corte é

classificado como um problema de Programação Linear Inteira (PLI), pois as variáveis de decisão do problema são inteiras e positivas. Este problema é aparentemente simples, e tem grande aplicabilidade prática, porém, é um problema NP-difícil¹. Nem todos os problemas NP-difíceis são igualmente difíceis do ponto de vista prático. Alguns deles podem ser resolvidos através de técnicas exatas, algoritmos aproximativos, programação dinâmica e outros métodos: Branch-and-Cut, Branch-and-Bound e heurísticas [26]. As técnicas Branch-and-Cut e Branch-and-Bound tornam-se inviáveis quando o problema envolve várias dezenas de itens a serem produzidos, pois demandam um alto tempo computacional e as heurísticas são métodos sem garantia formal de desempenho.

É importante observar que a evolução de *software* otimizador teve um papel fundamental no progresso da Otimização Linear. Um exemplo disto é o pacote XPRESS-MP, desenvolvido pela *Dash Optimization* (posteriormente adquirida pela FICO_{TM}²), e do CPLEX, comercializado desde 1988 pela *CPLEX Optimization Inc.* e atualmente desenvolvido pela *IBM World Trade Corporation*.

Devido ao desenvolvimento da computação e de algoritmos especializados, a possibilidade de resolver problemas relativamente grandes em pequena quantidade de tempo vem tornando a Otimização Linear Inteira muito mais comum. Para lidar com o número exponencialmente grande de variáveis, pode-se então recorrer ao Método Simplex com Geração de Colunas, o qual pode ser aplicado eficientemente na resolução de Problemas de Programação Linear e de Programação Inteira. A Técnica de Geração de Colunas corresponde a resolução de um subproblema que absorve parte da natureza combinatória do problema original.

Em geral, a resolução de um PCE por Simplex com Geração de Colunas é um processo iterativo baseado em um modelo matemático, em que um subproblema gerador de padrões de corte é resolvido. Neste trabalho são apresentadas duas modelagens para o PCES, as quais consideram objetos de estoque com comprimentos diversos, e utilizam como abordagem de resolução uma técnica de geração de colunas. Em cada uma das modelagens, são resolvidos dois tipos de subproblemas: um gerador de padrões com perdas aceitáveis (perdas que serão permitidas na solução), e um gerador de padrões com sobras (em intervalos pré-definidos). Na primeira modelagem, são solucionados dois conjuntos de subproblemas (com perda e com sobra para cada um dos objetos distintos em estoque). Na segunda, existem apenas dois geradores, de forma que o tipo de objeto faz parte da solução. Em ambas as

¹ Polinomial Não Determinístico Difícil, [26].

² Empresa que atua no gerenciamento de decisões.

modelagens busca-se minimizar o custo total de produção, que envolve custo de processo de corte, custo de perda ou sobra de material.

No Capítulo 2, é apresentado o Problema de Corte de Estoque Unidimensional. Serão discutidas algumas metodologias de resolução, com destaque ao Método Simplex com Geração de Colunas e também ao Método Branch-and-Bound. O Capítulo 3 trata do Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras. São discutidos alguns trabalhos existentes na literatura, considerando abordagens com e sem limitações de estoque, que utilizam modelos matemáticos baseados no PCE clássico com a proposta de investigar o manuseio dos retalhos. No Capítulo 4, são propostas algumas abordagens de resolução para o PCES. O Capítulo 5 trata das simulações realizadas, e os resultados obtidos são discutidos no Capítulo 6. Por fim, são apresentadas as conclusões no Capítulo 7, no qual são discutidas as contribuições deste trabalho e algumas sugestões para trabalhos futuros.

CAPÍTULO 2

PROBLEMA DE CORTE DE ESTOQUE UNIDIMENSIONAL (PCE)

Neste capítulo será feita uma discussão a respeito do Problema de Corte de Estoque Unidimensional e as maneiras de resolvê-lo, de forma a permitir uma melhor compreensão acerca do objetivo da pesquisa: a resolução de um Problema de Corte de Estoque com Reaproveitamento de Sobras por meio do Método Simplex com Geração de Colunas, que será apresentado nos capítulos seguintes.

O Problema de Corte é classificado como um problema de Programação Linear Inteira (PLI), pois as variáveis de decisão do problema são inteiras e positivas. Os principais modelos matemáticos para este problema surgiram na década de 1940, mas as principais pesquisas surgiram na década de 1960. A primeira documentação que trata do problema é devida ao economista russo Leonid Vitaliyevich Kantorovich [19], em que apresenta modelos matemáticos de programação linear para o planejamento e organização da produção. O PCE foi um dos problemas abordados em seu trabalho, escrito em russo em 1939, mas traduzido para o inglês e publicado apenas em 1960 por causa da Guerra Fria.

O estudo da programação linear tem raízes nos trabalhos de Fourier sobre inequações lineares, publicados em 1826, mas a programação linear ganhou grande impulso quando, durante a Segunda Guerra Mundial, um grupo formado por Dantzig, Wood e seus associados no Departamento da Força Aérea dos Estados Unidos, foi convocado para pesquisar a viabilidade de aplicar a Matemática e técnicas relacionadas para resolver problemas de planejamento e programação militar [24]. Em 1947, Dantzig divulgou a ferramenta desenvolvida pelo grupo: o Método Simplex. Em pouco tempo a programação linear foi aplicada a diversas áreas do conhecimento e os avanços computacionais também contribuíram para o aumento na utilização dessa ferramenta, permitindo resolver problemas de escalas maiores. Além disso, problemas aparentemente não relacionados com produção e planejamento podem ser expressos em termos de programação linear, e resolvidos pelo Método Simplex.

Devido a importância e aplicabilidade dos Problemas de Corte de Estoque nas indústrias, muitos pesquisadores em todo o mundo tem concentrado esforços no desenvolvimento de métodos para sua resolução [25]. As primeiras tentativas consistiam em relaxar as condições de integralidade das variáveis e resolver o problema aplicando o Método Simplex. Obtido o resultado, a solução era truncada para o valor inteiro mais próximo e ajustes pós-otimização eram feitos, se necessários, para atender o restante da demanda.

As modelagens matemáticas e os métodos de resolução de maior repercussão na literatura foram publicados por Gilmore e Gomory [11], [12] e [13], que propuseram uma técnica de geração de colunas para obtenção de uma solução ótima contínua (isto é, modelo com a condição de integralidade relaxada). Esta técnica é utilizada para solucionar problemas lineares de grande escala. Métodos heurísticos também foram desenvolvidos para aumentar a eficiência do processo de arredondamento de variáveis [32], [36].

Um trabalho que merece destaque, sobre Geração de Colunas em Otimização Inteira, encontra-se em Barnhart *et al.* [5]. Nesse trabalho, os autores procuraram unificar os vários trabalhos da literatura, apresentando uma estrutura geral para a geração de colunas e dando ideias gerais sobre a sua utilização em vários contextos. Carvalho [6] apresentou uma abordagem diferente para o problema geral de corte de estoque, formulado como fluxo em redes.

Na próxima seção, será apresentada a formulação clássica para o PCE, e algumas metodologias utilizadas para sua resolução.

2.1 FORMULAÇÃO MATEMÁTICA DO PCE

O PCE consiste em cortar objetos de comprimento $L_j, j \in J = \{1, \dots, M\}$ para produzir m diferentes tipos de itens de comprimento $l_i, i \in I = \{1, \dots, m\}, l_i \leq L_j$, em quantidades $d_i, i \in I$, conhecidas, otimizando uma determinada função objetivo que pode ser: minimizar as perdas referentes ao processo de corte; o número de objetos cortados que atende a demanda; o custo dos objetos cortados; etc. Supõe-se que os objetos L_j são suficientes para a produção, isto é, para atender toda a demanda. Conforme os tamanhos dos itens encomendados, pode-se ter diversas maneiras possíveis de cortar os objetos em estoque, denominadas padrões de corte. O conjunto de índices do número de padrões de corte possíveis para o objeto de índice j será denotado por $P(j)$. A cada padrão de corte para o objeto de índice j está associado um vetor m -dimensional, $a_{jp} = (a_{1jp}, \dots, a_{mjp})$, $p \in P(j)$ e a_{ijp} denotará o número de itens do tipo i no padrão de corte p no j -ésimo objeto, que será um padrão de corte se, e somente se o sistema

$$\sum_{i \in I} l_i a_{ijp} \leq L_j, \quad a_{ijp} \geq 0 \text{ e inteiro, } j \in J$$

é satisfeito.

Existe um número finito de padrões de corte. Porém, na prática, este número é muito grande. O padrão de corte que produz apenas um só tipo de item é chamado padrão de corte homogêneo, isto é, o vetor associado tem apenas uma coordenada não-nula: $(0, \dots, a_{ijp}, \dots, 0)^T$, $a_{ijp} \neq 0$. Um padrão de corte homogêneo é chamado maximal quando a_{ijp} é o número máximo possível do item de índice i cortado no padrão p . Note que sempre teremos m padrões homogêneos maximais, cujos vetores associados definem uma matriz diagonal. Um padrão de corte heterogêneo, por sua vez, possui mais de uma coordenada não nula, e é maximal quando o máximo possível do objeto de estoque é cortado. Uma vez definido os padrões de corte, o problema consiste em determinar quantas vezes esses padrões serão repetidos que satisfaça algum critério a ser otimizado. Define-se a variável x_{jp} como o número de objetos de índice j cortados conforme o padrão de corte p . Então o problema de corte pode ser formulado como:

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (2.1)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} = d_i, \quad i \in I \quad (2.2)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, p \in P(j) \quad (2.3)$$

A função objetivo visa minimizar o custo de cortar os objetos. A restrição 2.2 garante que a demanda seja satisfeita, e a restrição 2.3 garante que a frequência com que o padrão de corte p é repetido seja um número não negativo. Além disso, esse modelo exige que as variáveis x_{jp} sejam inteiras o que o torna difícil de ser resolvido computacionalmente. Em notação matricial o modelo pode ser escrito da seguinte maneira:

$$\text{minimizar} \quad c^T x \quad (2.4)$$

$$\text{sujeito a} \quad Ax = d \quad (2.5)$$

$$x \geq 0 \text{ e inteiro} \quad (2.6)$$

em que A é uma matriz $m \times n$ de números inteiros cujas colunas representam os padrões de corte. É comum os problemas de corte possuírem um elevado número de padrões o que torna inviável a determinação de todos eles a priori, como suposto no modelo anterior.

Na prática, uma perda é definida como sendo um retalho que não existe a possibilidade de reuso para um corte futuro, ao contrário da sobra que pode ser reutilizada. Um objetivo interessante a ser otimizado seria a minimização da perda e o surgimento de

sobras. Pode-se ver, por exemplo, em Cherri e Arenales (2005). A função objetivo para o problema de minimizar a perda é tal que:

$$c_{jp} = L_j - \sum_{i=1}^m a_{ijp} l_i$$

em que c_{jp} define a perda no padrão de corte p no objeto de índice j , $p \in P(j)$. Note que minimizar a perda total é equivalente a minimizar o total de objetos cortados ($c_{jp} = 1, \forall j \in J, p \in P(j)$). A demonstração pode ser encontrada no Apêndice A.

Geralmente, os métodos de resolução para Programação Linear Inteira (PLI) estão fundamentados na obtenção de sucessivas soluções de subproblemas, fornecendo valores limitantes cada vez mais próximos da solução ótima do problema de PLI original. Os métodos de enumeração implícita como Balas e Branch-and-Bound, obtêm limitantes para o valor ótimo através de uma enumeração das soluções viáveis do problema, no sentido de que a análise de certas propriedades de cada subproblema evita a enumeração exaustiva (explícita) de todas as soluções inteiras viáveis, o que é geralmente impraticável. O algoritmo de Balas é utilizado para problemas de PLI em que cada uma de suas variáveis assumem apenas os valores 0 ou 1, diferentemente do método Branch-and-Bound que pode ser aplicado em qualquer problema de PLI. Também os métodos de planos de corte como, por exemplo, o Método de Plano de Corte de Gomory, obtêm limitantes para o valor ótimo através da redução do tamanho do conjunto de soluções viáveis de relaxações dos subproblemas, acrescentando restrições válidas (cortes) às restrições originais do problema. Algumas destas técnicas tornam-se inviáveis quando o problema envolve várias dezenas de itens a serem produzidos, pois demandam um alto tempo computacional. Assim, para lidar com o número exponencialmente grande de variáveis, pode-se então relaxar as condições de integralidade das variáveis e recorrer ao Método Simplex com Geração de Colunas. Partindo da solução obtida com o Método de Geração de Colunas, uma maneira de se obter uma solução viável inteira para o problema é através de heurísticas de arredondamento. Se esta solução inteira ainda for inviável para o problema inteiro é escrito e resolvido um outro problema com baixa demanda, chamado de Problema Residual, que Poldi [28] mostrou que também pode ser resolvido eficientemente pelo mesmo modelo linear.

Em uma das modelagens apresentadas neste trabalho, é utilizado o Método Simplex com Geração de Colunas para resolver o PCES proposto com as condições de integralidade das variáveis relaxadas e, para obtenção da solução inteira, é aplicado o método Branch-and-Bound. Na outra modelagem, a obtenção da solução inteira é feita truncando a

solução contínua para o inteiro mais próximo. Se a solução apresentada for inviável ao problema é resolvido um Problema Residual.

2.2 O MÉTODO SIMPLEX APLICADO AO PCE

As dificuldades computacionais mencionadas na resolução do PCE tem sido contornada com a relaxação linear da condição de integralidade sobre as variáveis x_{jp} e a posterior utilização do Método Simplex com Geração de Colunas, que fornece uma solução ótima e contínua para o problema. Este método de solução foi estudado por Gilmore e Gomory [11] e [12], e consiste basicamente em substituir um dos padrões de corte básicos (uma determinada coluna a_{jp}) por outro padrão (determinado pela resolução de um subproblema) que melhore a solução básica atual. Gilmore e Gomory salientam que a solução contínua do problema relaxado é uma boa solução aproximada para o problema original, quando a demanda é suficientemente grande e exige uma alta repetição dos padrões de corte. Em busca da solução inteira heurísticas de arredondamento são aplicadas às soluções contínuas encontradas no problema relaxado. Mais detalhes podem ser encontrados em Stadtler [34], Wascher e Gau [36], entre outros.

2.2.1 Descrição do Método

Considere o seguinte problema de Programação Linear:

$$\begin{array}{ll} \text{minimizar} & c^T x \\ \text{sujeito a} & Ax = d \\ & x \geq 0 \end{array}$$

em que $A \in \mathbb{R}^{m \times n}$ com $\text{Posto}(A) = m$ de números inteiros cujas colunas representam os padrões de corte e d é um vetor m -dimensional.

A solução geral do sistema pode ser descrita da seguinte maneira. Considere uma partição das colunas da matriz A :

$$A = (B, N)$$

onde $B \in \mathbb{R}^{m \times m}$ não-singular e $N \in \mathbb{R}^{m \times (n-m)}$. A matriz B é chamada de matriz básica e a matriz N , não-básica. A mesma partição é feita no vetor das variáveis $x = (x_B, x_N)^T$, onde x_B é chamado vetor das variáveis básicas e x_N vetor das variáveis não-básicas.

Assim, o sistema $Ax = d$ pode ser escrito da seguinte forma:

$$Ax = d \Leftrightarrow Bx_B + Nx_N = d \Leftrightarrow x_B = B^{-1}d - B^{-1}Nx_N$$

A solução particular de x obtida por: $x_B^0 = B^{-1}d$ e $x_N^0 = 0$, é chamada solução básica. Se $x_B^0 = B^{-1}d \geq 0$, então a solução básica é primal-factível e a partição básica também é dita primal-factível.

Considere também a partição básica nos coeficientes da função objetivo $c^T = (c_B, c_N)^T$. O vetor $\pi \in \mathbb{R}^m$, dado por $\pi^T = c_B^T B^{-1}$ é chamado de vetor multiplicador simplex. A função objetivo em termos das componentes das variáveis não-básicas pode ser escrita da seguinte maneira:

$$\begin{aligned} c^T x &= c_B^T x_B + c_N^T x_N \\ &= c_B^T (B^{-1}d - B^{-1}N x_N) + c_N^T x_N \\ &= c_B^T B^{-1}d - c_B^T B^{-1}N x_N + c_N^T x_N \\ &= c_B^T x_B^0 + \sum_{p \in \mathfrak{N}} (c_p - \pi^T a_p) x_j \end{aligned} \quad (2.7)$$

onde $a_p \in \mathfrak{N}$ corresponde à coluna p de N e \mathfrak{N} é o conjunto dos índices das colunas da matriz N . Os coeficientes das variáveis não-básicas x_p : $c_p - \pi^T a_p$ são chamados custos reduzidos ou custos relativos.

Estratégia Simplex é o nome dado à seguinte perturbação da solução básica inicial, faz-se:

1. $x_k = \epsilon \geq 0$ para algum $k \in \mathfrak{N}$, tal que $c_k - \pi^T a_k < 0$;
2. $x_p = 0$ para todo $p \neq k$, $p \in \mathfrak{N}$.

Observe que diminuir o valor da função objetivo, com $x \geq 0$, implica em escolher, necessariamente, uma componente x_k de forma que o custo relativo desta seja estritamente menor que zero.

A estratégia simplex produz uma nova solução dada por :

$$\begin{cases} x_B &= x_B^0 + \epsilon y \\ x_N &= \epsilon e_k \end{cases} \quad (2.8)$$

e o valor da função objetivo é dado por:

$$c_B^T x_B^0 + (c_k - \pi^T a_k) \epsilon, \quad (2.9)$$

em $y = -B^{-1}a_k$ e $e_k = (0, \dots, 1, \dots, 0)^T \in \mathbb{R}^{n-m}$, com 1 na k -ésima componente.

Note que a direção $dir \in R^n$, dada por $dir = (dir_B, dir_N)^T = (y, e_k)^T$, define uma perturbação da solução básica e é chamada direção simplex. Se a solução for não degenerada, ou seja $x_B^0 > 0$, de (2.8) e (2.9) segue que diré uma direção factível. Note ainda, que o produto escalar da direção dir com o gradiente da função objetivo é $c^T dir = c_k - \pi^T a_k$ e se o índice k for escolhido tal que $c^T dir < 0$, então a partir de (2.9) pode-se dizer que a direção é de descida.

De (2.8), pode-se determinar o maior valor do passo ϵ , impondo $x_B \geq 0$:

$$\epsilon^0 = -\frac{x_{B_r}^0}{y_r} = \text{Mínimo}\left\{-\frac{x_{B_i}^0}{y_i}, y_i < 0, i = 0, \dots, m\right\}$$

onde $x_{B_i}^0$ é a i -ésima componente de x_B^0 .

Para esta escolha de ϵ^0 , a r -ésima componente de x_B^0 se anula ($x_{B_r} = x_{B_r}^0 - \frac{x_{B_r}^0}{y_r} y_r = 0$), enquanto apenas uma variável de x_N torna-se positiva ($x_k = \epsilon$) (a menos de degeneração: $\epsilon^0 = 0$). Isto sugere uma nova partição básica $A = (B', N')$ em que a r -ésima coluna de B é substituída por $a_k \in N$. Esta nova partição é básica factível e a sua solução associada é dada por: $x^1 = x^0 + \epsilon^0 dir$.

Pode-se escolher qualquer k tal que $(c_k - \pi^T a_k) < 0$ para que uma nova solução básica factível seja gerada com um valor menor para a função objetivo. Um dos critérios mais utilizados para escolher o índice k , é o critério da Máxima Descida de Dantzig [18]. O critério de Dantzig escolhe o índice k tal que

$$c_k - \pi^T a_k = \min\{(c_p - \pi^T a_p), p \in \mathfrak{N}\}.$$

A solução ótima é obtida quando todos os custos reduzidos forem não negativos.

No método Simplex, a cada iteração são percorridas as colunas correspondentes as variáveis não-básicas uma a uma para determinar a variável de entrada à base. A Técnica de Geração de Colunas oferece uma alternativa mais eficiente. Mostrou-se uma abordagem bem sucedida para resolver problemas de PLI.

2.3 A TÉCNICA DE GERAÇÃO DE COLUNAS

Enquanto que a relaxação linear do modelo (2.1) supera a dificuldade inerente de encontrar uma solução para o programa inteiro, o procedimento de geração de colunas supre a dificuldade de armazenar todas as colunas (padrões de corte) da matriz A (2.4) a priori e testar, a cada iteração, todas as colunas não básicas para determinar aquela que apresenta o menor custo relativo.

Baseado na decomposição de Dantzig-Wolfe, o Método de Geração de Colunas é uma das ferramentas mais importantes para se obter soluções aproximadas de problemas de PLI [23].

2.3.1 Descrição da Técnica

O procedimento de geração de colunas consiste em gerar uma coluna k , isto é, um novo padrão de corte, com a utilização do critério de Dantzig, que procura a variável x_k com o menor custo relativo. A geração da coluna é feita através de um problema auxiliar, chamado de subproblema ou problema *pricing out*, dado por:

$$\begin{array}{ll} \text{minimizar} & c_p - \pi^T a_p \\ \text{sujeito a} & a_p \text{ corresponde a um padrão de corte} \end{array}$$

Considerando que os objetos de estoque tem comprimento L , $a_p = (a_{1p}, \dots, a_{mp})$ e os valores de c_p , $p = 1, \dots, n$, são constantes, podemos omitir c_p da função objetivo e reescrever o subproblema como:

$$\begin{array}{ll} \text{maximizar} & \sum_{i=1}^m \pi_i a_{ip} \\ \text{sujeito a} & \sum_{i=1}^m l_i a_{ip} \leq L \\ & a_{ip} \geq 0 \text{ e inteiro} \end{array}$$

que é um Problema da Mochila irrestrito, isto é, não há um limitante superior para a quantidade máxima de a_{ip} . Mais informações sobre o Problema da Mochila podem ser encontradas no Apêndice C. A primeira restrição assegura que todos os itens cortados não ultrapassem o tamanho do objeto (restrição física), e a segunda restrição garante que os padrões de corte sejam constituídos por números inteiros e positivos.

Se o menor dos custos relativos for maior ou igual a zero, a solução do problema relaxado é ótima. Caso contrário, a coluna de menor custo relativo entra na base e as iterações continuam com os passos usuais do Método Simplex.

2.4 MÉTODO SIMPLEX COM GERAÇÃO DE COLUNAS

Não sendo necessário conhecer todas as colunas da matriz A a priori, o problema de PLI (2.1) é formulado sem dificuldade, chamado de Problema Mestre (PM). A partir do PM é escrito um Problema Mestre Restrito (PMR), que possui apenas um subconjunto de colunas do PM. Então a técnica consiste em:

1. Criar um PMR inicial e relaxar as condições de integralidade;
2. Resolver o PMR relaxado utilizando o Método Simplex;
3. Pelo fato do número das variáveis não básicas ser muito grande (pois na prática $n \gg m$) resolver um subproblema, em que se maximiza a contribuição da coluna para a redução do valor da função objetivo;
4. Verificar se existe alguma coluna que possa melhorar o valor atual da função objetivo, caso não exista, é feito alguns ajustes na solução encontrada para obtenção da solução inteira.

Apesar do Método de Geração de Colunas ser muito eficiente, este procedimento tem problemas de estabilização e pode ter uma convergência lenta para a solução ótima do problema relaxado. No início do processo o valor da função objetivo decresce rapidamente, mas à medida que a solução se aproxima da ótima, este decréscimo é cada vez menor. Isto ocorre por causa das degenerações primais e pelas excessivas oscilações das variáveis duais [3]. Este comportamento é típico na geração de colunas e é conhecido como *tail* (cauda) [4].

2.5 O MÉTODO BRANCH-AND-BOUND

O Método Branch-and-Bound é muito eficiente em resolver problemas de programação não linear e linear inteira mista. O método foi originalmente desenvolvido por A. H. Land e A. G. Doig [22] para resolver problemas de PLI e foi posteriormente modificado por R.J. Dakin.

Neste método o problema de PLI não é resolvido diretamente. Primeiramente resolve-se o problema com as condições de integralidade relaxadas. Se a

solução do problema contínuo for uma solução inteira, esta representa a solução ótima. Caso contrário, pelo menos uma das variáveis, por exemplo x_p , deve ser não inteira. Se x_p não é inteira, pode-se sempre encontrar um inteiro $[x_p]$ tal que

$$[x_p] < x_p < [x_p] + 1.$$

Então dois subproblemas são formulados, um com a restrição de limitante superior:

$$x_p \leq [x_p]$$

e o outro com a restrição de limitante inferior:

$$x_p \geq [x_p] + 1$$

O processo de encontrar estes subproblemas é chamado de *branching*.

Os dois subproblemas formulados são resolvidos como problemas contínuos. Pode-se observar que a solução de um problema contínuo forma um nó e de cada nó dois *branches* (ramos) podem surgir.

O processo de *branching* e de resolver uma sequência de problemas contínuos continua até uma solução factível inteira ser encontrada para um dos dois problemas. Quando tal solução é encontrada, o valor da função correspondente torna-se um limitante superior sobre o valor mínimo da função objetivo. Com isto pode-se eliminar todas as soluções contínuas seguintes (nós) cujo valor da função objetivo são maiores que o limitante superior. Os nós que são eliminados são denominados *fathomed* (podados) porque a partir deles não é possível encontrar uma solução inteira melhor. O valor do limitante superior sobre a função objetivo é substituída quando um limitante melhor é obtido.

Pode-se ver que um nó é podado se qualquer das condições seguintes são verdadeiras:

1. A solução contínua é uma solução factível inteira;
2. O problema não tem uma solução factível contínua;
3. O valor ótimo do problema contínuo é maior que o limitante superior atual.

O algoritmo continua a selecionar um nó para ramificações posteriores até todos os nós terem sido podados. O nó podado particularmente que tem a solução factível inteira com o menor valor objetivo dá a solução ótima para o problema de PLI.

CAPÍTULO 3

PCE COM REAPROVEITAMENTO DE SOBRAS (PCES)

Uma abordagem alternativa para o PCE surgiu na década de 80: o Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras (PCES). Tem sido estudada desde 1986 com os trabalhos de Roodman [30], Scheithauer [31], Sinuany Stern e Weiner [33], Gradisar *et al.* [14], [15] e [16], Chu e Antonio [9], Kos e Duhovnik [21], Gradisar e Trkman [17], Abuabara [1], Cherri [7], Trkman e Gradisar [35], Koch *et al.* [20]. Este problema consiste em cortar peças menores (itens) de objetos em estoque, disponíveis em quantidade suficiente para atender aos pedidos de clientes (demandas) e, diferentemente do PCE, obter sobras deste processo que possam ser reutilizadas futuramente, minimizando uma função objetivo. Neste trabalho será apresentado um PCES cuja função objetivo é minimizar o custo total de produção. Neste tipo de problema, busca-se obter soluções mais eficientes através de um novo critério de seleção de uma solução, a respeito dos retalhos resultantes do processo de corte. Para isto, o objetivo não está limitado a minimizar as perdas, ou equivalentemente o número de objetos cortados, mas sim em realizar um planejamento do processo de corte tendo em vista a possibilidade de produção de sobras que possam ser reutilizadas futuramente. Esta abordagem baseia-se na observação de que a busca pela perda mínima nem sempre é a melhor opção, uma vez que pode-se obter soluções que reduzam ou mesmo eliminem as perdas por meio de um remanejamento dos itens nos padrões de corte para geração de sobras. De forma geral, a produção de sobras, em detrimento das perdas, implica na utilização de uma maior quantidade de objetos. Entretanto, uma abordagem que prevê o reaproveitamento de sobras pode ser interessante por evitar a geração de perdas e o corte de itens excedentes, em problemas onde estes ocorrem, proporcionando a conversão destes em sobras, especialmente nos casos em que os tipos de itens demandados sejam variáveis ao longo do tempo.

Um dos primeiros trabalhos que trata deste tipo de problema foi escrito por Roodman (1986), em que é descrito um procedimento heurístico para resolver o mesmo PCE proposto por Gilmore e Gomory (1963), no qual os objetos em estoque possuem comprimentos diferentes e têm disponibilidades limitadas. O trabalho foi desenvolvido para uma indústria de corte de aço que tem em sua linha de produção o corte de alguns itens grandes. O objetivo principal é a minimização das perdas, entretanto existe um objetivo secundário que é a possibilidade de concentrar os retalhos em poucos padrões de corte para que possam ser utilizadas no corte de futuros itens. O procedimento heurístico desenvolvido

consiste de três fases. Na primeira fase, a solução inicial é obtida pela relaxação do modelo utilizando uma técnica modificada de geração de colunas seguida de um arredondamento, que resulta em soluções inteiras. A segunda fase do procedimento heurístico considera os itens em excesso como sobras e as reorganiza concentrando em poucos padrões de corte. A terceira fase consiste em fazer melhorias na solução obtida trocando padrões de corte que utilizam longos objetos por objetos de comprimentos menores ou vice-versa. Esta abordagem apresentou bons resultados, segundo o autor.

Com o objetivo de mostrar que se deve dar atenção à reutilização da matéria prima, Scheithauer *et al.*(1991), utilizando o modelo de corte de estoque unidimensional clássico, propôs investigar o manuseio de sobras empregando o Método Simplex e a Técnica de Geração de Colunas. Foram apresentados dois exemplos para ilustrar a estratégia desenvolvida. Estes exemplos apontam a dificuldade em considerar sobras e a melhora que pode haver na solução quando consideradas.

O trabalho de Sinuany-Stern e Weiner (1994) considera o problema de corte de estoque com dois objetivos a serem alcançados simultaneamente: um consiste em minimizar o retalho (principal) e o outro em acumular a máxima quantidade de retalhos no último objeto a ser cortado para que este fosse maior que um determinado limitante (comprimento do menor item demandado) e retornasse ao estoque como sobra para ser utilizada futuramente. O algoritmo proposto foi desenvolvido para uma pequena oficina de Kibbutz Samar em Israel que trabalha com o corte de objetos de metal. O estoque da oficina era composto por objetos de comprimento padrão e sobras originadas de cortes anteriores. Para resolver este problema, a estratégia desenvolvida foi aplicar o Método Branch-and-Bound, o qual iniciava com um limitante inferior simples para o número de objetos, isto é, o comprimento mínimo para atender demanda. Quando existia uma solução viável para o objetivo principal era resolvido o segundo objetivo, caso contrário, havia um aumento do número de objetos (um de cada vez) até obter uma solução viável. Os autores efetuaram simulações com alguns dados da indústria metalúrgica e concluíram que a abordagem foi bem sucedida.

Gradisar *et al.* (1997) apresentaram um estudo sobre o Problema de Corte de Estoque em indústrias de tecido, as quais tinham o estoque composto por objetos (rolos) de diversos comprimentos. Para resolver este problema os autores propuseram um modelo bi-objetivo para minimizar o número de itens que não são atendidos durante o processo de corte e a perda total (soma de perdas inferiores a um limitante). Embora tendo proposto este modelo, não o utilizaram para resolver o problema, talvez pela complexidade computacional

e, desta forma desenvolveram um procedimento heurístico, denominado COLA (COMputerized LAYing out), o qual pode ser considerado como um algoritmo guloso. Segundo os autores, os testes computacionais realizados com problemas gerados aleatoriamente mostraram um bom desempenho desse algoritmo.

Gradisar *et al.* (1999a) apresentaram um estudo sobre o PCE com diferentes situações que podem ocorrer na prática, dentre elas, o aproveitamento de sobras e o estoque composto por objetos de comprimentos distintos. Para resolver este problema, os autores não utilizaram o modelo proposto, mas desenvolveram um procedimento heurístico denominado CUT, o qual utiliza o Problema da Mochila (Gilmore e Gomory [12]). Segundo os autores, CUT é uma versão melhorada e generalizada do algoritmo COLA (Gradisar *et al.* [14]) e encontra soluções melhores em um tempo computacional muito baixo.

No trabalho de Gradisar *et al.* (1999b), é proposta uma abordagem para resolver um PCE, cujo o estoque é composto por poucos objetos com comprimentos distintos, sendo que somente uma pequena parte do estoque pode conter todos os comprimentos diferentes, que são sobras dos cortes anteriores. Esta abordagem, que combina programação linear e procedimento heurístico, é baseada em alterações no algoritmo COLA (Gradisar *et al.* [14]). Além de atender a demanda dos itens, a abordagem busca também acumular retalhos em um único objeto cortado para que possam ser utilizados no futuro. Neste trabalho, os autores apresentaram apenas um exemplo para ilustrar o desenvolvimento do método proposto e não fizeram qualquer comentário sobre possíveis testes realizados.

O trabalho desenvolvido por Chu e Antonio (1999) considera um PCE real em uma metalúrgica, cujos tubos metálicos podem ser circulares, quadrangulares, triangulares, entre outros. O objetivo é minimizar os custos com a perda, tempo de corte necessário para a produção dos itens (peças) e os possíveis sobras que podem ser geradas durante o processo de corte. Para resolver este problema, os autores consideram um PCE típico com algumas restrições técnicas que surgem na indústria de cortes de metal, como por exemplo a tolerância. Existem duas categorias para a tolerância: estrita e normal. Itens com tolerância estrita são entregues ao cliente com comprimento exato e em boa qualidade (sem deformações geométricas e composição química uniforme). Em itens com tolerância normal, estas exigências não são consideradas. A função objetivo não é linear, assim um algoritmo de programação linear, para resolver este problema, não pôde ser aplicado. Deste modo, os autores mostraram que é possível resolvê-lo utilizando programação dinâmica, porém, devido a complexidade computacional, sua utilização era impossível quando problemas reais eram

considerados. Contudo, após algumas modificações nesse algoritmo, os autores concluíram que soluções muito satisfatórias foram obtidas.

Diferentemente dos trabalhos anteriores citados, Kos e Duhovnik (2002) utilizaram um algoritmo genético híbrido para resolver um PCE. O objetivo era minimizar as perdas considerando a possibilidade de retalhos retornarem ao estoque, desde que fossem superiores a um determinado limitante fornecido pelo planejador, baseado no tipo e planejamento da produção. Considerando que o estoque era composto por objetos de diferentes comprimentos, os autores utilizaram a formulação *Variable-Sized Bin Packing Problem* (VBPP). Como os autores consideram o aproveitamento de retalhos, para manter o estoque com quantidades razoáveis de sobras estes objetos tem prioridade de uso em relação aos demais objetos do estoque (objetos padronizados). Segundo os autores, o método proposto apresentou boas soluções para o problema.

Gradisar e Trkman (2005) mostraram que os métodos existentes até então para resolver problema de corte poderiam ser melhorados fazendo uma combinação de métodos exatos e procedimentos heurísticos. Os autores desenvolveram uma abordagem na qual combinam o procedimento heurístico CUT (Gradisar *et al.* [15]), que resolve a maior parte do problema, e o método Branch-and-Bound, que resolve o problema residual final. O objetivo da combinação desses dois métodos era obter perda mínima em um tempo computacional aceitável. Ao final do processo, os retalhos gerados, se suficientemente grandes, são aproveitadas no futuro. O algoritmo proposto foi denominado de C-CUT (Combined Cutting). A ideia básica do C-CUT é encontrar uma solução temporária com o algoritmo CUT e então melhorá-la resolvendo um subproblema como método exato. De acordo com os autores, os testes computacionais realizados mostraram que a nova abordagem foi superior em relação ao algoritmo CUT.

Abuabara (2006) aplicou o PCES em uma empresa brasileira que tem em sua linha de produção o corte de tubos estruturais metálicos para a produção de aeronaves agrícolas. Para resolver o problema, o autor propôs dois modelos matemáticos, os quais foram resolvidos utilizando um software comercial. O primeiro modelo corresponde ao modelo de Gradisar (1997), porém reescrito como um Problema Inteiro Misto (PIM) e, o segundo consiste em simplificações realizadas no modelo anterior. Com algumas alterações nos dois modelos, os autores também apresentaram a possibilidade de planejar a produção em multiperíodos. O autor concluiu que os testes computacionais realizados, considerando o objetivo de minimizar a perda gerada, mostrou um melhor desempenho no segundo modelo em relação ao primeiro e as soluções apresentadas pela empresa.

O trabalho de Cherri *et al.* (2006) considera o PCES, cujo o estoque é composto por objetos padronizados (comprados pela empresa) ou não padronizados (sobras de cortes anteriores). Para resolvê-lo alterações foram realizadas em procedimentos heurísticos (construtivos e residuais) clássicos da literatura. Os procedimentos heurísticos construtivos FFD_R e $Gulosa_R$, foram desenvolvidas a partir dos procedimentos clássicos FFD (First Fit Decreasing) e Gulosa, com a finalidade de gerar padrões com perdas pequenas ou sobras. Heurísticas residuais também foram alteradas, estas consistem em obter uma solução contínua a partir da relaxação do problema linear proposto por Gilmore e Gomory (1963) e utilizando alguma técnica de arredondamento obter uma solução inteira. Com esta estratégia a demanda pode não ser atendida completamente devido ao arredondamento das soluções, o problema residual resultante é resolvido a partir de procedimentos heurísticos. Para o PCES os autores alteraram as Heurísticas Residuais FFD e Gulosa. As Heurísticas Residuais FFD_R e $Gulosa_R$ consistem em obter uma solução inteira aproximada determinada por um truncamento trivial dado pelo inteiro inferior ao fracionário obtido, se ainda restar demanda residual, aplica-se os procedimentos heurísticos FFD_R e $Gulosa_R$. Outros procedimentos heurísticos residuais alterados para resolver o PCES foram propostos por Poldi e Arenales (2005). As Heurísticas Residuais de Arredondamento Guloso (RAG -versões 1, 2 e 3) originaram as Heurísticas RAG_R versões 1, 2 e 3. Se a perda/sobra estiverem em um intervalo aceitável nos padrões gerados, estes são aceitos; caso contrário, são desfeitos. Após analisados todos os padrões, aplica-se a Heurística FFD_R na demanda residual formada pelos padrões desfeitos. Segundo os autores, suas heurísticas obtiveram um desempenho superior que suas versões clássicas.

O trabalho desenvolvido por Trkman e Gradisar (2007) tem como objetivo a minimização da perda ou dos custos sobre um determinado período de tempo, pois o estudo foi realizado considerando um PCE multiperíodo. Esses períodos podem significar turnos de trabalho (horas), dias, ou semanas. Em cada período é resolvido um PCE. A disponibilidade dos objetos e a demanda dos itens dependem de cada período, porém podem ser transferidos de um para outro durante o processo de resolução. Os autores consideram um estoque composto por objetos de comprimentos distintos e consideram a possibilidade de retalhos retornarem ao estoque, desde que sejam superiores a um determinado limitante. Como o problema se insere em multiperíodos, a sobra gerada em um determinado período é armazenada para ser cortada no período seguinte. Os autores propuseram dois modelos, pois o primeiro deles não apresentou resultados satisfatórios para o PCE multiperíodo. O segundo modelo proposto, baseado no primeiro, tinha como objetivo a minimização das perdas e

incluía custos de retornar sobras para o estoque. Segundo os autores, a solução deste modelo apresentou uma pequena perda, utilizando poucos objetos e retornando uma pequena quantidade de sobras ao estoque.

Koch (2008) centrou-se em um processo de corte em uma madeira visando minimizar os custos de produção que depende do número de repetição dos padrões de corte e a movimentação de compartimentos que contém a matéria-prima. Além disso, após o processo de corte, os retalhos gerados nos padrões de corte podem retornar ao estoque desde que o seu comprimento seja aceitável. Para resolver este problema, a abordagem utilizada foi baseada em programação linear e possibilita ao planejador de produção especificar o comprimento máximo aceitável para a perda e quais são os limitantes (inferior e superior) para o comprimento de uma sobra. Ao final do processo de corte, os planos gerados com perdas e custos são apresentados ao planejador para que ele escolha qual solução será implementada. De acordo com os autores, as soluções obtidas foram bem aceitas pela empresa.

Pinto (2008), em sua dissertação de mestrado, apresentou uma modelagem matemática apresentando três abordagens de resolução fundamentadas em minimizar o comprimento total a ser cortado e gerar sobras que são reaproveitadas em cortes futuros, e duas abordagens de resolução que combina a minimização do comprimento total a ser cortado e geração de sobras. Segundo Pinto (2008), pode-se considerar que as estratégias de utilizar soluções ótimas ou quase-ótimas do PCER para obtenção de soluções para o PCES maximizando as sobras e distribuindo-as no menor número de objetos apresentaram bons resultados.

Para o desenvolvimento das abordagens de resolução do PCES propostas, em especial foram estudadas as abordagens de Scheithauer *et al.* [31], Koch *et al.* [20] e Pinto [27]. Estes trabalhos são descritos nas seções seguintes.

3.1 MODELO PROPOSTO POR SCHEITHAUER *ET AL.* (1991)

No problema apresentado por Scheithauer *et al.* (1991), itens T_i de comprimento l_i e demandas d_i , $i = 1, \dots, m$ deveriam ser cortados de objetos de comprimento L_j , $j = 1, \dots, M$, onde o número desses objetos era suficiente para satisfazer a demanda total de itens.

Para fazer o reaproveitamento de sobras, foi utilizada a estratégia de associar um custo v_j a cada objeto de comprimento L_j , $j = 1, \dots, M$ e criar sobras R_k de

comprimento r_k com um valor w_k , $k=1,\dots,r$ que poderiam ser produzidas no processo de corte quando fosse necessário. O objetivo do problema consiste em atender a demanda, minimizando o custo dos objetos utilizados menos o valor da sobras obtidas.

Com a finalidade de descrever o modelo proposto por Scheithauer *et al.* [31], considere que o vetor $a_{jp} = (a_{1jp}, \dots, a_{(m+r)jp})^T$ representa o p -ésimo padrão de corte do j -ésimo comprimento de estoque, $p=1,\dots,n_j$, $j=1,\dots,M$, onde n_j é o número de padrões de corte admissíveis, isto é, os padrões tal que $\sum_{i=1}^m l_i a_{ijp} + \sum_{k=1}^r r_k a_{(i+k)jp} \leq L_j$, de comprimentos de estoque L_j . No modelo a variável inteira x_{jp} representa o número de vezes que o padrão de corte a_{jp} é utilizado. Desta forma, a formulação matemática é dada por:

$$\text{minimizar } Z = \sum_{j=1}^M \sum_{p=1}^{n_j} v_j x_{jp} - \sum_{j=1}^M \sum_{p=1}^{n_j} \sum_{k=1}^r w_k a_{(m+k)jp} x_{jp} \quad (3.1)$$

$$= \sum_{j=1}^M \sum_{p=1}^{n_j} (v_j - \sum_{k=1}^r w_k a_{(m+k)jp}) x_{jp} \quad (3.2)$$

$$\text{sujeito a } \sum_{j=1}^M \sum_{p=1}^{n_j} a_{ijp} x_{jp} \geq b_i, \quad i = 1, \dots, m \quad (3.3)$$

$$x_{jp} \geq 0, \quad p = 1, \dots, n_j, \quad j = 1, \dots, M \quad (3.4)$$

$$x_{jp} \text{ inteiro}, \quad p = 1, \dots, n_j, \quad j = 1, \dots, M \quad (3.5)$$

A função objetivo (3.2) minimiza o custo de utilizar os objetos menos o valor das sobras. A restrição (3.3) garante que a quantidade de itens produzidos seja maior ou igual a demanda. Por fim, a restrição (3.4) garante que as repetições x_{jp} de cada padrão de corte seja um número inteiro e não negativo.

O modelo matemático (3) tem a mesma estrutura que o modelo tradicional de corte de estoque unidimensional, de tal maneira que, pode-se empregar o Método Simplex e a Técnica de Geração de Colunas como estratégia de solução.

Dado um conjunto de m vetores colunas linearmente independentes que formam uma base B do problema relaxado ($x_{jp} \geq 0, \text{real}$) e seja $\pi = (\pi_1, \dots, \pi_m)^T$ o vetor multiplicador simplex, então o subproblema gerador de colunas é o seguinte:

$$\text{minimizar} \quad \left(v_j - \sum_{k=1}^r w_k a_{(m+k)jp} \right) - \sum_{i=1}^m \pi_i a_{ijp} \quad (3.6)$$

$$\text{sujeito a} \quad \sum_{i=1}^m l_i a_{ijp} + \sum_{k=1}^r r_k a_{(m+k)jp} \leq L_j \quad (3.7)$$

$$a_{ijp} \geq 0, \text{ inteiro}, i = 1, \dots, m+r \quad (3.8)$$

$$j = 1, \dots, M \quad (3.9)$$

para cada objeto de comprimento $L_j, j=1, \dots, M$.

Conclui-se a ideia escolhendo a coluna, ou o padrão de corte, no qual

$$c^* = \min \left\{ \min_j Z_j, \min_i \pi_i \right\}$$

$$\text{com } Z_j = \min \left(v_j - \sum_{k=1}^r w_k a_{(m+k)jp} \right) - \sum_{i=1}^m \pi_i a_i^{jp}.$$

Feito isto analisa-se o valor de c^* , se este é maior ou igual a zero então a

base corrente B (o conjunto de padrões de corte escolhido) é um ótimo com respeito ao problema relaxado. Caso contrário, uma nova coluna (um padrão de corte ou um vetor unitário correspondente as variáveis de folga) é obtida e o algoritmo simplex continua. Para a computação de Z_j os autores definiram

$$f(l) = \max \left\{ \sum_{i=1}^m \pi_i a_i \mid \sum_{i=1}^m l_i a_i \leq l, a_i \geq 0, \text{ inteiro}, i = 1, \dots, m \right\}$$

Note que $f(l)$ é o valor ótimo

do Problema da Mochila com *right hand side* (RHS) l . Desta forma Z_j pode ser reescrito da seguinte maneira:

$$Z_j = v_j - \max \left\{ \sum_{k=1}^r w_k a_{m+k} + f \left(L_j - \sum_{k=1}^r r_k a_{m+k} \right) \mid \right. \quad (3.10)$$

$$\left. \sum_{k=1}^r r_k a_{m+k} \leq L_j, \right. \quad (3.11)$$

$$a_{m+k} \geq 0, \text{ inteiro}, k = 1, \dots, r \}, \quad (3.12)$$

$$j = 1, \dots, M. \quad (3.13)$$

E foi assumido que

1. $w_k/r_k \leq v_j/L_j$ para todo k, j com $r_k \leq L_j$;

2. $w_k/r_k \leq w_t/r_t$ para todo $r_k \leq r_t$;
3. se r_k e r_t são sobras então $r_k + r_t$ é também uma sobra, para todo k, t .

Por (1), (2) e (3) existe uma solução ótima de (3.10) com

$$\sum_{k=1}^r a_{m+k} \leq 1, \quad (3.14)$$

sendo reduzida a

$$\begin{aligned} Z_j &= v_j - \max\{f(L_j), w_k + f(L_j - r_k) | \\ &\quad r_k \leq L_j, k = 1, \dots, r\}, \\ &\quad j = 1, \dots, M. \end{aligned}$$

Se (1), (2) e (3) ou (3.14) são válidos segue que

$$\begin{aligned} c^* &= \min \{ \min_i \pi_i, \min_j v_j - \max\{f(L_j), w_k + f(L_j - r_k) | \\ &\quad r_k \leq L_j, k = 1, \dots, r\} \} \end{aligned}$$

Os autores concluíram que deve-se ter cautela com a escolha dos valores para as sobras pois altera significativamente a função objetivo resultando também numa variação da quantidade de objetos cortados. Sem a utilização de sobras definidas inicialmente os resultados foram piores no sentido de que houve grandes perdas, que aqueles utilizando os padrões de corte com tais sobras que gerou muito menos perda, mas com o uso de mais objetos.

3.2 APLICAÇÃO EM UMA INDÚSTRIA DE PROCESSAMENTO DE MADEIRA APRESENTADA POR KOCH *ET AL.* (2008)

O trabalho de Koch et al. (2008) centra-se em um processo de corte em uma indústria de processamento de madeira em que o material de estoque é cortado de modo a fornecer os itens exigidos pelos clientes nas qualidades, tamanhos e quantidades desejadas.

Nordlam GmbH é uma companhia de operação internacional desta indústria de processamento de madeira que produz vigas em vários tamanhos e qualidades em sua fábrica Magdeburg. Estas vigas são fabricadas colando lâminas de madeira em um processo de produção multiperíodo. O cliente faz a encomenda e é verificado se as vigas que estão no estoque servirão para satisfazer o que foi encomendado ou novas vigas serão produzidas. O caso considerado no trabalho é que o pedido é satisfeito com as vigas do depósito. O problema então é cortar nos comprimentos desejados. O processo de corte está fortemente

ligado ao processo de transporte e os processos de inventário que não podem ser negligenciados no planejamento de corte.

Planejamento do corte analisa uma família de produtos, produtos que podem ser agrupados com respeito as características tais como largura, altura e qualidade.

No depósito há dois tipos de vigas, uma chamada de material padrão que tem 24 m e as sobras.

O plano de corte fica determinado tendo a lista dos materiais disponíveis e os comprimentos em que estes materiais serão cortados.

As vigas são armazenadas em compartimentos móveis com comprimentos e qualidades diferentes. Esses compartimentos são movidos do depósito para a área de corte. Os objetos requeridos ficam e o compartimento retorna ao depósito. Para satisfazer a demanda total pode ser que necessite mover mais de um compartimento do depósito.

O corte é feito e os tamanhos gerados que não fazem parte das encomendas, os retalhos, voltam para o depósito ou não, dependendo de seu comprimento.

Segundo os autores, Nordlam GmbH afirmavam que o processo que eles adotavam não era bom. Os custos eram elevados (perda e o custo de planejamento). Normalmente, o planejamento era feito três dias antes do processo efetivo de corte. O planejador, um trabalhador da indústria, determinava quais pedidos seriam cortados, a quantidade de material de estoque que seria usado e os planos de corte que seriam aplicados. O planejador levava um dia para fazer isso.

Com relação aos custos tem-se os custos de materiais (o que foi encomendado e a perda), os custos de manuseio e transporte (isto é, buscar o compartimento para a área de corte e devolvê-lo depois) e o custo de inventário (as quantidades e o tempo em que as vigas ficam estocadas). Como o pedido do cliente tem de ser satisfeito, este custo está fixo e como os pedidos são aleatórios os custos de inventário não são especificados.

Segundo os autores minimizar a soma de material e os custos de manuseio acaba reduzindo os outros fatores de custo. Eles concluíram que a entrada de material padrão e a criação de novas sobras deve ser controlável. Além de reaproveitar as sobras, o problema requer uma análise de custo associado ao manuseio de compartimentos móveis. Desejando reduzir o custo do processo de corte e manuseio dos compartimentos, os autores apresentam uma abordagem de solução baseada num modelo de programação linear. Esta abordagem alguns componentes que devem ser apresentados pelo planejador. Ele deve entrar com a perda aceitável máxima e os limitantes superior e inferior das sobras, os pedidos dos clientes e o material disponível. A partir daí um modelo de programação linear é resolvido.

Conjuntos de índices

I : conjunto de índices de tipos de itens ($I = \{1, \dots, m\}$);

J : conjunto de índices de tipos de objetos ($J = \{1, \dots, M\}$);

K : conjunto de índices de tipos de compartimentos;

$P(j)$: conjunto de índices de tipos de todos os padrões de corte factíveis que podem ser aplicados a tipos de objetos de índice j ($j \in J$);

R : conjunto de índices de tipos de intervalos que definem sobras factíveis.

Constantes

a_{ijp} : número de vezes que o tipo de item i ($i \in I$) aparece no padrão de corte p ($p \in P(j)$) para tipos de objetos de índice j ($j \in J$);

c^{comp} : custo de manuseio do compartimento;

c^{perda} : custo de perda por unidade de comprimento;

c^{sobra} : custo de sobra por unidade de comprimento;

$c_{jp}^{retalho}$: custo de retalho gerado por padrões de corte p ($p \in P(j)$) para tipos de objetos de índice j ($j \in J$);

$$c_{jp}^{retalho} = \begin{cases} c^{perda}(L_j - \sum_{i \in I} l_i a_{ijp}), & \text{se o padrão } p \text{ (} p \in P(j) \text{) contém uma perda;} \\ c^{sobra}(L_j - \sum_{i \in I} l_i a_{ijp}), & \text{se o padrão } p \text{ (} p \in P(j) \text{) contém uma sobra;} \\ 0, & \text{caso contrário.} \end{cases}$$

d_i : demanda do item de índice $i = 1, \dots, m$;

l_i : comprimento do item de índice $i = 1, \dots, m$;

s_{jk} : quantidade disponível do objeto de comprimento L_j no compartimento móvel $k \in K$;

t_{jp} : comprimento da sobra no padrão de corte p ($p \in P(j)$) aplicado no objeto de índice j ($j \in J$);

L_j : comprimento do tipo de objeto de índice j ($j \in J$).

Variáveis

x_{jp} : número de vezes que o padrão de corte p ($p \in P(j)$) é aplicado no objeto de índice j ($j \in J$);

y_k : indicador de utilização do compartimento k ($k \in K$);

$$y_k = \begin{cases} 1, & \text{se ao menos um objeto do compartimento móvel de índice } k \text{ foi usado;} \\ 0, & \text{caso contrário.} \end{cases}$$

Parâmetros

L_{perda} : comprimento máximo de retalho aceito como perda;

$qntObj_j$: número máximo de objetos de índice j ($j \in J$) aceito na solução;

L_{sobra_r} : comprimento mínimo de retalho no intervalo r ($r \in R$) aceito como sobra;

$L_{sobra_r}^{max}$: comprimento máximo de sobra no intervalo r ($r \in R$) aceito como sobra.

Com as seguintes condições:

$$\begin{aligned} L_{perda} &< \min(L_{sobra_r}, r \in R) \\ L_{sobra_r} &< L_{sobra_r}^{max}, r \in R \end{aligned}$$

Um padrão de corte p no objeto de índice j é representado por um vetor $a_{jp} =$

$\{a_{1jp}, \dots, a_{mjp}\}$ que deve satisfazer as seguintes propriedades:

$$\begin{aligned} \sum_{i \in I} l_i a_{ijp} + t_{jp} &= L_j, \quad j \in J, p \in P(j); \\ \sum_{i \in I} a_{ijp} &\geq 1, \quad j \in J, p \in P(j); \\ a_{ijp} &\geq 0 \text{ e inteiro}, \quad i \in I, \quad j \in J, p \in P(j); \\ t_{jp} &\geq 0, \quad j \in J, p \in P(j). \end{aligned}$$

Um padrão de corte é dito factível se uma das condições valem:

1. O objeto disponível de comprimento L_j é usado completamente, isto é,

$$t_{jp} = 0 \Leftrightarrow \sum_{i \in I} l_i a_{ijp} = L_j;$$

2. O retalho é suficientemente pequeno para ser aceito como perda, isto é,

$$0 < t_{jp} = L_j - \sum_{i \in I} l_i a_{ijp} \leq L_{perda};$$

3. O retalho tem um comprimento que é aceito como sobra, isto é,

$$L_{sobra_r} \leq t_{jp} = L_j - \sum_{i \in I} l_i a_{ijp} \leq L_{sobra_r}^{max}, \quad r \in R$$

Assim o modelo apresentado por Koch *et al.*(2008) é dado por:

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp}^{retalho} x_{jp} + \sum_{k \in K} c^{comp} y_k \quad (3.15)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} = d_i, \quad i \in I \quad (3.16)$$

$$\sum_{p \in P(j)} x_{jp} \leq \sum_{k \in K} s_{jk} y_k, \quad j \in J \quad (3.17)$$

$$\sum_{p \in P(j)} x_{jp} \leq qntObj_j, \quad j \in J \quad (3.18)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, \quad p \in P(j) \quad (3.19)$$

$$y_k \in \{0, 1\} \quad k \in K \quad (3.20)$$

A função objetivo (3.15) apresenta os custos do problema. O primeiro termo inclui os custos de perda e as sobras enquanto o segundo termo o custo de movimentação dos compartimentos. A restrição 3.16 garante que a demanda seja satisfeita. A (3.17) é caracterizada por uma restrição de estoque, garantindo que para cada tipo de material somente o número disponível de objetos sejam usados. (3.18) limita o número de objetos de índice j . Em particular para $j=1$ tem-se o material padrão. Restrição (3.19) garante que o número de

vezes que o padrão de corte é utilizado é inteiro. E (3.20) indica se um compartimento k está sendo usado ou não.

Os autores confeccionaram um *software* em plataforma *Windows*, que ao iniciar um planejamento, as informações são importadas diretamente do banco de dados da indústria. Entre esses dados estão os comprimentos e as quantidades correspondentes aos pedidos dos clientes, bem como o inventário disponível juntamente com o número dos compartimentos em que está armazenado. Se necessário o planejador especifica outros parâmetros, como por exemplo, os intervalos de sobra e os custos por unidade de comprimento de perda.

Os resultados da otimização são posteriormente apresentados ao planejador e cabe a ele selecionar um para ser executado. No intuito de apoiá-lo na sua decisão, as informações adicionais sobre os fatores de custos (informação explícita de perda, custo de manuseio dos compartimentos), e os correspondentes padrões de corte são apresentados pelo *software*.

Segundo os autores o sistema de suporte à decisão foi bem aceito pelo planejador já que os resultados apresentados foram melhores que os seus planos gerados manualmente. Do ponto de vista econômico o suporte à decisão é também muito satisfatório. Todavia o surgimento de um conjunto de dados que continha um grande número de padrões de corte demonstrou uma deficiência deste sistema, a impossibilidade de gerar o modelo de otimização descrito.

3.3 ESTRATÉGIAS DE RESOLUÇÃO POR MEIO DE DOIS OBJETIVOS PROPOSTAS POR PINTO (2008)

Pinto (2008) apresenta uma modelagem matemática para o Problema de Corte de Estoque unidimensional, com Reaproveitamento de Sobras, e propõe diferentes abordagens de resolução. Este tipo de problema tem grande importância para o planejamento da produção, e a abordagem de reaproveitamento de sobras pode representar uma enorme vantagem econômica ao reduzir os efeitos negativos associados a perda de material. O trabalho desenvolvido visa resolver o problema apresentado com cinco abordagens de solução:

- três fundamentadas em duas etapas:
 - primeira etapa (comum às três): minimizar comprimento total a ser cortado;
 - segunda etapa: gerar sobras que possam ser reaproveitadas futuramente.

- duas em uma única etapa, que tenta combinar os objetivos das etapas descritas anteriormente.

Apresenta três modelos matemáticos para o PCE. O modelo matemático para o problema de corte de estoque unidimensional, considerando o objetivo de minimizar o número de objetos a serem usados, o modelo com vários tipos de barras em estoque, em quantidades limitadas, considerando o objetivo de minimizar a perda produzida e o modelo em que é conhecido um limitante superior para o número de objetos necessários para o atendimento da demanda. Estes problemas foram formulados da seguinte maneira:

O primeiro modelo, denominado Problema de Corte de Estoque Unidimensional Clássico (PCEC) é formulado da seguinte forma:

P : são padrões de corte possíveis;

L : comprimento dos objetos;

l_i : comprimento do item i , $i = \{1, \dots, m\}$;

d : o vetor de demanda, onde $d = (d_1, \dots, d_m)^T$, com d_i o número de itens de comprimento l_i encomendados;

x_p : representa o número de vezes que o padrão de corte de índice p é usado, $p = \{1, \dots, P\}$;

a_p : representa um padrão de corte unidimensional, onde $a_p = (a_{1p}, \dots, a_{mp})^T$ satisfazendo a restrição física $l_1 a_{1p} + l_2 a_{2p} + \dots + l_m a_{mp} \leq L$, $a_{ip} \geq 0$, $i = \{1, \dots, m\}$.

$$\min \sum_{p=1}^P x_p \quad (3.21)$$

$$\text{s. a} \quad a_1 x_1 + a_2 x_2 + \dots + a_p x_p = d \quad (3.22)$$

$$x_p \geq 0 \text{ e inteiro, } p = \{1, \dots, P\} \quad (3.23)$$

A função objetivo (3.21) minimiza o total de objetos em estoque a serem cortados. A restrição (3.22) garante que a quantidade de itens produzidos seja exatamente igual à demanda e, a (3.23) garante que a repetição de cada padrão de corte de índice p seja um número inteiro não negativo.

O segundo modelo, denominado Problema de Corte de Estoque Unidimensional Restrito (PCER) é escrito como:

L_j : comprimento do objeto de índice j , $j = \{1, \dots, M\}$;

e_j : quantidade disponível do objeto de comprimento L_j ;

x_{jp} : o total de objetos que foram cortados, segundo o padrão de corte de índice p ;

l_i : comprimento do item i , $i = \{1, \dots, m\}$;

$P(j)$: conjunto de índices dos padrões possíveis para o j -ésimo objeto;

a_{jp} : vetor de padrão de corte p no objeto de comprimento L_j , onde $a_{jp} = \{a_{1jp}, \dots, a_{mjp}\}$;

a_{ijp} : quantidade de vezes que o item de comprimento l_i aparece no padrão de índice p , no objeto de comprimento L_j ;

T_{jp} : perda produzida pelo padrão de índice p no objeto de comprimento L_j , onde $T_{jp} = L_j - \sum_{i=1}^m l_i a_{ijp}$;

$$\begin{aligned}
& \text{minimizar} && \sum_{p \in P(1)} T_{1p}x_{1p} + \sum_{p \in P(2)} T_{2p}x_{2p} + \dots + \sum_{p \in P(M)} T_{Mp}x_{Mp} \\
& \text{sujeito a} && \sum_{p \in P(1)} a_{1p}x_{1p} + \sum_{p \in P(2)} a_{2p}x_{2p} + \dots + \sum_{p \in P(M)} a_{Mp}x_{Mp} = d \\
& && \sum_{p \in P(j)} x_{jp} \leq e_j, \quad j = 1, 2, \dots, M \\
& && x_{jp} \geq 0 \text{ e inteiro, } p \in P(j), \quad j = 1, 2, \dots, M
\end{aligned}$$

O terceiro modelo possui um limitante superior K , que é conhecido, para o número de objetos necessários para o atendimento da demanda.

a_{ik} : número de vezes que o item de índice i é cortado do objeto de índice k ;

$$y_k = \begin{cases} 1, & \text{se o } k\text{-ésimo objeto é utilizado;} \\ 0, & \text{caso contrário.} \end{cases}$$

$$\text{minimizar} \quad \sum_{k=1}^K y_k \quad (3.24)$$

$$\text{sujeito a} \quad \sum_{k=1}^K a_{ik} \geq d_i, \quad i = 1, \dots, m \quad (3.25)$$

$$\sum_{i=1}^I l_{ik}a_{ik} = Ly_k, \quad k = 1, \dots, K \quad (3.26)$$

$$a_{ik} \geq 0 \text{ e inteiro, } i = 1, \dots, m, \quad k = 1, \dots, K \quad (3.27)$$

$$y_k \in \{0, 1\}, \quad k = 1, \dots, K \quad (3.28)$$

A função objetivo (3.24) minimiza o número de objetos utilizados. A restrição (3.25) garante o atendimento à demanda dos itens, a restrição (3.26) garante que se o padrão de corte de índice k for utilizado, então ele satisfaz a restrição física. Por fim, as restrições (3.27) e (3.28) definem os tipos de variáveis do problema.

Apresenta um modelo para o Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras (PCES), que consiste em, dado um estoque composto por objetos de vários comprimentos, cuja disponibilidade é limitada, mas, suficiente para o atendimento de uma demanda de itens, encontre um conjunto de padrões que atenda a demanda diminuindo as sobras ao nível mínimo, gerando o mínimo de perdas e o máximo de sobras, e que estas sobras estejam distribuídas no menor número de objetos.

Conjuntos de índices

I : conjunto de índices de tipos de itens ($I = \{1, \dots, m\}$);

J : conjunto de índices de tipos de objetos;

$P(j)^\alpha$: conjunto de índices para os padrões com perdas;

$P(j)^\beta$: conjunto de índices para os padrões com sobras;

Constantes

L_j : comprimento do objeto de índice j ;

l_i : comprimento do item de índice i ;

S : comprimento mínimo que representa uma sobra, ($S > 0$);

α_{ijp} : quantidade de vezes que o item de comprimento l_i , $i = \{1, \dots, m\}$ aparece no padrão de índice p , $p = 1, \dots, P(j)^\alpha$, no objeto de comprimento L_j ;

α_{jp} : representa um padrão com perda se $T_{jp} = L_j - \sum_{i=1}^m l_i \alpha_{ijp} < S$;

β_{ijq} : quantidade de vezes que o item de comprimento l_i , $i = \{1, \dots, m\}$ aparece no padrão de índice p , $p = 1, \dots, P(j)^\beta$, no objeto de comprimento L_j ;

β_{jq} : representa um padrão com sobra se $S_{jq} = L_j - \sum_{i=1}^m l_i \beta_{ijq} \geq S > 0$;

e^j : quantidades disponível do objeto de comprimento L_j , ($j = 1, \dots, M$);

d : o vetor de demanda, onde $d = (d_1, \dots, d_m)^T$, com d_i o número de itens de comprimento l_i encomendados;

Variáveis

x_{jp}^α : define o total de objetos de comprimento L_j que foram cortados, segundo o padrão de corte de índice p , $p = 1, \dots, P(j)^\alpha$ produzindo perdas;

x_{jq}^β : define o total de objetos de comprimento L_j que foram cortados, segundo o padrão de corte de índice p , $p = 1, \dots, P(j)^\beta$ produzindo sobras;

Nestas condições o modelo PCES consiste em:

$$\begin{aligned}
 &\text{minimizar} && \sum_{p \in P(1)^\alpha} T_{1p} x_{1p}^\alpha + \dots + \sum_{p \in P(M)^\alpha} T_{Mp} x_{Mp}^\alpha + \sum_{q \in P(1)^\beta} S_{1q} x_{1q}^\beta + \dots + \sum_{q \in P(M)^\beta} S_{Mq} x_{Mq}^\beta \\
 &\text{maximizar} && \sum_{q \in P(1)^\beta} S_{1q} x_{1q}^\beta + \dots + \sum_{q \in P(M)^\beta} S_{Mq} x_{Mq}^\beta \\
 &\text{minimizar} && \sum_{q \in P(1)^\beta} x_{1q}^\beta + \dots + \sum_{q \in P(M)^\beta} x_{Mq}^\beta \\
 &\text{sujeito a} && \sum_{p \in P(1)^\alpha} \alpha_{1p} x_{1p}^\alpha + \dots + \sum_{p \in P(M)^\alpha} \alpha_{Mp} x_{Mp}^\alpha + \sum_{q \in P(1)^\beta} \beta_{1q} x_{1q}^\beta + \dots + \sum_{q \in P(M)^\beta} \beta_{Mq} x_{Mq}^\beta = d \\
 &&& \sum_{p \in P(j)^\alpha} x_{jp}^\alpha + \sum_{q \in P(j)^\beta} x_{jq}^\beta \leq e_j, \quad j = 1, \dots, M \\
 &&& x_{jp}^\alpha, x_{jq}^\beta \geq 0 \text{ e inteiros, } p \in P(j)^\alpha, q \in P(j)^\beta, j = 1, \dots, m
 \end{aligned}$$

A diferença entre os modelos do PCER e PCES é que neste último foi feita uma distinção entre os padrões viáveis, aqueles que geram sobras.

Os resultados obtidos após as primeiras simulações foram:

Propriedade 1: solução ótima do PCES também é do PCER;

Propriedade 2: se a soma das perdas da solução do PCER é menor que S , então esta solução é ótima para o PCES.

Estas propriedades indicaram que é uma boa tática modelar heurísticas baseadas em obter o conjunto das soluções ótimas ou quase ótimas do PCER e, em seguida buscar nesse conjunto soluções que maximizem as sobras, distribuindo-as no menor número de objetos. Com isso Pinto (2008) propôs cinco estratégias de resolução para o PCES. Três estratégias desenvolvidas possuem duas etapas:

Primeira etapa: é definida pelo seguinte modelo,

Conjuntos de índices

I : conjunto de índices de tipos de itens ($I = 1, \dots, m$);

J : conjunto de índices de tipos de objetos;

Constantes

L_j : comprimento do objeto de índice j ($j \in J$);

d_i : demanda do item de índice i ($i \in I$);

l_i : comprimento do item de índice i ($i \in I$);

Variáveis

a_{ij} : número de itens de índice i cortados do objeto de índice j ($j \in J$);

$$y_j = \begin{cases} 1, & \text{se o } j\text{-ésimo objeto é utilizado;} \\ 0, & \text{caso contrário.} \end{cases}$$

$$\text{minimizar} \quad \sum_{j=1}^J y_j L_j \quad (3.29)$$

$$\text{sujeito a} \quad \sum_{j=1}^J a_{ij} = d_i, \quad i = 1, \dots, m \quad (3.30)$$

$$\sum_{i=1}^m l_i a_{ij} \leq L_j y_j, \quad j = 1, \dots, J \quad (3.31)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i = 1, \dots, m, \quad j = 1, \dots, J \quad (3.32)$$

$$y_j \in \{0, 1\}, \quad j = 1, \dots, J \quad (3.33)$$

que consiste em selecionar os objetos que deverão ser cortados para atender a demanda, minimizando o comprimento total a ser cortado. Esta etapa é comum às três estratégias.

A função objetivo (3.29) minimiza o comprimento total utilizado para atender a demanda. A restrição (3.30) garante que a quantidade de itens produzidos seja igual à demanda. A restrição (3.31) funciona da seguinte forma: se o objeto de índice j está sendo usado ($y_j = 1$) a mesma equivale a uma restrição de mochila, por outro lado, se o objeto de índice j não está sendo usado ($y_j = 0$) a restrição obriga que $a_{ij} = 0$, $i = 1, \dots, m$. Por fim, as restrições (3.32) e (3.33) definem as variáveis do problema.

Segunda etapa: é encontrar um conjunto de padrões que atenda a demanda diminuindo as sobras ao nível mínimo, gerando o mínimo de perdas e o máximo de sobras, e que estas sobras estejam distribuídas no menor número de objetos.

- Estratégia 1

Constantes

J : conjunto de índices de tipos de objetos ($j = 1, \dots, M$);

I : conjunto de índices de tipos de itens ($i = 1, \dots, m$);

N : número de objetos escolhidos na primeira etapa;

L_j : comprimento do objeto de índice j ($j \in J$);

d_i : demanda do item de índice i ($i \in I$);

l_i : comprimento do item de índice i ($i \in I$);

S : comprimento mínimo aceito para resíduo.

Variáveis

a_{ij} : número de itens de índice i ($i \in I$), cortados do objeto de índice j ($j \in J$);

$w_j = \begin{cases} 1, & \text{se o } j\text{-ésimo objeto é escolhido para ter sobra;} \\ 0, & \text{caso contrário.} \end{cases}$

$$\text{minimizar} \quad \sum_{j=1}^N (L_j - \sum_{i=1}^m l_i a_{ij}) \quad (3.34)$$

$$\text{sujeito a} \quad \sum_{j=1}^N w_j = R \quad (3.35)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in I \quad (3.36)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq w_j L_j + (1 - w_j)(S - 1), \quad j \in J \quad (3.37)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq w_j S, \quad j \in J \quad (3.38)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in I, j \in J \quad (3.39)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (3.40)$$

Esta estratégia minimiza as perdas visando a geração de sobras.

A função objetivo (3.34) minimiza as perdas e sobras do processo. A restrição (3.35) garante que o número de objetos utilizados para concentrar as perdas seja igual ao número R previamente fixado. A restrição (3.36) corresponde ao atendimento exato das demandas dos itens. As restrições (3.37) e (3.38) juntas, indicam para cada objeto, que o comprimento remanescente do corte dos itens deve estar no intervalo $[S, L_j]$ quando $w_j = 1$ (o objeto deve concentrar perdas), ou deve estar no intervalo $[0, S)$ quando $w_j = 0$ (o objeto não deve concentrar perdas). Como as variáveis do problema são inteiras, na restrição (3.37), $(1 - w_j)$ foi multiplicado por $(S - 1)$ para que o comprimento remanescente do corte dos itens

esteja no intervalo $[0,S)$ quando $w_j = 0$. Por fim, as restrições (3.39) e (3.40) definem as variáveis do problema.

- Estratégia 2:

Constante

R : número de objetos usados para concentrar perdas.

Variáveis

T_j : perda no objeto de índice $j=1, \dots, N$;

$$y_j = \begin{cases} 1, & \text{se o } j\text{-ésimo objeto é utilizado;} \\ 0, & \text{caso contrário.} \end{cases}$$

$$w_j = \begin{cases} 1, & \text{se o } j\text{-ésimo objeto é escolhido para ter resíduo;} \\ 0, & \text{caso contrário.} \end{cases}$$

$$\text{minimizar} \quad \sum_{j=1}^N T_j \quad (3.41)$$

$$\text{sujeito a} \quad \sum_{j=1}^N w_j = R \quad (3.42)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in m \quad (3.43)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq L_j + (1 - w_j)M, \quad j \in N \quad (3.44)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq S - (1 - w_j)M, \quad j \in N \quad (3.45)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq T_j + w_j M, \quad j \in N \quad (3.46)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq T_j - w_j M, \quad j \in N \quad (3.47)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i \in m, \quad j \in N \quad (3.48)$$

$$T_j \geq 0 \text{ e inteiro}, \quad j \in N \quad (3.49)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (3.50)$$

Esta estratégia minimiza as perdas, na forma de uma variável, visando a geração de sobras.

A função objetivo (3.41) minimiza as perdas do processo. As restrições (3.42)e(3.43) são semelhantes às (3.35) e (3.36), respectivamente. A restrição (3.44) em conjunto com a restrição (3.45), visa gerar sobras quando $w_j = 1$ e ficam inativas quando $w_j = 0$.

As restrições (3.46) e (3.47) juntas, definem perdas quando $w_j = 0$ e ficam inativas quando $w_j = 1$. Por fim, as restrições (3.48) - (3.50) definem as variáveis do problema.

- **Estratégia 3:**

As restrições do modelo da segunda etapa é parecida com a da primeira etapa, mas a função objetivo bem diferente.

$$\text{maximizar} \quad \sum_{j=N-(R-1)}^N (L_j - \sum_{i=1}^m l_i a_{ij}) \quad (3.51)$$

$$\text{sujeito a} \quad \sum_{j=1}^N a_{ij} = d_i, \quad i = 1, \dots, m \quad (3.52)$$

$$\sum_{i=1}^m l_i a_{ij} \leq L_j, \quad j = 1, \dots, J \quad (3.53)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i = 1, \dots, m, \quad j = 1, \dots, J \quad (3.54)$$

Esta estratégia maximiza as perdas em um número reduzido de objetos.

O objetivo (4.33) maximiza as perdas nos objetos de índices $j = N - (R - 1), \dots, N$. A restrição (4.34) garante que a quantidade de itens produzidos seja igual à demanda. A restrição (4.35) é uma restrição física de mochila e, a restrição (4.36) define a variável do problema. Observe que ao fazer $R = 1$, optou-se por concentrar as perdas no último objeto e, neste caso, foi feita uma ordenação decrescente dos comprimentos dos objetos, isto é, $L_1 \leq \dots \leq L_N$. Optou-se ordenar os objetos desta forma para que o modelo priorizasse cortar os objetos de comprimentos menores que são possivelmente sobras de outros processos de corte realizados anteriormente. As outras duas estratégias são:

- **Estratégia 4:**

Com a mesma nomenclatura das estratégias 1, 2 e 3 o modelo consiste em

$$\text{minimizar } \sum_{j=1}^N y_j L_j \quad (3.55)$$

$$\text{sujeito a } \sum_{j=1}^N w_j = R \quad (3.56)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in I \quad (3.57)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq w_j L_j + (1 - w_j)(S - 1), \quad j \in J \quad (3.58)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq w_j S, \quad j \in N \quad (3.59)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i \in m, \quad j \in J \quad (3.60)$$

$$y_j \in \{0, 1\}, \quad j \in J \quad (3.61)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (3.62)$$

A função objetivo (3.55) minimiza o comprimento total cortado para atender a demanda. A restrição (3.56) garante que o número de objetos que o modelo usa para acumular perdas seja igual a R . A restrição (3.57) garante que a quantidade de itens produzidos seja igual à demanda. As restrições (3.58) e (3.59) são equivalentes às (3.37) e (3.38) e, juntas indicam para cada objeto selecionado ($y_j = 1$), que o comprimento remanescente do corte dos itens, deve estar no intervalo $[S, L_j]$ quando $w_j = 1$ (o objeto deve concentrar perdas), ou deve estar no intervalo $[0, S)$ quando $w_j = 0$ (o objeto não deve concentrar perdas). Por fim, as restrições (3.60)-(3.62) definem as variáveis do problema.

- **Estratégia 5:**

Explora o modelo da estratégia 2.

$$\text{minimizar} \quad \sum_{j=1}^N (T_j + y_j L_j) \quad (3.63)$$

$$\text{sujeito a} \quad \sum_{j=1}^N w_j = R \quad (3.64)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in m \quad (3.65)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq L_j + (1 - w_j)M, \quad j \in N \quad (3.66)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq S - (1 - w_j)M, \quad j \in N \quad (3.67)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq T_j + w_j M, \quad j \in N \quad (3.68)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq T_j - w_j M, \quad j \in N \quad (3.69)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i \in m, \quad j \in N \quad (3.70)$$

$$T_j \geq 0 \text{ e inteiro}, \quad j \in N \quad (3.71)$$

$$y_j \in \{0, 1\}, \quad j \in J \quad (3.72)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (3.73)$$

Nestas duas últimas estratégias tenta-se combinar objetivos da primeira e da segunda etapa no intuito de gerar sobras.

A função objetivo (3.65) minimiza a perda e o comprimento total cortado para atender a demanda. A restrição (3.64) define o número de objetos onde serão acumuladas as perdas e a restrição (3.66) garante que a quantidade de itens produzidos seja igual à demanda. Para cada objeto selecionado ($y_j=1$), a restrição (3.67) em conjunto com a restrição (3.68), visam gerar sobras quando $w_j=1$ e ficam inativas quando $w_j=0$, agindo de forma similar às restrições (3.44) e (3.45). Do mesmo modo que as restrições (3.46) e (3.47), as restrições (3.69) e (3.70) juntas, definem perdas quando $w_j=0$ e ficam inativas quando $w_j=1$. Por fim, as restrições (3.71)-(3.73) definem as variáveis do problema.

Concluiu-se que selecionar objetos que atendam a demanda e concentrar as sobras no menor número desses objetos é uma boa estratégia de solução já que na prática as indústrias privilegiam manter as perdas no seu patamar mínimo (que é equivalente a minimizar o comprimento total cortado).

Os exemplos analisados mostraram que a estratégia 5 se destacou em relação as outras quatro estratégias relacionadas anteriormente e em relação ao algoritmo iterativo

de Sinuany-Stern e Weiner [33], as heurísticas de Cherri [7]. A estratégia 1 não teve grandes resultados. E as outras estratégias tiveram seus bons resultados entre uns e outros.

Se permitido um corte excedente para o atendimento da demanda percebeu-se que a tática de minimizar o comprimento total a ser cortado não é a melhor para selecionar os objetos na primeira etapa, pois ao selecionar os objetos já é necessário visar a geração de sobras, ficando para uma segunda etapa a obtenção dos padrões de corte que melhor correspondem a isto.

CAPÍTULO 4

ESTRATÉGIAS DE RESOLUÇÃO PARA O PCES

Algumas observações foram feitas a respeito de características dos modelos apresentados nas seções 3.1 e 3.2, as quais poderiam ser modificadas, e então foram tomadas como base para o desenvolvimento das abordagens propostas neste trabalho. Conforme apresentado, o modelo de Scheithauer *et al.* [31] considera sobras de comprimentos pré-definidos, com um valor associado a cada comprimento, os quais poderiam ser produzidos no processo de corte quando fosse necessário. Entretanto, observa-se que definir o comprimento das sobras, bem como atribuir valores, pode não ser uma tarefa simples, e pode alterar significativamente o resultado final. No trabalho desenvolvido por Koch *et al.* [20], por sua vez, existe a necessidade de se conhecer todos os padrões de corte possíveis, o que torna intratáveis problemas com dezenas de itens a serem produzidos. Nesse sentido, é proposta uma metodologia de resolução por geração de colunas, que permite a resolução de problemas maiores, em que as sobras são geradas a partir de subproblemas, com comprimentos em um intervalo definido pelo planejador. Para isso, os subproblemas geradores de padrões de corte foram escritos a partir do trabalho de Scheithauer *et al.* [31], na forma de um problema de mochila. Além disso, foi utilizada a função objetivo apresentada por Koch *et al.* [20] porém desconsiderando a movimentação de compartimentos, e foram acrescentadas restrições que estabelecem intervalos não-aceitáveis para os retalhos, consideração que não é feita por eles. Nesta metodologia proposta, foi utilizada a definição de padrão dada em Koch *et al.* [20], o que não torna trivial a definição de um único tipo de gerador de colunas. Por isso, foram desenvolvidos dois tipos de geradores de colunas (perdas e sobras, excluindo intervalos não-aceitáveis), como será apresentado a seguir.

Assim, como metodologia para resolução do PCES, foram sugeridas quatro diferentes abordagens para este problema, utilizando o Método Simplex com Geração de Colunas. Nas três primeiras abordagens, após encontrar a solução do PCES com as condições de integralidade das variáveis relaxadas, a solução inteira do problema é obtida através do Método Branch-and-Bound. Na última abordagem, a solução inteira é obtida arredondando a solução contínua para o inteiro mais próximo, e resolvendo um problema residual se a solução inteira obtida for inviável. Cada uma das abordagens propostas são mostradas a seguir.

4.1 ABORDAGEM M1P&R

Nesta primeira abordagem de resolução, o PCES é modelado desconsiderando limitações no estoque, isto é, a quantidade de material disponível é suficiente para atender à demanda, e todo tipo de objeto está disponível quantas vezes for necessário, ou seja, disponível enquanto for vantajoso para o processo de corte.

4.1.1 Modelagem Matemática Proposta I

Dessa forma, para esta abordagem, o PCES é definido como a seguir:

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (4.1)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i, \quad i \in I \quad (4.2)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, \quad p \in P(j) \quad (4.3)$$

Onde:

1. L_j é o comprimento do objeto de índice j do estoque;
2. $c_{jp}(a_{jp}) = c^{\text{processo}}(\sum_{i \in I} a_{ijp}) + c^{\text{retalho}}(L_j - \sum_{i \in I} a_{ijp} l_i)$, tal que c^{processo} é um custo associado ao processo de produção;
3. c^{retalho} representa o custo de retalho, que pode ser c^{sobra} ou c^{perda} , onde o primeiro representa o custo de sobra e o segundo denota o custo de perda, ambos por unidade de comprimento;
4. x_{jp} é a variável que representa o número de vezes que o padrão de corte de índice p no objeto de índice j foi aplicado;
5. a_{ijp} é a quantidade de vezes que o item de comprimento l_i é cortado no objeto de índice j no padrão de corte de índice p ;
6. d_i é a demanda de cada item de índice i .

Neste modelo, a função objetivo (4.1) minimiza o custo do processo de produção; a restrição (4.2) garante que a demanda seja satisfeita; a restrição (4.3) estabelece a condição de integralidade das variáveis. O processo de geração de colunas foi dividido em dois grupos de geradores, apresentados a seguir: um que produz padrões com perdas aceitáveis e o outro que produz padrões com sobras.

$$\text{maximizar } Z_j^{perda} = \sum_{i \in I} (\pi_i + c^{perda} l_i - c^{processo}) a_{ij} - c^{perda} L_j \quad (4.4)$$

$$\text{sujeito a } \sum_{i \in I} a_{ij} l_i \leq L_j \quad (4.5)$$

$$\sum_{i \in I} a_{ij} l_i \geq L_j - L_{perda_j} \quad (4.6)$$

$$\sum_{i \in I} a_{ij} \geq 1 \quad (4.7)$$

$$a_{ij} \leq d_i, \quad i \in I \quad (4.8)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in I, j \in J \quad (4.9)$$

$$\text{maximizar } Z_j^{sobra} = \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ij} - c^{sobra} L_j \quad (4.10)$$

$$\text{sujeito a } \sum_{i \in I} a_{ij} l_i \leq L_j - L_{sobra_j} \quad (4.11)$$

$$\sum_{i \in I} a_{ij} l_i \geq L_j - L_{sobra_j}^{max} \quad (4.12)$$

$$\sum_{i \in I} a_{ij} \geq 1 \quad (4.13)$$

$$a_{ij} \leq d_i, \quad i \in I \quad (4.14)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in I, j \in J \quad (4.15)$$

Onde:

1. a_{ij} são as variáveis que representam o número de itens l_i cortados do objeto de índice j ;
2. L_{perda_j} representa um comprimento de perda definido como aceitável no j -ésimo objeto;
3. L_{sobra_j} representa o comprimento mínimo aceito como sobra no j -ésimo objeto;
4. π_i são as coordenadas do multiplicador simplex;
5. $L_{sobra_j}^{max}$ é o comprimento máximo possível definido como sobra; $i \in I$; $j \in J$.

As restrições (4.5) e (4.6) garantem que o retalho existente no padrão de corte gerado esteja em um intervalo de perda aceitável, enquanto as restrições (4.11) e (4.12) garantem um retalho em um intervalo de sobra; as restrições (4.7) e (4.13) estabelecem que pelo menos um item seja cortado em cada padrão; as restrições (4.8) e (4.14) estabelecem que a repetição de cada item no padrão de corte gerado não supere a sua demanda, evitando assim a produção de itens excedentes em um único padrão de corte; e as restrições (4.9) e (4.15) fornecem valores inteiros não-negativos às variáveis. A determinação dos valores dos parâmetros L_{perda_j} , L_{sobra_j} e $L_{sobra_j}^{max}$ possibilita escolher quais objetos concentrarão perdas aceitáveis ou sobras.

Em muitos problemas práticos, entretanto, existe uma limitação de estoque que deve ser considerada. Se esta limitação implicar em o número de objetos disponíveis menor que os tipos de itens, a utilização dos padrões homogêneos maximais não é imediata, pois com os objetos disponíveis não se consegue a base inicial trivial para o Método Simplex. Para resolver este tipo de problema, são propostas três outras abordagens.

4.2 ABORDAGEM M2P&R

Nesta segunda abordagem, são utilizadas novas restrições relacionadas à limitação de objetos em estoque e à produção de itens em excesso. Os geradores de padrões também são modificados, conforme é mostrado a seguir.

4.2.1 Modelagem Matemática Proposta II

Deste modo, o modelo pode ser escrito como segue,

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (4.16)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i, \quad i \in I \quad (4.17)$$

$$\sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \leq \left(\frac{A}{100} + 1\right) d_i, \quad i \in I \quad (4.18)$$

$$\sum_{p \in P(j)} x_{jp} \leq e^j, \quad j \in J \quad (4.19)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, \quad p \in P(j) \quad (4.20)$$

A restrição (4.18) foi adicionada ao modelo para que não sejam produzidos muitos itens em excesso e, conforme a limitação estabelecida pelo planejador; a restrição (4.19) estabelece que não seja utilizado um número maior de objetos do que o disponível em estoque.

Para controlar a produção dos itens nos padrões gerados, o parâmetro *qteItemMax* foi criado. Os subproblemas geradores de padrões são apresentados a seguir.

$$\text{maximizar } Z_j^{perda} = \sum_{i \in I} (\pi_i + c^{perda} l_i - c^{processo}) a_{ijp} - c^{perda} L_j \quad (4.21)$$

$$\text{sujeito a } \sum_{i \in I} a_{ij} l_i \leq L_j \quad (4.22)$$

$$\sum_{i \in I} a_{ij} l_i \geq L_j - L_{perda_j} \quad (4.23)$$

$$\sum_{i \in I} a_{ij} \geq 1 \quad (4.24)$$

$$a_{ij} \leq qteItemMax_i, \quad i \in I \quad (4.25)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in I, j \in J \quad (4.26)$$

$$\text{maximizar } Z_j^{sobra} = \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ijp} - c^{sobra} L_j \quad (4.27)$$

$$\text{sujeito a } \sum_{i \in I} a_{ij} l_i \leq L_j - L_{sobra_j} \quad (4.28)$$

$$\sum_{i \in I} a_{ij} l_i \geq L_j - L_{sobra_j}^{max} \quad (4.29)$$

$$\sum_{i \in I} a_{ij} \geq 1 \quad (4.30)$$

$$a_{ij} \leq qteItemMax_i, \quad i \in I \quad (4.31)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in I, j \in J \quad (4.32)$$

Onde:

1. *qteItemMax_i*: é número máximo de itens de índice *i* que pode ser utilizado no padrão de corte gerado.

Ao invés de resolver dois problemas geradores de padrões para cada objeto em estoque, a próxima abordagem visa resolver apenas dois problemas geradores em cada iteração. Os objetos em estoque utilizados fazem parte das soluções apresentadas pelos geradores.

4.3 ABORDAGEM M3P&R

Nesta abordagem, também são utilizadas restrições relacionadas à limitação de objetos em estoque e à produção de itens em excesso. O número de geradores de padrões de corte independe da quantidade de tipos de objetos disponíveis em estoque. São resolvidos, a cada iteração, dois geradores: um que produz padrões com perdas aceitáveis e o outro que produz padrões com sobras. Ambos indicando qual objeto em estoque deverá ser utilizado.

Logo, um padrão de corte para esta abodagem é um vetor com $m + M$ coordenadas: $a_{jp} = (a_{1jp}, \dots, a_{mjp}, a_{(m+1)jp}, \dots, a_{(m+M)jp})$, em que suas m primeiras coordenadas representam os itens cortados e as demais coordenadas indicam qual objeto foi utilizado. Assim, apenas uma entre as componentes $a_{(m+1)jp}, \dots, a_{(m+M)jp}$ do vetor que representa um padrão de corte é diferente de zero: $a_{ijp} = (a_{1jp}, \dots, a_{mjp}, 0, \dots, 0, 1, 0, \dots, 0)$, com 1 na $(m + j)$ -ésima posição.

4.3.1 Modelagem Matemática Proposta III

A terceira abordagem com a atual definição de padrão de corte é apresentada a seguir.

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (4.33)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i, \quad i \in I \quad (4.34)$$

$$\sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \leq \left(\frac{A}{100} + 1\right) d_i, \quad i \in I \quad (4.35)$$

$$\sum_{p \in P(j)} a_{(m+j)jp} x_{jp} \leq e^j, \quad j \in J \quad (4.36)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, \quad p \in P(j) \quad (4.37)$$

Onde:

1. $c_{jp}(a_{jp}) = c^{\text{processo}} \left(\sum_{i \in I} a_{ijp} \right) + c^{\text{retalho}} \left(\sum_{i \in J} L_i a_{(i+m)jp} - \sum_{i \in I} a_{ijp} l_i \right)$, tal que c^{processo} é um custo associado ao processo de produção;
2. c^{retalho} representa o custo de retalho, que pode ser c^{sobra} ou c^{perda} , onde o primeiro representa o custo de sobra e o segundo denota o custo de perda, ambos por unidade de comprimento;
3. x_{jp} é a variável que representa o número de vezes que o padrão de corte de índice p no objeto de índice j foi aplicado;
4. a_{ijp} é a quantidade de vezes que o item de comprimento l_i é cortado no objeto de índice j no padrão de corte de índice p para $i \in I$ e o objeto utilizado é indicado no índice $m + j$;
5. d_i é a demanda de cada item de índice i .

A função objetivo (4.33) minimiza o custo do processo de produção. As restrições (4.34) e (4.35) garantem que a quantidade de itens produzidos seja menor que uma quantidade a mais da demanda estipulada pelo planejador e maior ou igual à demanda. A restrição (4.36) garante que os objetos a serem utilizados no processo de corte estão

disponíveis. Por fim, a restrição (4.37) define as variáveis do problema. Os problemas geradores de padrões são apresentados a seguir.

$$\text{maximizar } Z^{perda} = \sum_{i \in I} (\pi_i + c^{perda} l_i - c^{processo}) a_i + \sum_{i \in J} (\pi_{i+m} - c^{perda} L_i) a_{i+m} \quad (4.38)$$

$$\text{sujeito a } \sum_{i \in I} a_i l_i \leq \sum_{i \in J} L_i a_{i+m} \quad (4.39)$$

$$\sum_{i \in I} a_i l_i \geq \sum_{i \in J} (L_i - L_{perda_i}) a_{i+m} \quad (4.40)$$

$$\sum_{i \in I} a_i \geq 1 \quad (4.41)$$

$$\sum_{i \in J} a_{i+m} = 1 \quad (4.42)$$

$$a_i \leq qteItemMax_i \quad (4.43)$$

$$a_i \geq 0 \text{ e inteiro} \quad (4.44)$$

$$\text{maximizar } Z^{sobra} = \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_i + \sum_{i \in J} (\pi_{i+m} - c^{sobra} L_i) a_{i+m} \quad (4.45)$$

$$\text{sujeito a } \sum_{i \in I} a_i l_i \leq \sum_{i \in J} (L_i - L_{sobra_i}) a_{i+m} \quad (4.46)$$

$$\sum_{i \in I} a_i l_i \geq \sum_{i \in J} (L_i - L_{sobra_i}^{max}) a_{i+m} \quad (4.47)$$

$$\sum_{i \in I} a_i \geq 1 \quad (4.48)$$

$$\sum_{i \in J} a_{i+m} = 1 \quad (4.49)$$

$$a_i \leq qteItemMax_i \quad (4.50)$$

$$a_i \geq 0 \text{ e inteiro} \quad (4.51)$$

Onde:

1. a_i são as variáveis que representam o número de itens l_i cortados para $i \in I$ e para $i = \{m + 1, \dots, m + M\}$ indica o objeto utilizado;
2. L_{perda_j} representa um comprimento de perda definido como aceitável no j -ésimo objeto;
3. L_{sobra_j} representa o comprimento mínimo aceito como sobra no j -ésimo objeto;
4. π_i são as coordenadas do multiplicador simplex, $i = 1, \dots, m + M$;
5. $L_{sobra_j}^{max}$ é o comprimento máximo possível definido como sobra; $j \in J$.

Na última abordagem, apresentada a seguir, supõe-se que a demanda seja suficientemente grande e que o problema envolve várias dezenas de itens a serem produzidos. Diferentemente das abordagens anteriores, o processo de resolução se faz em duas etapas. Na

primeira etapa a solução encontrada, caso não seja inteira, é arredondada. Se esta solução for infactível, na segunda parte é resolvido um problema residual.

4.4 ABORDAGEM M4P&R

Conforme discutido anteriormente, a solução contínua do problema relaxado é uma boa solução aproximada para o problema original, quando a demanda é suficientemente grande e exige uma alta repetição dos padrões de corte. Assim, esta modelagem visa resolver problemas suficientemente grandes, considerando limitação de estoque, obtendo-se uma solução contínua através do Método Simplex com Geração de Colunas, e posteriormente a solução inteira do problema arredondando a solução contínua para o inteiro mais próximo. Se a solução inteira for inviável para o problema de PLI proposto, isto é, ainda faltar itens a serem produzidos, um problema residual é escrito. Espera-se que este problema residual tenha poucos itens e demanda bem menor que a do problema inicial já que quando a demanda é suficientemente grande a solução contínua é uma boa solução aproximada. Assim, o problema residual é solucionado utilizando a abordagem M2P&R, que mantém as características de aceitação de um padrão, determinadas pelo planejador de produção na etapa inicial.

4.4.1 Modelagem Matemática Proposta IV

Esta abordagem foi desenvolvida para solucionar problemas com demanda grande, com o objetivo de diminuir o tempo de resolução. O Problema é estruturado em duas etapas. A primeira etapa é semelhante à abordagem M1P&R, porém com a limitação de estoque. O modelo é apresentado a seguir.

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (4.52)$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i, \quad i \in I \quad (4.53)$$

$$\sum_{p \in P(j)} x_{jp} \leq e^j, \quad j \in J \quad (4.54)$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, \quad p \in P(j) \quad (4.55)$$

Os geradores de colunas desta abordagem continuam sendo os mesmos da estratégia M1P&R.

Neste caso, de demandas grandes, para resolução do problema apresentado, a solução contínua encontrada na primeira etapa é arredondada. Porém, arredondando a solução contínua pode ocorrer da demanda não ser satisfeita completamente, e um número menor de itens em quantidade relativamente pequena deixam de ser produzidos. Então, a segunda etapa visa determinar novos cortes de forma a completar a demanda, utilizando os objetos ainda disponíveis não utilizados na primeira etapa. Essa segunda parte é idêntica à abordagem M2P&R.

Um diferencial dessas abordagens está na eficiente possibilidade de se eleger quais objetos concentrarão perdas aceitáveis ou sobras.

CAPÍTULO 5 SIMULAÇÕES

As quatro abordagens propostas foram implementadas, além do modelo de Scheithauer *et al.* [31]. Os exemplos utilizados nos testes foram extraídos da dissertação de mestrado de Pinto [27] e do trabalho de Koch *et al.* [20].

Conforme apresentado em Pinto [27], seis exemplos foram obtidos de uma carteira da indústria de aeronaves que gerou o trabalho de Abuabara [1] e um exemplo foi extraído de Sinuany-Stern e Weiner [33]. Koch *et al.* [20] apresentou um exemplo que foi utilizado neste trabalho sem a limitação de estoque. O exemplo sem limitação foi testado utilizando a abordagem M1P&R e o modelo de Scheithauer, pois estes não consideram limitação de estoque. Os demais exemplos foram testados utilizando as abordagens M2P&R e M3P&R. Os resultados obtidos foram comparados com os resultados fornecidos pelo algoritmo de Sinuany-Stern e Weiner [33], pelo modelo de Abuabara e pelas heurísticas de Cherri [7]¹, apresentados em Pinto [27].

Como descrito em Pinto [27], devido a um erro ocorrido nos códigos implementados por Cherri [7], algumas de suas heurísticas falharam ao obter as soluções dos exemplos selecionados. Desta forma, o termo “Código falhou” é utilizado na tabela para indicar que a solução não pode ser encontrada.

A abordagem M4P&R não foi utilizada nos testes pois as demandas não eram grandes o suficiente para que a solução contínua fosse maior que 1, e alterasse os dados que seriam enviados para a segunda parte da abordagem, a estratégia M2P&R.

Todas as implementações foram realizadas no software de otimização XPRESS-MP, utilizando a linguagem Mosel. Os testes foram realizados em um computador Intel Core 2 duo de 3 GHz e 4 GB de RAM.

Para dar início ao processo de corte alguns dados são fornecidos pelo planejador de produção, em cada uma das abordagens. Entre estes dados estão os comprimentos dos objetos (L_j), dos itens (l_i) e a demanda (d_i). A disponibilidade de objetos em estoque é fornecida quando no modelo há limitação de estoque (e^j). Outros parâmetros mais específicos são informados, como por exemplo, o valor associado a sobra (c^{sobra}), em unidade de comprimento, produzida pelo processo de corte, ou ainda, os intervalos dos retalhos

¹Heurística FFD_R , Heurística $Gulosa_R$, Heurística Residual FFD_R , Heurística Residual $Gulosa_R$ e Heurística RAG_R (versões 1, 2 e 3).

A serem considerados perda $([0, L_{perda_j}])$ ou sobra $([L_{sobra_j}, L_{sobra_j}^{max}])$. O planejador, ainda, tem liberdade de escolher um ou mais objetos, que ele preferir, para conter as perdas ou as sobras. Diante da possibilidade de alteração dos parâmetros, algumas soluções foram apresentadas fornecendo ao planejador possibilidades de escolha da solução que lhe for mais conveniente.

Na resolução das abordagens, é utilizado uma aplicação modificada do Método Simplex com Geração de Colunas descrito na seção 2.4, em que são resolvidos vários subproblemas lineares inteiros para geração de padrões com sobra e com perda em cada um dos objetos em estoque. Em cada iteração, são escolhidas as colunas relativas aos subproblemas com soluções ótimas para serem inseridos no PM restrito relaxado. Embora em geral apenas uma destas colunas melhoram a solução contínua (máximo entre Z_j^{perda} ou Z_j^{sobra}), as demais colunas são úteis pois melhoraram a solução inteira do Problema (4.1.1), quando resolvido por Branch-and-Bound. Cada coluna introduzida é um padrão de corte que possui uma perda aceitável ou uma sobra. Logo, os modelos determinam quais objetos e padrões são mais vantajosos de acordo com os dados de entrada fornecidos pelo planejador de produção. Uma síntese dos processos de resolução das abordagens é mostrada no final do capítulo por meio de fluxogramas. O algoritmo utilizado na implementação da abordagem M1P&R é mostrado na Figura 5.1.

Para iniciar o Método Simplex com Geração de Colunas, é necessária uma solução básica viável inicial, ou seja, uma solução não negativa cuja matriz básica associada é não singular (invertível). Assim, são criados padrões de corte homogêneos maximais. Estes padrões correspondem à maneira trivial de cortar os objetos para atender à demanda e de obter a matriz básica inicial. Para isto, considera-se sempre que a quantidade de objetos disponíveis em estoque é suficiente. Então o processo de geração de colunas é efetuado até que todos os subproblemas geradores de colunas apresentem solução menores ou iguais a zero, ou apresente solução infactível. Ao final do processo, uma solução inteira é obtida pelo método Branch-and-Bound.

Na implementação da abordagem M2P&R, foi utilizado o algoritmo mostrado no fluxograma da Figura 5.2. O processo se inicia com padrões de corte heterogêneos maximais, a obtenção da matriz básica inicial e termina quando nenhuma nova coluna que melhore a solução contínua pode ser encontrada, da mesma forma que na abordagem anterior. Inicia-se com padrões heterogêneos pois admitiu-se que a quantidade de objetos disponíveis é menor que os tipos de itens, excluindo a possibilidade de utilizar a

maneira trivial de obter a base inicial do Método Simplex. Para a obtenção de uma solução viável inicial, cortou-se todos os itens, misturados, na quantidade desejada nos objetos existentes, considerando que os objetos disponíveis em estoque eram suficientes. Fornecido estes padrões ao otimizador XPRESS-MP o mesmo obtém uma solução básica viável inicial, ou seja, a matriz básica inicial e inicia o Método Simplex com Geração de Colunas.

A cada iteração, são gerados padrões de corte em que a repetição de cada item é limitada pelo parâmetro *qteItemMax*. Assim, no início do processo são gerados padrões com pouca repetição de itens, o que proporciona a utilização de padrões que contribuem no controle à produção de itens excedentes. Quando nenhuma coluna que melhore a solução atual pode ser encontrada, o parâmetro *qteItemMax* é modificado, tendo como limitante superior a demanda.

Ainda, em cada iteração é analisado se o modelo forneceu uma solução inteira com itens excedentes. Se este for o caso, novos padrões são gerados a partir dos padrões utilizados na solução inteira atual, os quais são denominados subpadrões. Um subpadrão é calculado a partir de um padrão que corta itens em excesso. Então itens excedentes são retirados, visando manter o intervalo de aceitação dos padrões, isto é, mantendo o retalho em um intervalo de perda aceitável ou sobra.

Quando a retirada de itens implicar em padrões não aceitáveis, os subpadrões correspondentes não são gerados. Os subpadrões criados são então inseridos no problema inicial e o processo de gerar colunas continua até que a melhor solução seja encontrada. O cálculo dos subpadrões é feito pois estes fornecem mais opções para melhorar a solução inteira, ao proporcionarem uma produção final com menos itens excedentes. Porém, estes não são obtidos pelas mochilas porque não são as melhores opções para o Método Simplex, para o cálculo da solução contínua.

Já na abordagem M3P&R, o processo se inicia e termina do mesmo modo que a abordagem M2P&R. A cada iteração são gerados padrões de corte em que a repetição de cada item é limitada pelo parâmetro *qteItemMax*, e também são criados subpadrões, quando possível. Uma ilustração do processo é mostrada no fluxograma da Figura 5.3.

Quanto à abordagem M4P&R, a qual é realizada em duas etapas, o algoritmo utilizado na implementação é mostrado na Figura 5.4. Inicialmente, uma solução inteira é obtida em uma primeira etapa, a qual é análoga à primeira abordagem, porém através do arredondamento da solução contínua para o valor inteiro mais próximo. Em seguida, a parcela da demanda que falta ser atendida e o restante de objetos disponíveis em estoque, os quais não foram utilizados na primeira etapa, são utilizados em uma nova etapa como dados

de entrada para a abordagem M2P&R, a qual fornece uma solução inteira conforme descrito anteriormente. Dessa forma, uma solução inteira final é obtida, correspondente aos padrões de corte e aos objetos utilizados nas duas etapas de resolução.

5.1 ALGORITMOS IMPLEMENTADOS

Nesta sessão são apresentados os algoritmos utilizados nas implementações das quatro abordagens propostas, na forma de fluxogramas.

Figura 5.1 – Fluxograma da abordagem M1P&R

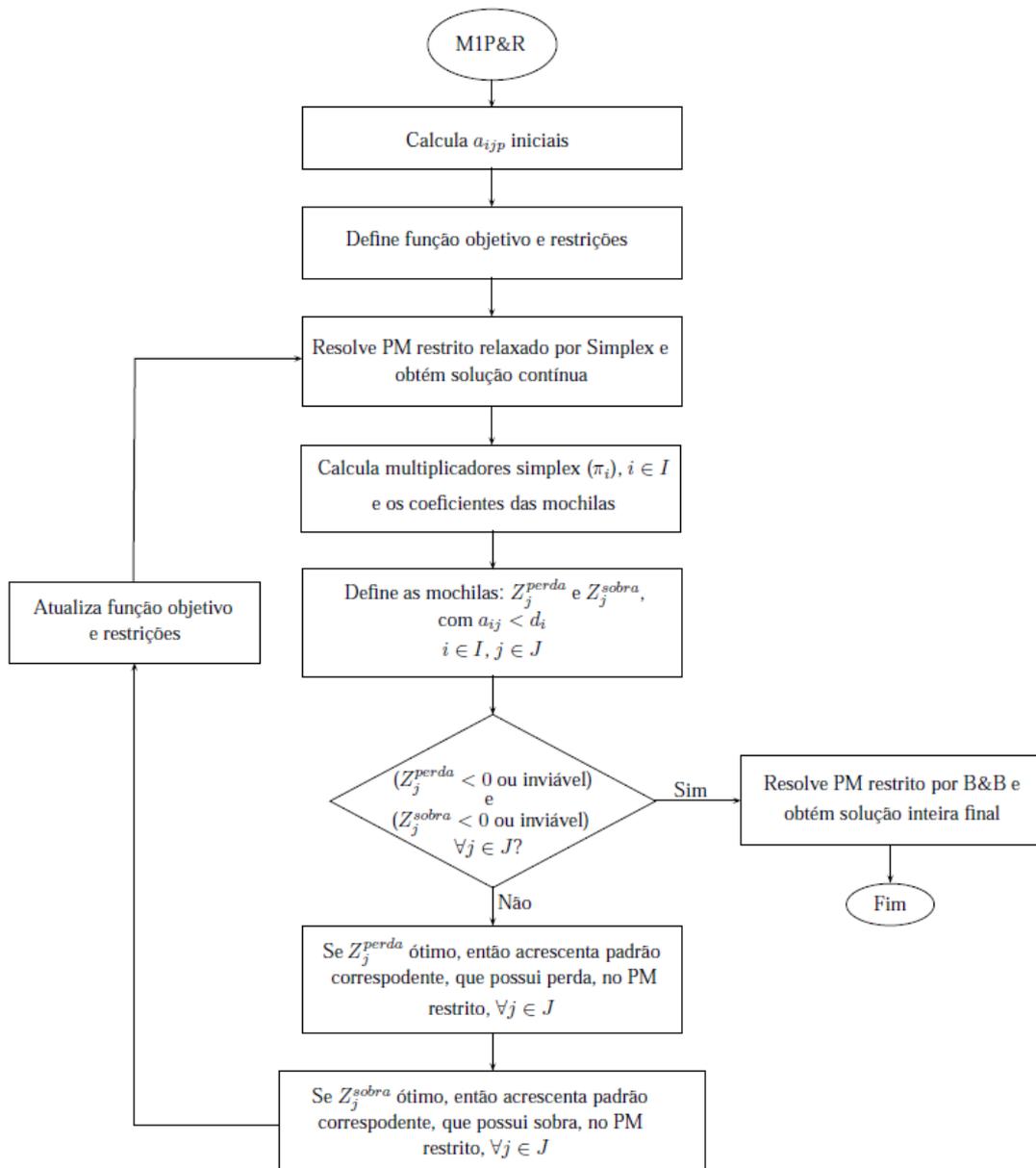


Figura 5.2 – Fluxograma da abordagem M2P&R

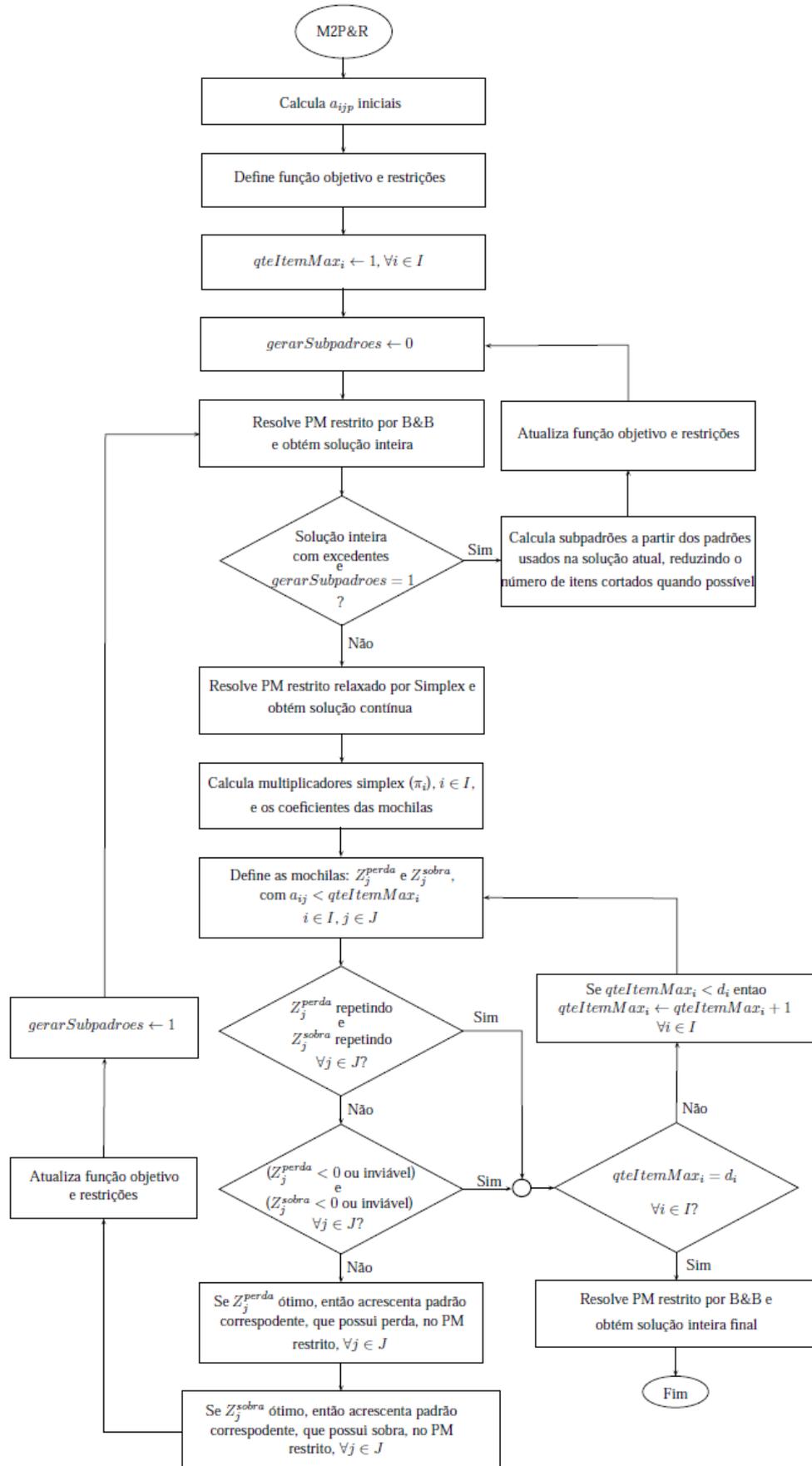


Figura 5.3 – Fluxograma da abordagem M3P&R

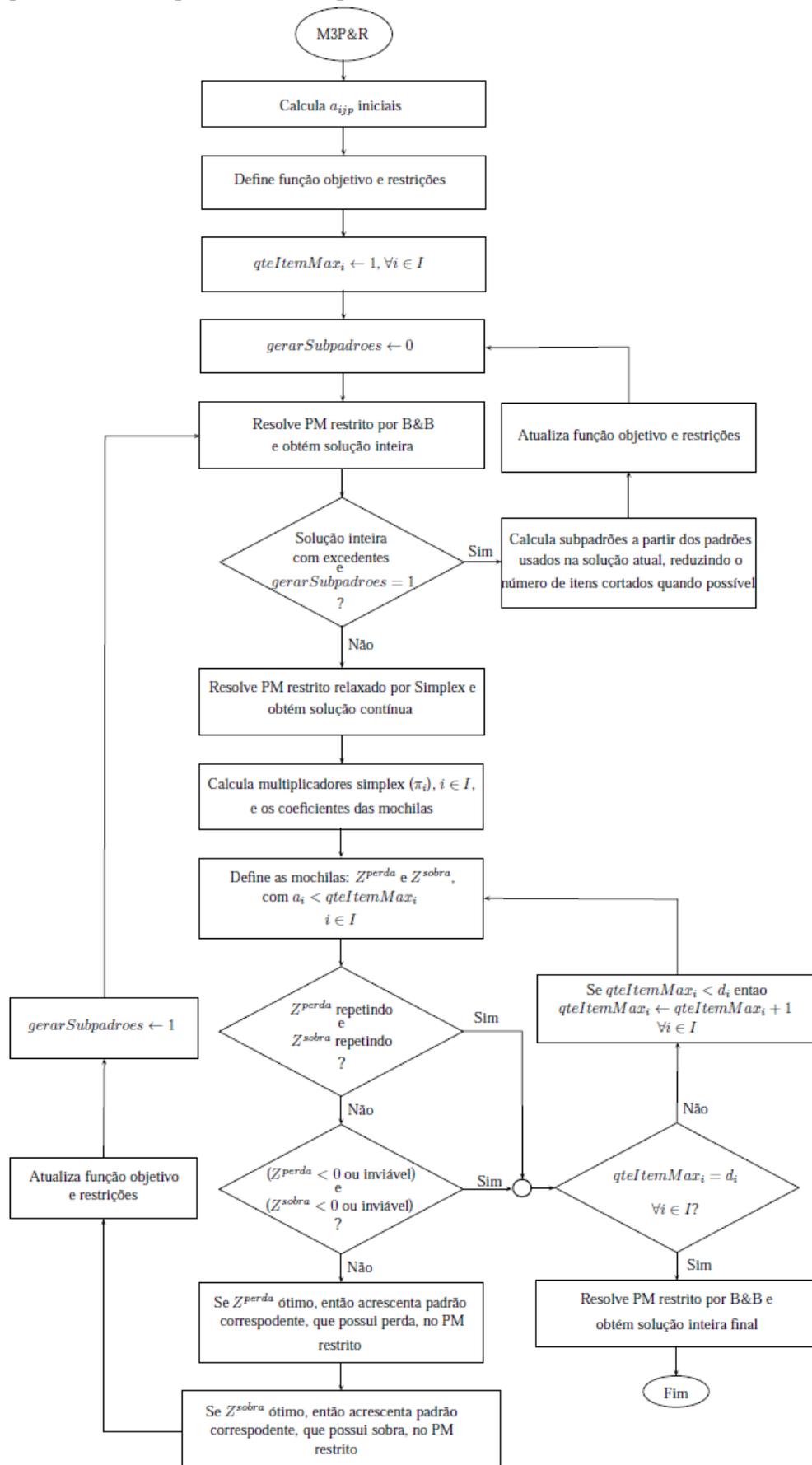
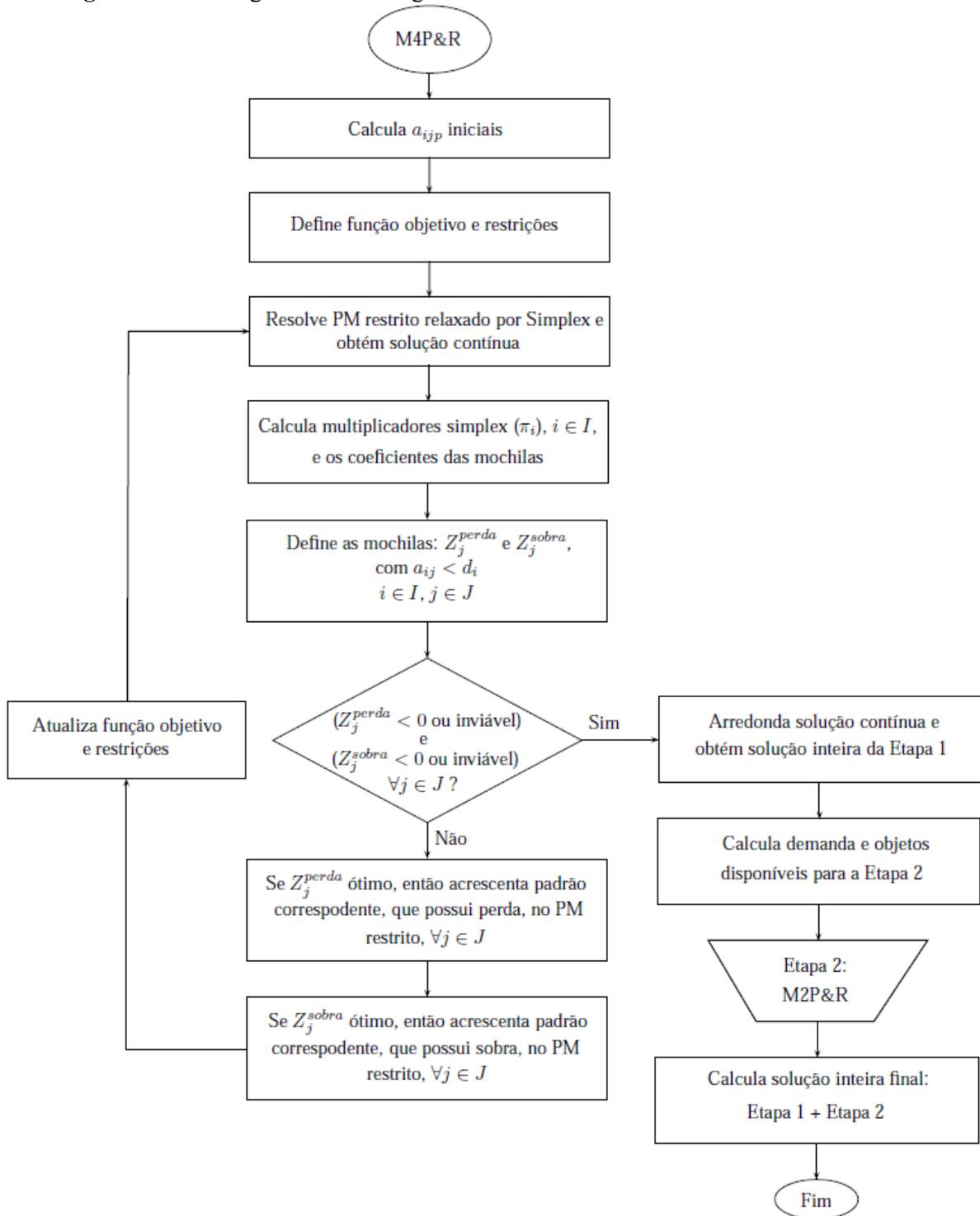


Figura 5.4 – Fluxograma da abordagem M4P&R



CAPÍTULO 6

RESULTADOS

Neste capítulo, são apresentados os resultados dos testes realizados com três abordagens propostas, os quais são comparados com as soluções das demais modelagens estudadas. Os dados dos exemplos estão apresentados no Apêndice B. Os exemplos são divididos em duas categorias: com e sem limitação de estoque. Os resultados obtidos são apresentados à seguir.

6.1 PROBLEMA SEM LIMITAÇÃO DE ESTOQUE

Este exemplo foi obtido do trabalho de Koch *et al.* [20], porém considerado sem limitação de estoque. Este problema é composto por 16 tipos de objetos em estoque, e 6 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.1 e comparados com o modelo de Scheithauer [31].

Tabela 6.1 – Solução do exemplo sem limitação

	Comp. total	N. de objetos (com)				Sobra	Perda
		cortados	aproveitamento total	sobra	perda		
MIP&R	123952	6	0	4	2	23110	262
Scheithauer	104564	7	0	0	7	0	3984

Pode-se verificar que o resultado obtido com a abordagem MIP&R utiliza um comprimento total cerca de 18,54% maior do que ao utilizado pelo modelo Scheithauer *et al.* [31]. Entretanto, a abordagem proposta utiliza um objeto a menos, e o percentual de perda em relação ao comprimento total cortado é de 0,21%, contra 3,81% do Scheithauer *et al.* [31]. Nota-se também que os comprimentos de sobras informados ao modelo de Scheithauer têm uma influência significativa no resultado.

Em modelos resolvidos pelo Método Simplex com Geração de Colunas, em que a quantidade de material disponível é suficiente para atender à demanda, e todo tipo de objeto está disponível enquanto é vantajoso para o processo de corte, objetos com comprimentos maiores são utilizados mais vezes, pois é possível cortar mais itens e de diversas maneiras, que implica em perdas menores ou sobras mais vantajosas.

6.2 PROBLEMAS COM LIMITAÇÃO DE ESTOQUE

Os exemplos foram obtidos da dissertação de mestrado de Pinto [27]. Os resultados das simulações são apresentados em tabelas.

6.2.1 Exemplo 1

Este problema é composto por 3 tipos de objetos em estoque, e 9 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.2¹.

Tabela 6.2 – Solução do Exemplo 1.

	Comp. total	N. de objetos (com)			Sobra	Perda	
		cortados	aproveitamento total	sobra perda			
M2P&R	12000	3	0	2	1	1145	1
M3P&R	12000	3	0	2	1	1145	1
Estratégia1	12000	3	0	1	2	685	461
Estratégia2	12000	3	0	1	2	1135	11
Estratégia3	12000	3	0	1	2	1076	70
Estratégia4	12000	3	0	1	2	620	526
Estratégia5	12000	3	0	1	2	1135	11
Sinuany	13000	4	0	1	3	2075	71
Abuabara	12000	3	0	1	2	1135	11
FFD_R	12000	3	0	2	1	1140	6
$Gulosa_R$	13000	4	0	2	2	2135	11
$R.FFD_R$	12000	3	0	2	1	1140	6
$R.Gulosa_R$	13000	4	0	2	2	2135	11
RAG_R 1	13000	4	0	2	2	2130	16
RAG_R 2	12000	3	0	1	2	1128	18
RAG_R 3	13000	4	0	2	2	2130	16

Neste exemplo, observa-se que todas as abordagens apresentaram soluções com perdas. Entretanto, observa-se que as abordagens M1P&R e M2P&R propostas foram as que apresentaram menor comprimento de perda, 1 u.c., com uma redução de 83,33% em relação ao valor fornecido pelas abordagens com segundo menor comprimento de perda

¹As estratégias 1, 2, 3, 4 e 5 são do trabalho de Pinto [27], Sinuany de Sinuany e Weiner [33], Abuabara de Abuabara [1] e as heurísticas FFD_R , $Gulosa_R$, $R.FFD_R$, $R.Gulosa_R$, RAG_R 1, RAG_R 2, RAG_R 3 de Cherri [7]

(FFD_R e $R.FFD_R$, com 6 u.c.). As Estratégias 2, 5, e Abuabara apresentaram a mesma solução, com 11 u.c. de perda. Com o mesmo comprimento de perda, porém com um objeto a mais cortado, estão $Gulosa_R$ e $R.Gulosa_R$. As demais estratégias apresentaram soluções com perdas maiores.

6.2.2 Exemplo 2

O problema tratado no exemplo 2 é composto por 3 tipos de objetos em estoque, e 4 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.3.

Tabela 6.3 – Solução do Exemplo 2.

	Comp. total	N. de objetos (com)				Sobra	Perda
		cortados	aproveitamento total	sobra	perda		
M2P&R	17394	4	2	1	1	368	1
M3P&R	18408	3	2	1	0	1383	0
Estratégia1	17374	4	0	1	3	204	145
Estratégia2	17374	4	1	1	2	347	2
Estratégia3	17374	4	1	1	2	347	2
Estratégia4	17374	4	0	1	3	201	148
Estratégia5	17374	4	1	1	2	347	2
Sinuany	19644	6	2	1	3	2616	3
Abuabara	18408	3	2	1	0	1383	0
FFD_R	21783	6	1	1	4	4744	14
$Gulosa_R$	19644	6	2	1	3	2615	4
$R.FFD_R$	21783	6	1	1	4	4744	14
$R.Gulosa_R$	19644	6	2	1	3	2615	4
RAG_R 1	18499	5	1	3	1	1473	1
RAG_R 2	18499	5	2	3	0	1474	0
RAG_R 3	18499	5	1	3	1	1473	1

As soluções do Exemplo 2 obtidas pelas Estratégias 1 e 4 apresentaram grandes perdas, já as Estratégias 2, 4, e 5 apresentaram perdas de 2u.c. concentradas em dois objetos. Contudo, a abordagem M2P&R forneceu uma perda de 1u.c. concentrada em um único objeto, utilizando, apenas, um comprimento total cerca de 0,11% maior do que ao utilizado pelos modelos com o menor comprimento cortado. Ainda, a abordagem M3P&R, e o modelo de Abuabara apresentaram uma solução, cortando cerca de 5,95% a mais do menor

comprimento utilizado, sem perda alguma de material, e com sobra concentrada em um objeto. As demais abordagens produziram perdas superiores ou distribuíram suas sobras em mais de um objeto.

6.2.3 Exemplo 3

O exemplo 3 possui 2 tipos de objetos disponíveis, e 4 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.4.

Tabela 6.4 – Solução do Exemplo 3.

	Comp. total	N. de objetos (com)				Sobra	Perda	
		cortados	aproveitamento total	sobra	perda			
M2P&R	3306	11	2	5	4	535	26	
M3P&R	3072	10	2	3	5	287	40	
Estratégia1	2838	9	0	1	8	34	59	
Estratégia2	2838	9	2	1	6	51	42	
Estratégia3	2838	9	2	1	6	48	45	
Estratégia4	2838	9	0	1	8	21	72	
Estratégia5	2838	9	1	1	7	51	42	
Sinuany	3306	11	0	4	7	504	57	
Abuabara	2838	9	2	1	6	51	42	
FFD_R			Solução inviável					
$Gulos_R$			Solução inviável					
$R.FFD_R$	2838	9	0	1	8	40	53	
$R.Gulos_R$	2838	9	0	1	8	40	53	
$RAG_R 1$			Solução inviável					
$RAG_R 2$			Solução inviável					
$RAG_R 3$			Solução inviável					

A solução apresentada pela abordagem M2P&R é a solução com menor comprimento de perda gerada, menos que 1% de material é perdido. Além disso, a abordagem M2P&R, juntamente com a M3P&R geram sobras que fornecem maiores possibilidades de reuso, pois possuem comprimentos maiores, mesmo sendo distribuídas em mais de um objeto. As Estratégias de 1 a 5 e as heurísticas que apresentaram resultado, cortam o menor comprimento, porém apenas as Estratégias 2 e 5, juntamente com o modelo de Abuabara [1] possuem uma perda de aproximadamente 1,47% do comprimento utilizado. As demais abordagens possuem perdas superiores.

6.2.4 Exemplo 4

Já no exemplo 4, tem-se 3 tipos de objetos em estoque, e 23 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.5.

Tabela 6.5 – Solução do Exemplo 4.

	Comp. total	N. de objetos (com)				Sobra	Perda
		cortados	aproveitamento total	sobra	perda		
M2P&R	16000	5	2	1	2	3190	5
M3P&R	15500	4	4	0	0	0	0
Estratégia1	13000	4	0	2	2	189	6
Estratégia2	13000	4	3	1	0	195	0
Estratégia3	13000	4	3	1	0	195	0
Estratégia4	13000	3	0	2	1	193	2
Estratégia5	13000	3	2	1	0	195	0
Sinuany	16000	5	4	1	0	3195	0
Abuabara	13000	4	3	1	0	195	0
FFD_R	16000	5	2	1	2	3190	5
$Gulos_R$	16000	5	4	1	0	3195	0
$R.FFD_R$			Código falhou				
$R.Gulos_R$			Código falhou				
$RAG_R 1$			Código falhou				
$RAG_R 2$			Código falhou				
$RAG_R 3$			Código falhou				

Os dados da Tabela 6.5 mostram que as soluções obtidas pelas Estratégias 4 e 5 são as que menos cortam objetos, sendo que a última não produz perda. Os resultados obtidos pelas Estratégias 2 e 3 são iguais aos obtidos pelo modelo simplificado de Abuabara [1]. Quanto às heurísticas propostas por Cherri et al. [7], e ao algoritmo iterativo proposto por Sinuany-Weiner [33], o comprimento total cortado foi superior ao obtido nas demais soluções, apesar de duas destas não gerarem perdas. Já a Estratégia 1, considerando o critério de menor perda, apresentou o pior resultado entre as abordagens que forneceram solução. A estratégia proposta M2P&R apresentou o mesmo resultado da estratégia FFD_R . Já a abordagem M3P&R apresentou um resultado interessante, em que todos os objetos tiveram aproveitamento total. Como consequência, itens excedentes foram produzidos. Este resultado pode ser considerado satisfatório para aplicações em que a produção de itens seja contínua, de

tal forma que estes itens excedentes possam ser utilizados para atender uma demanda futura, o que é muito comum em diversos segmentos industriais.

6.2.5 Exemplo 5

Na Tabela 6.6, são apresentados os resultados das simulações feitas para um problema composto por 3 tipos de objetos em estoque, e 38 tipos de itens distintos a serem cortados.

Tabela 6.6 – Solução do Exemplo 5.

	Comp. total	N. de objetos (com)				Sobra	Perda	
		cortados	aproveitamento total	sobra	perda			
M2P&R	23000	6	3	1	2	184	41	
M3P&R	23000	6	3	1	2	184	41	
Estratégia1	23000	6	1	1	4	137	88	
Estratégia2	23000	6	5	1	0	225	0	
Estratégia3	23000	6	5	1	0	225	0	
Estratégia4	23000	6	0	1	5	139	86	
Estratégia5	23000	6	5	1	0	225	0	
Sinuany	23000	6	5	1	0	225	0	
Abuabara	23000	6	5	1	0	225	0	
FFD_R	23000	6	4	1	1	220	5	
$Gulosar$	23000	6	5	1	0	225	0	
$R.FFD_R$			Código falhou					
$R.Gulosar$			Código falhou					
$RAG_R 1$			Código falhou					
$RAG_R 2$			Código falhou					
$RAG_R 3$			Código falhou					

Neste exemplo, observa-se que todas as soluções obtidas cortam o mesmo comprimento total. A melhor solução apresentou uma perda nula e a sobra concentrada em um único objeto. Entre os modelos que apresentaram perda não nula, as abordagens M1P&R e M2P&R encontram-se em segundo lugar.

6.2.6 Exemplo 6

Este problema é composto por 3 tipos de objetos em estoque, e 7 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.7.

Tabela 6.7– Solução do Exemplo 6.

	Comp. total	N. de objetos (com)				Sobra	Perda	
		cortados	aproveitamento total	sobra	perda			
M2P&R	18000	3	1	1	1	2715	1	
M3P&R	18000	3	1	1	1	2715	1	
Estratégia1			Solução inviável					
Estratégia2			Solução inviável					
Estratégia3	15500	3	1	0	2	0	216	
Estratégia4	18000	3	0	1	2	2205	511	
Estratégia5	18000	3	2	1	0	2716	0	
Sinuanay	19000	5	0	2	3	2871	845	
Abuabara	18000	3	2	1	0	2716	0	
FFD_R	18500	4	1	2	1	3215	1	
$Gulosa_R$	18500	4	2	2	0	3216	0	
$R.FFD_R$	18500	4	1	2	1	3215	1	
$R.Gulosa_R$	18500	4	2	2	0	3216	0	
RAG_R 1	18500	4	1	2	1	3215	0	
RAG_R 2	18500	4	1	2	1	3215	0	
RAG_R 3	18500	4	1	2	1	3215	0	

Os dados da tabela 6.7 mostram que a Estratégia 3 foi a abordagem que cortou o menor comprimento total, mas foi a que forneceu apenas perda, nenhuma sobra. A abordagem de Abuabara [1] e a Estratégia 5 forneceram uma solução em que a demanda é atendida com o corte de 18000 u.c., concentrando a sobra em um objeto e não produzindo perda. Porém, as abordagens M2P&R e M3P&R, apesar de cortar o mesmo comprimento para satisfazer a demanda, produz 1 u.c. de perda e a sobra concentrada em um único objeto.

6.2.7 Exemplo 7

O exemplo 7 possui 3 tipos de objetos em estoque, e 8 tipos de itens distintos a serem cortados. Os resultados das simulações são apresentados na Tabela 6.8.

Tabela 6.8 – Solução do Exemplo 7.

	Comp. total	N. de objetos (com)				Sobra	Perda	
		cortados	aproveitamento total	sobra	perda			
M2P&R	9000	2	1	1	0	2020	0	
M3P&R	9000	2	1	1	0	2020	0	
Estratégia1			Solução inviável					
Estratégia2			Solução inviável					
Estratégia3	7000	2	1	0	1	0	20	
Estratégia4	9000	2	0	1	1	1820	200	
Estratégia5	9000	2	1	1	0	2020	0	
Sinuany	9000	3	1	1	1	2010	10	
Abuabara	9000	2	1	1	0	2020	0	
FFD_R	10000	3	1	2	0	3020	0	
$Gulosa_R$	9500	3	1	2	0	2520	0	
$R.FFD_R$	10000	3	1	2	0	3020	0	
$R.Gulosa_R$	9500	3	1	2	0	2520	0	
RAG_R 1	9000	2	1	1	0	2020	0	
RAG_R 2	9000	2	1	1	0	2020	0	
RAG_R 3	9000	2	1	1	0	2020	0	

Com base nos dados da Tabela 6.8, verifica-se que as abordagens M2P&R e M3P&R apresentaram a mesma solução dos modelos de Abuabara [1], Nova RAG_R 1, 2 e 3, e da Estratégia 5. As Estratégias 3, 4 e Sinuany são as únicas que forneceram soluções com perdas. Já as Estratégias FFD_R , $Gulosa_R$, $R.FFD_R$ e $R.Gulosa_R$ também apresentam soluções com sobras, porém com número de objetos e um comprimento total cortado superior ao das primeiras abordagens.

CONSIDERAÇÕES FINAIS

Apesar dos Problemas de Corte serem estudados desde a década de 1940, o Problema de Corte visando o reaproveitamento do retalho produzido no processo de corte começou a se desenvolver na década de 80, e pode ser aplicado em uma diversidade de casos como: em problemas com ou sem limitação de estoque; modelos que consideram movimentação de compartimentos; em problemas com mais de uma dimensão; entre outros, o que implica em uma dificuldade em definir um único modelo. Deste modo, o estudo apresentado nesse trabalho foi orientado para o PCES, sendo que os objetivos são orientados para a minimização do custo total da produção, que depende dos valores atribuídos aos custos de corte, perda e sobra. Para resolvê-los, o Método Simplex com Geração de Colunas, que é utilizado para resolver Problemas de Corte, foi alterado para incluir a possibilidade de geração de sobras. Então foram propostas quatro abordagens, todas elas tendo como principal objetivo minimizar o custo de produção. A primeira abordagem não considera limitação de estoque e, para cada objeto, duas mochilas são resolvidas para gerar padrões que são inseridos no problema mestre. Do mesmo modo, os padrões de corte são gerados na abordagem 2, porém considerando um estoque limitado. Na terceira abordagem, uma coluna (padrão) gerada contém a informação de como os itens são cortados e em qual objeto, sendo assim, apenas duas mochilas são resolvidas a cada iteração. Nestas duas últimas abordagens, o método para se resolver o problema de PLI proposto utiliza o parâmetro $qteItemMax$, que aumenta (um de cada vez) até o seu limitante, a demanda. Logo, problemas em que as demandas são muito grandes, estes procedimentos se tornam bastante lentos. Assim, a abordagem 4 foi proposta para suprir tal deficiência. Porém as demandas dos exemplos utilizados não foram suficientemente grandes para testá-la.

A Técnica de Geração de Colunas mostrou-se uma ferramenta importante por permitir resolver o problema sem necessidade de se conhecer todas as colunas a priori. Entretanto, buscando resolver o problema sem a resolução de um Problema Residual, constatou-se que a Técnica de Geração de Colunas clássica não mostra-se muito eficiente, pois ao resolver o problema de PLI envolvido, a solução apresentada não é viável e, assim, deixa de produzir muitos itens para atender a demanda. Por isso, foi utilizada nas abordagens propostas uma aplicação modificada da Técnica de Geração de Colunas tradicional, em que várias colunas são adicionadas ao Problema Mestre Restrito em cada iteração, o que apresentou melhores resultados para a solução inteira e sem necessidade de resolver um Problema Residual.

O procedimento de calcular os subpadrões permitiu a redução de itens excedentes e o surgimento de sobras, dependendo dos custos, proporcionando melhores resultados; a introdução do parâmetro *qteItemMax* nas mochilas, que faz com que o espaço de busca seja variável em cada iteração, foi importante por permitir gerar padrões de corte que mantinham o controle da produção de itens excedentes e permitiam atender à restrição de objetos em estoque, o que também melhorou a solução inteira.

Os desempenhos dos procedimentos desenvolvidos foram analisados considerando exemplares da literatura. Com relação aos resultados obtidos, quando não há limitação de estoque, observa-se que objetos de mesmo índice são utilizados até deixarem de ser lucrativos, o que não é permitido ocorrer no caso onde há limitação: nesse caso, o modelo exige a troca dos objetos quando as unidades disponíveis forem utilizadas, o que pode implicar em um aumento no valor da função objetivo e uma utilização maior de objetos se comparado ao caso anterior. De posse dos resultados obtidos com diferentes valores atribuídos aos parâmetros dos modelos, o planejador de produção tem total liberdade de repassar, ao processo de corte da fábrica, a solução que ele julgar mais conveniente.

Com relação a identificação das melhores soluções, pode-se notar que existe uma dificuldade na avaliação do resultado mais vantajoso, pois características podem se relacionar umas às outras e tal relação depende da decisão a ser tomada: custo de perda, de sobra e de corte; os intervalos para as sobras e perdas aceitáveis. Contudo, pode-se dizer que os resultados foram satisfatórios. Uma das razões para se afirmar isso foi o resultado obtido pelo Exemplo 1, que apresentou uma melhora significativa em relação aos demais trabalhos comparados, considerando o menor comprimento cortado para atender demanda e a diminuição da perda total de produção.

Como contribuições deste trabalho, pode-se destacar o desenvolvimento dos modelos apresentados no Capítulo 4, envolvendo uma divisão nos geradores de colunas para a produção de padrões com sobras e perdas aceitáveis, pois não foi encontrado na revisão teórica um modelo que fizesse esta disjunção. Os geradores de colunas, mesmo entre as abordagens propostas, sofreram alterações. A abordagem M2P&R, apesar de ser mais demorada por resolver várias mochilas, apresentou possibilidades maiores de melhorar a solução inteira que a abordagem M3P&R com a resolução de duas mochilas a cada iteração. A vantagem de se resolver menos problemas de mochila é que aumenta a velocidade de obtenção de uma boa solução do problema mestre. Não existem métodos de soluções gerais e eficientes para os problemas de corte devido principalmente à complexidade computacional dos problemas e à diversidade de casos em que estes podem aparecer. De modo geral,

observou-se que foi vantajosa a separação em dois geradores, já que foram vencidas algumas dificuldades existentes no trabalho de Koch (2008), tais como: limitação a problemas pequenos, pela necessidade de se conhecer todos os padrões a priori; e a impossibilidade de se escrever um gerador (único) utilizando a definição de padrão dada em seu trabalho. Uma outra vantagem está em não necessitar da definição dos comprimentos das sobras, como ocorre no trabalho de Scheithauer.

Como perspectiva de continuidade deste trabalho, merecem investigação a relação existente entre os custos de corte, perda e sobra que são fornecidos pelo planejador de produção, pois influenciam fortemente os resultados. Propõe-se ainda testar a abordagem 4 através de exemplos com demandas maiores, e estudar a possibilidade de, na abordagem 4, as sobras geradas na primeira etapa se tornarem um objeto para a etapa 2. Além disso, feito o cálculo do menor comprimento que atende a demanda, pelo primeiro estágio das Estratégias 1, 2 e 3 de Pinto (2008), notou-se que este poderia ser informado como limitação de estoque para alguns exemplos, condicionando o problema a utilizar o menor número de objetos e produzir perdas e sobras de acordo com os valores assumidos pelos parâmetros do problema. Então, essa possibilidade poderia ser estudada. Para uma nova abordagem, propõe-se acrescentar aos modelos desenvolvidos a movimentação de compartimentos que contém os objetos. O modelo de Scheithauer *et al.* [31] pode ser implementado utilizando outras técnicas de arredondamento, já que não foi publicado na língua inglesa o procedimento de arredondamento adotado para a obtenção de uma solução inteira. Outra possibilidade seria modificar o gerador de colunas inserindo a possibilidade de geração de sobras, ao invés de utilizar comprimentos determinados previamente. Quanto ao trabalho de Koch *et al.* [20], uma proposta seria escrever um gerador de colunas e aplicar o seu modelo em exemplos maiores, e no trabalho de Pinto [27], utilizar exemplos com demandas maiores e verificar como se comportam as estratégias apresentadas.

Durante a pesquisa, alguns trabalhos foram submetidos à periódicos e oficinas, os principais são apresentados nos Apêndices E e F.

REFERÊNCIAS

- [1] ABUABARA, A., *Otimização no corte de tubos estruturais: aplicação na indústria aeronáutica agrícola*. Dissertação de mestrado em Engenharia de Produção, Departamento de Engenharia de Produção, UFSCAR, (2006)
- [2] ARENALES, M., ARMENTANO, V., MORABITO, R., YANASSE, H., *Pesquisa Operacional para cursos de engenharia*. pp174
- [3] ALVEZ, C., CARVALHO, J.M.V., *Accelerating column generation for the single and variable sized-bin packing problem*. In Anais do V optimization, (2004), V. optimization
- [4] ARENALES, M.N., LOPES, M.A.L.P., *Estabilização da geração de colunas aplicada no problema de corte unidimensional de tamanhos variados*. In Livro de Resumos: IX Oficina Nacional de Problemas de Corte e Empacotamento, (2005)
- [5] BARNHART, C. et al., *Branch-and-price: column generation for solving huge integer programs*. Operations Research, v.46, p.316-329, (1998)
- [6] CARVALHO, J.M.V., *Exact solution of cutting stock problems using column generation and branch-and-bound*. Elsevier Science Ltd, v.5, No. 1, pp. 35-44, (1998)
- [7] CHERRI, A.C., *O problema de corte de estoque com reaproveitamento de sobras de material*. Dissertação de mestrado em Ciências de Computação e Matemática Computacional, Instituto de Ciências Matemáticas e de Computação, USP, (2006)
- [8] CHERRI, A.C. et al., *The one-dimensional cutting stock problem with usable leftover - A heuristic approach*. European Journal of Operational Research, doi:10.1016/j.ejor.2008.04.039, (2008)
- [9] CHU, C., ANTONIO, J., *Approximation algorithms to solve real life multicriteria cutting stock problems*. Operations Research, 47: 495-508, (1999)
- [10] DANTZIG, G.B., WOLFE, P., *Decomposition principle for linear programs*. Operations Research, 101-111, (1959)
- [11] GILMORE, P. C., GOMORY, R. E., *A linear programming approach to the cutting stock problem*. Operation Research, 9 (1961) 848-859
- [12] GILMORE, P. C., GOMORY, R. E., *A linear programming approach to the cutting stock problem-Part II*. Operation Research, 11 (1963) 863-888
- [13] GILMORE, P. C., GOMORY, R. E., *Multi-stage cutting stock problems of two and more dimensions*. Operation Research, 13 (1965) 94-120
- [14] GRADISAR, M., JESENKO, J., RESINOVIC, C., *Optimization of roll cutting in clothing industry*. Computers & Operational Research, 10:945-953, (1997)
- [15] GRADISAR, M., KLJAJIC, M., RESINOVIC, C., JESENKO, J., *A sequential heuristic procedure for one-dimensional cutting*. European Journal of Operational Research, 114: 557-568, (1999a)

- [16] GRADISAR, M., KLJAJIC, M., RESINOVIC, C., *A hybrid approach for optimization of one-dimensional cutting stock problem*. Computers & Operational Research, 119: 719-728, (1999b)
- [17] GRADISAR, M., TRKMAN, P., *A combined approach to the solution to the general one-dimensional cutting stock problem*. Computers & Operations Research, 32: 1793-1807, (2005)
- [18] JUNIOR, D.J.A., *O problema de corte de estoque com demanda estocástica*. Dissertação de mestrado, USP-São Carlos, (2006)
- [19] KANTOROVICH, L. V., *Mathematical Methods of Organizing and Planning Production*. Management Science, Vol. 6, No. 4 (Jul., 1960), pp. 366-422.
- [20] KOCH, S., KONIG, S., WASCHER, G., *Linear programming for a cutting problem in the wood processing industry-a case study*. FEMM Working Paper, No. 14, julho (2008)
- [21] KOS, L., DUHOVNIK, J., *Cutting optimization with variable-sized stock and inventory status data*. International Journal of Production Research, 40: 2289-2301, (2002)
- [22] LAND, A.H., DOIG, A.G., *An automatic method of solving discrete programming problems*. Econometrica, 28(3): pp. 497-520
- [23] LORENA, L.A., DOIG, A.G., *An automatic method of solving discrete programming problems*. Econometrica, 28(3): pp. 497-520 procurar se é assim
- [24] MENEZES, M. A. F., *Programação Linear*, Março 2006. Em elaboração. LabPL UCG - Universidade Católica de Goiás. último acesso: 13/12/2006.
- [25] MOSQUERA, G.P. *Contribuições para o Problema de Corte de Estoque Bidimensional na Indústria Moveleira* Dissertação de Mestrado-Pós-Graduação em Matemática Aplicada
- [26] MULLER, F. M., *Problemas de sequenciamento*. XXXV Simpósio Brasileiro de Pesquisa Operacional-SOBRAPO, novembro 2003, Natal, RN.
- [27] PINTO, T.S., *Uma proposta para resolver o problema de corte de estoque unidimensional com reaproveitamento de sobras por meio de dois objetivos*. Dissertação de mestrado em Matemática Aplicada e Computacional, Departamento de Matemática, UEL, (2008)
- [28] POLDI, K. C., *Algumas extensões do problema de corte de estoque*. Dissertação de mestrado, ICMC-USP, (2003)
- [29] POLDI, K. C., ARENALES, M.N., *Dealing with small demand in integer cutting stock problems with limited different stock lengths*. Notas do ICMC-Série Computação, No. 85, ICMC-USP, (2005)
- [30] ROODMAN, G.M., *Near-optimal solutions to one-dimensional cutting stock problem*. Computers and Operations Research, 13:713-719, (1986)
- [31] SCHEITHAUER, G., *A note on handling residual lengths*. Optimization, 22:3, 461-466, (1991)
- [32] SILVA, D. T., *Algoritmos Genéticos e o Problema de Corte Multiobjetivo*. Dissertação de Mestrado.

- [33] SINUANY, Z., WEINER, I., *The one dimensional cutting stock problem using two objectives*. J. Oper. Res. Soc. (UK), 45, 2, 231-236, (1994)
- [34] STADTLER, H., *A one-dimensional cutting stock problem in the aluminium industry and its solution*. European Journal of Operational Research, 44, 209-223, (1990)
- [35] TRKMAN, P., GRADISAR, M., *One dimensional cutting stock optimization in consecutive time periods*. European Journal of Operational Research, 179: 291-301, (2007)
- [36] WASCHER, G., GAU, T., *Heuristics for integer one-dimensional cutting stock problem: a computational study*. OR Spektrum, 18, pp 131-144, 1996.

APÊNDICES

APÊNDICE A – Demonstração de equivalência entre minimizar perda e objetos

A seguir, a demonstração da equivalência entre minimizar perda e minimizar o número de objetos a serem cortados, no caso em que só há um tipo de objeto no estoque.

Tem-se que:

$$\begin{aligned}
 \sum_{p=1}^n c_p x_p &= \sum_{p=1}^n (L - \sum_{i=1}^m a_{ip} l_i) x_p \\
 \sum_{p=1}^n L x_p - \sum_{p=1}^n \sum_{i=1}^m a_{ip} l_i x_p &= \\
 \sum_{p=1}^n L x_p - \sum_{i=1}^m \sum_{p=1}^n a_{ip} l_i x_p &= \\
 \sum_{p=1}^n L x_p - \sum_{i=1}^m l_i \sum_{p=1}^n a_{ip} x_p &= \\
 \sum_{p=1}^n L x_p - \sum_{i=1}^m l_i d_i &
 \end{aligned}$$

Note que $\sum_{p=1}^n L x_p$ é o comprimento total que foi cortado para o cumprimento da demanda. A soma $\sum_{i=1}^m l_i d_i$ é constante, assim como L . Logo basta minimizar o número de objetos a serem cortados.

APÊNDICE B – Exemplos

B.1 Problema sem limitação de estoque

Neste exemplo, o estoque é formado por dezesseis tipos de objetos sem limitação, cujos dados estão na Tabela B.1. O comprimento de cada item e sua respectiva demanda estão na Tabela B.2.

Tabela B.1: Estoque 1 (sem limitação).

Tipo de Objeto	Comprimento (u.c.)
1	3644
2	4036
3	4468
4	4500
5	4536
6	5120
7	5344
8	6000
9	9652
10	10052
11	10284
12	13744
13	15032
14	15444
15	21060
16	24060

O menor item mede 3330 u.c., qualquer comprimento remanescente do objeto que não corresponda a um item demandado e que possua comprimento superior a 3330 u.c. pode ser considerado

Tabela B.2: Dados do Exemplo sem limitação.

Tipo de Item	Comprimento (u.c.)	Demanda
1	3330	1
2	9200	2
3	9600	1
4	10100	2
5	11250	1
6	12600	3

sobra. Retalhos menores que 3330 u.c. seria uma perda, mas nas abordagens propostas neste trabalho existe ainda, o intervalo de perda aceitável.

B.2 Problemas com limitação de estoque

B.2.1 Exemplo 1

No primeiro exemplo, o estoque é formado por três tipos de objetos cujos dados estão na Tabela B.3. A demanda e o comprimento de cada item estão na Tabela B.4.

Tabela B.3: Estoque do Exemplo 1.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	2
2	3500	2
3	6000	1

Tabela B.4: Dados do Exemplo 1.

Tipo de Item	Comprimento (u.c.)	Demanda
1	485	2
2	490	4
3	495	3
4	500	4
5	402	2
6	415	1
7	420	1
8	460	2
9	470	4

Neste exemplo, qualquer retalho maior ou igual a 402 u.c. pode ser considerado sobra.

B.2.2 Exemplo 2

No exemplo 2, tem-se três tipos de objetos cujos dados estão na Tabela B.5. O comprimento de cada item e sua respectiva demanda estão na Tabela B.6.

Tabela B.5: Estoque do Exemplo 2.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	1125	3
2	5423	3
3	7562	3

Tabela B.6: Dados do Exemplo 2.

Tipo de Item	Comprimento (u.c.)	Demanda
1	254	39
2	186	15
3	123	21
4	97	18

Neste exemplo, o menor item mede 347 u.c..

B.2.3 Exemplo 3

Este exemplo 3 foi extraído de Sinuany-Stern e Weiner (1994). O estoque é formado por dos tipos de objetos cujos dados estão na Tabela B.7. O comprimento de cada item e sua respectiva demanda estão na Tabela B.8.

Tabela B.7: Estoque do Exemplo 3.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	234	9
2	600	2

Tabela B.8: Dados do Exemplo 3.

Tipo de Item	Comprimento (u.c.)	Demanda
1	123	15
2	103	6
3	30	6
4	17	6

Neste exemplo, o menor item mede 17 u.c..

B.2.4 Exemplo 4

No quarto exemplo o estoque é formado por três tipos de objetos, cujos dados estão na Tabela B.9. O comprimento de cada item e sua respectiva demanda estão na Tabela B.10.

Tabela B.9: Estoque do Exemplo 4.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	3
2	3500	3
3	6000	3

Tabela B.10: Dados do Exemplo 4.

Tipo de Item	Comprimento (u.c.)	Demanda
1	80	20
2	30	10
3	50	10
4	70	5
5	10	20
6	12	20
7	14	20
8	15	40
9	20	10
10	28	10
11	30	35
12	31	20
13	32	10
14	33	20
15	35	15
16	36	5
17	39	10
18	40	5
19	43	20
20	65	5
21	75	5
22	90	25
23	100	5

Neste exemplo, o menor item mede 10 u.c., qualquer comprimento remanescente do objeto que não corresponda a um item demandado e que possua comprimento superior a 10 u.c. pode ser considerado sobra. Retalhos menores que 10 u.c. seria uma perda, mas ainda existe o intervalo de perda aceitável, que permite, por exemplo, perder até 1 u.c. por objeto cortado, considerado nos modelos propostos neste trabalho.

B.2.5 Exemplo 5

No exemplo 5, tem-se três tipos de objetos cujos dados estão na Tabela B.11. O comprimento de cada item e sua respectiva demanda estão na Tabela B.12.

Tabela B.11: Estoque do Exemplo 5.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	1
2	3500	4
3	6000	1

Este é o exemplo com maior número de itens distintos. O menor deles mede 100 u.c..

Tabela B.12: Dados do Exemplo 5.

Tipo de Item	Comprimento (u.c.)	Demanda
1	100	1
2	110	2
3	115	2
4	140	1
5	150	4
6	160	9
7	170	1
8	180	2
9	190	2
10	205	1
11	230	1
12	235	2
13	250	3
14	260	1
15	285	4
16	310	2
17	350	4
18	360	4
19	370	1
20	380	1
21	400	2
22	402	2
23	415	1
24	420	1
25	460	2
26	470	4
27	475	1
28	480	1
29	485	2
30	120	1
31	148	1
32	210	2
33	315	2
34	320	2
35	490	2
36	495	1
37	500	2
38	273	1

B.2.6 Exemplo 6

No exemplo 6, tem-se três tipos de objetos cujos dados estão na Tabela B.13. O comprimento de cada item e sua respectiva demanda estão na Tabela B.14.

Tabela B.13: Estoque do Exemplo 6.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	2
2	3500	2
3	6000	3

Tabela B.14: Dados do Exemplo 6.

Tipo de Item	Comprimento (u.c.)	Demanda
1	525	4
2	505	2
3	530	4
4	535	2
5	550	4
6	556	4
7	570	8

Neste exemplo, o menor item mede 505.

B.2.7 Exemplo 7

O exemplo 7 tem os mesmos objetos de estoque, mas em quantidades diferentes, os dados estão na Tabela B.15. O comprimento de cada item e sua respectiva demanda estão na Tabela B.16.

Tabela B.15: Estoque do Exemplo 7.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	3
2	3500	4
3	6000	3

Tabela B.16: Dados do Exemplo 7.

Tipo de Item	Comprimento (u.c.)	Demanda
1	315	2
2	320	2
3	350	6
4	310	2
5	360	4
6	370	1
7	380	1
8	400	2

Neste exemplo, o menor item mede 310 u.c., assim um retalho será considerado sobra apenas se for maior ou igual a 310 u.c., porém dependerá do valor que o planejador fornecer ao programa.

APÊNDICE C – Problema da Mochila e métodos de resolução

O Problema da Mochila é um problema de otimização combinatória. O nome se dá devido ao modelo de uma situação em que é necessário preencher uma mochila com objetos de diferentes pesos e valores. O objetivo é que se preencha a mochila com o maior valor possível, não ultrapassando o peso máximo. Este problema está relacionado com um grande número de outros modelos de programação, o que mostra sua importância. O modelo em si pode ser aplicado em casos como: investimento de capital, corte e empacotamento, carregamento de veículos, entre outros.

Por exemplo, suponha que um objeto deva ser cortado ao longo de seu comprimento L em itens de comprimento l_i , $i = 1, \dots, m$. Cada item tem um valor associado que denomina-se valor de utilidade. Itens cujos comprimentos não foram especificados são considerados perdas e tem valor de utilidade nulos. Sendo assim, surge o problema de maximizar o valor da utilidade total. Se considerado que a quantidade máxima de itens demandados não seja especificada, tem-se o Problema da Mochila Irrestrito.

C.1 Modelagem matemática

Seja α_i uma variável de decisão: quantidade de itens de índice i selecionados, $i = 1, \dots, m$; L o comprimento do objeto em estoque; m o número de itens distintos; l_i o comprimento do item de índice i , $i = 1, \dots, m$ e v_i o valor da utilidade do item i , $i = 1, \dots, m$. Assim, o modelo pode ser escrito como segue:

$$\begin{aligned} \text{maximizar} \quad & \sum_{i=1}^m v_i \alpha_i \\ \text{sujeito a} \quad & \sum_{i=1}^m l_i \alpha_i \leq L \\ & \alpha_i \geq 0, \quad i = 1, \dots, m \text{ e inteiros} \end{aligned}$$

Quando ocorre limitação na quantidade de itens de um determinado tipo, o problema passa a ser chamado de Problema da Mochila Restrito ou Limitado. Neste caso, além dos dados necessários para o modelo descrito anteriormente, deve-se considerar ainda d_i denotando a quantidade máxima de itens de índice i que pode ser selecionada, $i = 1, \dots, m$. O modelo é apresentado a seguir.

$$\begin{aligned} \text{maximizar} \quad & \sum_{i=1}^m v_i \alpha_i \\ \text{sujeito a} \quad & \sum_{i=1}^m l_i \alpha_i \leq L \\ & 0 < \alpha_i < d_i, \quad i = 1, \dots, m \text{ e inteiros} \end{aligned}$$

C.2 Métodos de resolução

Por ser um problema combinatório é possível resolvê-lo por enumeração exaustiva, ou seja tentar todas as soluções possíveis e comparando-as para identificar a melhor solução. Porém isso se torna completamente inviável, uma vez que, mesmo em pequenas dimensões como um problema com apenas 20 itens, haveria um número enorme de respostas. Em um problema com mais de 80 itens, o algoritmo levaria bilhões de anos para ser concluído. Assim, o método de resolução por enumeração exaustiva é de utilidade praticamente nula para problemas de grande dimensão.

Uma maneira de se resolver um Problema da Mochila é utilizando o Backtracking, que é um algoritmo refinado de enumeração exaustiva, no qual boa parte das soluções podem ser eliminadas sem serem explicitamente examinadas.

O Método Guloso é um dos algoritmos mais simples que pode ser aplicado a uma grande variedade de problemas. No Método Guloso é construído um algoritmo que trabalha em estágios, considerando uma entrada por vez. Em cada estágio é tomada uma decisão considerando se uma entrada particular é uma solução ótima. Isto é feito considerando as entradas em uma ordem determinada por um processo de seleção, que é baseado em alguma medida de otimização que pode ou não ser a função objetivo. Na maioria das vezes, porém, essas medidas de otimização resultam em algoritmos que geram soluções sub-ótimas.

Programação Dinâmica, ou Função Recursiva, foi uma técnica usada para resolver esse problema, na década de 50. É um método aprimorado de usar recursividade que ao invés de chamar uma função várias vezes, armazena o resultado na primeira vez que é chamada para que, cada vez que a função for chamada novamente retorne o resultado e não uma requisição para ser resolvida.

Gilmore e Gomory (1961) propuseram um método de Programação Dinâmica para o Problema de Corte. Este método apresenta uma fórmula recursiva para a resolução do Problema da Mochila utilizando uma estratégia de percorrer todas as possíveis decisões.

Foi proposto por George Dantzig, em 1957, um algoritmo de aproximação guloso para resolver o problema da mochila. A versão dele dispõe os itens em ordem decrescente de valor por unidade de peso. Em seguida, estes começam a ser inseridos na mochila com tantas cópias quanto possível do primeiro tipo de item, até que não haja mais espaço na mochila. Caso o problema seja delimitado, ou seja, a oferta para cada tipo de item tenha um limite, o algoritmo pode ficar muito custoso.

Um método de Enumeração Implícita foi proposto por Gilmore e Gomory (1963). O método proposto é implementado utilizando primeiramente uma busca em profundidade numa árvore de decisões e basicamente consiste em enumerar implicitamente todas as soluções do Problema da Mochila. Para isto, utiliza-se limitantes para o problema que permitem que soluções “piores” que a atual sejam descartadas, sem perder a otimalidade.

APÊNDICE D – Obtenção das mochilas para as abordagens propostas

Geralmente, em problemas práticos, o número de variáveis é muito grande. Para tanto, o Método Simplex torna-se impraticável devido ao grande número de variáveis que devem ser investigadas. Uma técnica eficiente para contornar este tipo de problema é a Técnica de Geração de Colunas, que foi proposta por Gilmore e Gomory (1961).

Por simplicidade, considere o problema de otimização linear apresentado no capítulo 2:

$$\text{minimizar} \quad \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \quad (\text{D.1})$$

$$\text{sujeito a} \quad \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} = d_i, \quad i \in I \quad (\text{D.2})$$

$$x_{jp} \geq 0 \text{ e inteiro}, \quad j \in J, p \in P(j) \quad (\text{D.3})$$

Dado início ao Método Simplex, o próximo passo consiste em determinar uma coluna a entrar na base. Essa escolha é realizada selecionando-se o custo relativo $(c_p - \pi^T a_p)$ que apresente um valor negativo. Uma escolha inicial, que procura a variável x_k com o menor custo relativo é dada por:

$$c_k - \pi^T a_k = \min\{c_p - \pi^T a_p, p \in \mathfrak{N}\}$$

Se o custo relativo, $c_k - \pi^T a_k \geq 0$ então a solução básica é ótima, caso contrário, a coluna a_k entra na base.

De outro modo, encontrar uma coluna não básica corresponde a resolver o seguinte Problema da Mochila:

$$\begin{aligned} \text{maximizar} \quad & \sum_{i=1}^m \pi_i a_{ip} - c_p \\ \text{sujeito a} \quad & \sum_{i=1}^m l_i a_{ip} \leq L \\ & a_{ip} \geq 0 \text{ e inteiro} \end{aligned}$$

que é um Problema da Mochila irrestrito, isto é, não há um limitante superior para a quantidade máxima de a_{ip} .

Como para as abordagens M1P&R, M2P&R e M4P&R os custos dependem dos objetos e dos padrões de corte com

$$c_{jp} = c^{\text{processo}} \left(\sum_{i \in I} a_{ijp} \right) + c^{\text{retalho}} \left(L_j - \sum_{i \in I} a_{ijp} l_i \right)$$

então,

$$\begin{aligned} c_{jp}(a_{jp}) - \pi^T a_{jp} &= c_{jp} = c^{\text{processo}} \left(\sum_{i \in I} a_{ijp} \right) + c^{\text{retalho}} \left(L_j - \sum_{i \in I} a_{ijp} l_i \right) - \sum_{i \in I} \pi_i a_{ijp} \\ &= \sum_{i \in I} c^{\text{processo}} a_{ijp} + c^{\text{retalho}} L_j - \sum_{i \in I} c^{\text{retalho}} a_{ijp} l_i - \sum_{i \in I} \pi_i a_{ijp} \\ &= \sum_{i \in I} (c^{\text{processo}} a_{ijp} - c^{\text{retalho}} a_{ijp} l_i - \pi_i a_{ijp}) + c^{\text{retalho}} L_j \\ &= \sum_{i \in I} (c^{\text{processo}} - c^{\text{retalho}} l_i - \pi_i) a_{ijp} + c^{\text{retalho}} L_j \end{aligned}$$

O subproblema correspondente é mostrado a seguir e t_{jp} representa o retalho produzido no processo de corte.

$$\begin{aligned}
 \text{maximizar} \quad & \sum_{i \in I} (\pi_i - c^{\text{processo}} + c^{\text{retalho}} l_i) a_{ijp} - c^{\text{retalho}} L_j \\
 \text{sujeito a} \quad & \sum_{i \in I} l_i a_{ijp} + t_{jp} = L_j, j \in J, p \in P(j) \\
 & \sum_{i \in I} a_{ijp} \geq 1, j \in J, p \in P(j) \\
 & a_{ijp} \geq 0 \text{ e inteiro}, i \in I, j \in J, p \in P(j) \\
 & t_{jp} \geq 0, j \in J, p \in P(j)
 \end{aligned}$$

Um padrão de corte é dito factível se uma das seguintes condições valem.

1. O objeto disponível de comprimento L_j é usado completamente, isto é,

$$t_{jp} = 0 \Leftrightarrow \sum_{i \in I} l_i a_{ijp} = L_j;$$

2. O retalho é suficientemente pequeno para ser aceito como perda, isto é,

$$0 < t_{jp} = L_j - \sum_{i \in I} l_i a_{ijp} \leq L_{\text{perda}_j};$$

3. O retalho tem um comprimento que é aceito como sobra, isto é,

$$L_{\text{sobra}_j} \leq t_{jp} = L_j - \sum_{i \in I} l_i a_{ijp} \leq L_{\text{sobra}_j}^{\text{max}}$$

Por definição, c^{retalho} representa o custo de retalho, que pode ser c^{perda} ou c^{sobra} , assim tem-se a seguinte divisão do Problema da Mochila:

$$\begin{aligned}
 \text{maximizar} \quad & Z_j^{\text{perda}} = \sum_{i \in I} (\pi_i + c^{\text{perda}} l_i - c^{\text{processo}}) a_{ij} - c^{\text{perda}} L_j \\
 \text{sujeito a} \quad & \sum_{i \in I} a_{ij} l_i \leq L_j \\
 & \sum_{i \in I} a_{ij} l_i \geq L_j - L_{\text{perda}_j} \\
 & \sum_{i \in I} a_{ij} \geq 1 \\
 & a_{ij} \leq d_i, i \in I \\
 & a_{ij} \geq 0 \text{ e inteiro}, j \in J
 \end{aligned}$$

$$\begin{aligned}
\text{maximizar} \quad & Z_j^{sobra} = \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ij} - c^{sobra} L_j \\
\text{sujeito a} \quad & \sum_{i \in I} a_{ij} l_i \leq L_j - L_{sobra_j} \\
& \sum_{i \in I} a_{ij} l_i \geq L_j - L_{sobra_j}^{max} \\
& \sum_{i \in I} a_{ij} \geq 1 \\
& a_{ij} \leq d_i, \quad i \in I \\
& a_{ij} \geq 0 \text{ e inteiro}, \quad j \in J
\end{aligned}$$

Já para a abordagem M3P&R, o custo relativo é dado por:

$$c_{jp}(a_{jp}) = c^{processo} \left(\sum_{i \in I} a_{ijp} \right) + c^{retalho} \left(\sum_{i \in J} L_i a_{(i+m)jp} - \sum_{i \in I} a_{ijp} l_i \right)$$

Então,

$$\begin{aligned}
c_{jp}(a_{jp}) - \pi^T a_{jp} &= c^{processo} \left(\sum_{i=1}^m a_{ijp} \right) + c^{retalho} \left(\sum_{i=1}^M L_i a_{(i+m)jp} - \sum_{i=1}^m a_{ijp} l_i \right) - \sum_{i=1}^{m+M} \pi_i a_{ijp} \\
&= \sum_{i=1}^m c^{processo} a_{ijp} + c^{retalho} \left(\sum_{i=1}^M L_i a_{(i+m)jp} - \sum_{i=1}^m l_i a_{ijp} \right) - \sum_{i=1}^m \pi_i a_{ijp} - \sum_{i=1}^M \pi_{(i+m)} a_{(i+m)jp} \\
&= \sum_{i=1}^m (c^{processo} - c^{retalho} l_i - \pi_i) a_{ijp} + \sum_{i=1}^M (c^{retalho} L_i - \pi_{(i+m)}) a_{(i+m)jp}
\end{aligned}$$

De acordo com a definição de $c^{retalho}$, as mochilas correspondentes são:

$$\begin{aligned}
\text{maximizar} \quad & Z^{perda} = \sum_{i \in I} (\pi_i + c^{perda} l_i - c^{processo}) a_i + \sum_{i \in J} (\pi_{i+m} - c^{perda} L_i) a_{i+m} \\
\text{sujeito a} \quad & \sum_{i \in I} a_i l_i \leq \sum_{i \in J} L_i a_{i+m} \\
& \sum_{i \in I} a_i l_i \geq \sum_{i \in J} (L_i - L_{perda_i}) a_{i+m} \\
& \sum_{i \in I} a_i \geq 1 \\
& \sum_{i \in J} a_{i+m} = 1 \\
& a_i \leq qteItemMax_i \\
& a_i \geq 0 \text{ e inteiro}
\end{aligned}$$

$$\begin{aligned}
\text{maximizar} \quad & Z^{sobra} = \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_i + \sum_{i \in J} (\pi_{i+m} - c^{sobra} L_i) a_{i+m} \\
\text{sujeito a} \quad & \sum_{i \in I} a_i l_i \leq \sum_{i \in J} (L_i - L_{sobra_i}) a_{i+m} \\
& \sum_{i \in I} a_i l_i \geq \sum_{i \in J} (L_i - L_{sobra_i}^{max}) a_{i+m} \\
& \sum_{i \in I} a_i \geq 1 \\
& \sum_{i \in J} a_{i+m} = 1 \\
& a_i \leq qteItemMax_i \\
& a_i \geq 0 \text{ e inteiro}
\end{aligned}$$

APÊNDICE E – Artigo submetido à TEMA

A revista TEMA (Tendências em Matemática Aplicada e Computacional) tem como objetivo principal publicar trabalhos completos originais de todas as áreas de Matemática Aplicada e Computacional. O artigo submetido a esta revista é apresentado a seguir.

Artigo: “**Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras: abordagens de resolução em estágios**”

P.C. FARIAS, PGMAC, UEL - Univ Estadual de Londrina, 86051-980 Londrina, PR, Brasil, T.S. PINTO, UTFPR - Univ Tecnológica Federal do Paraná, 86300-000 Cornélio Procópio, PR, Brasil, R.S.V. HOTO, Departamento de Matemática, UEL - Univ Estadual de Londrina, 86051-980 Londrina, PR, Brasil.

Resumo. Este trabalho apresenta cinco estratégias de resolução para o Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras, utilizando abordagens em uma e duas etapas. São apresentados resultados de simulações feitas com estas estratégias e com outros trabalhos da literatura. Além disto, uma nova abordagem utilizando uma técnica de geração de colunas é apresentada.

Palavras-chave. Problema de corte de estoque, reaproveitamento.

Introdução

Um dos grandes desafios das indústrias é conseguir produzir mais, com melhor qualidade e menor custo. Em muitos segmentos industriais, processos de corte possuem importância fundamental, tais como nas indústrias de móveis, papel, colchões, indústrias do segmento têxtil, de metalurgia, entre inúmeras outras.

O bom aproveitamento da matéria-prima é um dos fatores predominantes para a redução do custo da produção. Nesse sentido, pequenas melhorias nos processos que envolvem o corte de material podem levar a ganhos substanciais, dependendo da escala de produção e do custo da matéria-prima. Surge assim a necessidade de se planejar os cortes para atenuar os efeitos negativos associados à perda de material. Este tipo de problema é conhecido como Problema de Corte de Estoque (PCE).

Em geral, a matéria-prima pode ser melhor aproveitada se as sobras resultantes do processo de corte puderem ser reutilizadas. Assim, este trabalho aborda o PCE unidimensional com reaproveitamento de sobras. Este tipo de problema foi discutido por diversos autores, tais como Roodman (1986), Scheithauer (1991), Sinuany-Stern e Weiner (1994), Gradisar et al. (1997, 1999a, 1999b), Gradisar e Trkman (2005), Cherri (2006), Abuabara (2006), Cherri et al. (2008), Koch et al. (2008) e Abuabara e Morabito (2008).

Problema de Corte de Estoque Unidimensional

No problema de corte de estoque unidimensional, peças menores (itens) devem ser cortadas de objetos em estoque, disponíveis em quantidade suficiente para atender aos pedidos de clientes (demandas). O problema consiste em efetuar o processo de corte minimizando o número de objetos a serem utilizados, o custo total do processo, as perdas geradas, ou outro objetivo.

Será considerado que o estoque é composto por objetos de vários comprimentos, em quantidades limitadas suficientes para atender às demandas. Assim, um modelo para o problema de corte de estoque unidimensional com vários tipos de barras em estoque pode ser escrito como:

$$\begin{aligned}
 \min \quad & \sum_{p=1}^{P_1} T_{p1}x_{p1} + \sum_{p=1}^{P_2} T_{p2}x_{p2} + \dots + \sum_{p=1}^{P_J} T_{pJ}x_{pJ} \\
 \text{s. a} \quad & \sum_{p=1}^{P_1} a_{p1}x_{p1} + \sum_{p=1}^{P_2} a_{p2}x_{p2} + \dots + \sum_{p=1}^{P_J} a_{pJ}x_{pJ} = d \\
 & \sum_{p=1}^{P_j} x_{pj} \leq e_j, \quad j = 1, 2, \dots, J \\
 & x_{pj} \geq 0 \text{ e inteiro}, \quad p = 1, \dots, P_j, \quad j = 1, 2, \dots, J
 \end{aligned}$$

Onde: $a_{pj} = (a_{1pj}, a_{2pj}, \dots, a_{I pj})^T$ é um vetor associado a um padrão de corte; a_{ipj} é a quantidade de vezes que o item de comprimento l_i aparece no padrão de índice i , no objeto de comprimento L_j ; e_j é a quantidade de objetos de comprimento L_j disponível; x_{pj} é o total de objetos L_j que foram cortados, segundo o padrão de corte de índice p ; $T_{pj} = L_j - \sum_{i=1}^I l_i a_{ipj}$ representa a perda produzida pelo padrão de índice p no objeto de comprimento L_j , onde $i = 1, \dots, I$, $p = 1, \dots, P_j$ e $j = 1, \dots, J$.

Neste trabalho, o modelo anterior será denominado Problema de Corte de Estoque Unidimensional Restrito (PCER).

O PCEU com Reaproveitamento de Sobras

Durante o processo de corte são gerados retalhos, os quais serão denominados “sobras” quando puderem ser reutilizados, e “perdas” caso contrário. A definição de sobra varia e, em geral, depende de critérios adotados. Assim, serão considerados como sobras os retalhos com um comprimento mínimo $S > 0$.

O Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras (PCES) consiste em encontrar um conjunto de padrões de corte que proporcione menos perdas e mais sobras distribuídas no menor número possível de objetos, para que as mesmas não fiquem pulverizadas, já que as indústrias preferem dispor de poucas sobras de dimensões maiores, ao invés de muitas sobras de pequenas dimensões. Minimizar as perdas é equivalente a maximizar as sobras, desde que o compromisso de minimizar os retalhos seja mantido.

Neste modelo para o PCES, serão considerados dois conjuntos de padrões de corte: padrões com perdas e padrões com sobras. Para cada objeto de comprimento L_j ($j = 1, \dots, J$), existem P_j^α padrões de corte com perdas e P_j^β padrões com sobras. Então, $\alpha_{pj} = (\alpha_{1pj}, \alpha_{2pj}, \dots, \alpha_{I pj})^T$ representa um padrão com perda se $T_{pj} = L_j - \sum_{i=1}^I l_i \alpha_{ipj} < S$, onde α_{ipj} é a quantidade de vezes que o item de comprimento l_i ($i = 1, \dots, I$) aparece no padrão de índice $p = 1, \dots, P_j^\alpha$, no objeto de comprimento L_j . Analogamente, $\beta_{qj} = (\beta_{1qj}, \beta_{2qj}, \dots, \beta_{I qj})^T$ representa um padrão com sobra se $S_{qj} = L_j - \sum_{i=1}^I l_i \beta_{ipj} \geq S > 0$, onde β_{iqj} é a quantidade de vezes que o item de comprimento l_i ($i = 1, \dots, I$) aparece no padrão de índice $q = 1, \dots, P_j^\beta$, no objeto de comprimento L_j .

Além disso, x_{pj}^α define o total de objetos de comprimento L_j que devem ser cortados, segundo o padrão de corte de índice $p = 1, \dots, P_j^\alpha$, produzindo perdas, enquanto x_{qj}^β define o total de objetos de comprimento que devem ser cortados, segundo o padrão de corte de índice $q = 1, \dots, P_j^\beta$, produzindo sobras. Por fim, seja e^j a quantidade disponível do objeto de comprimento L_j ($j = 1, \dots, J$), e $d = (d_1, \dots, d_I)^T$ o vetor de demandas dos itens.

Assim, o modelo para o PCES proposto pode ser escrito como:

$$\begin{aligned}
 \min \quad & \sum_{p=1}^{P_1^\alpha} T_{p1} x_{p1}^\alpha + \dots + \sum_{p=1}^{P_J^\alpha} T_{pJ} x_{pJ}^\alpha + \sum_{q=1}^{P_1^\beta} S_{q1} x_{q1}^\beta + \dots + \sum_{q=1}^{P_J^\beta} S_{qJ} x_{qJ}^\beta \\
 \max \quad & \sum_{q=1}^{P_1^\beta} S_{q1} x_{q1}^\beta + \dots + \sum_{q=1}^{P_J^\beta} S_{qJ} x_{qJ}^\beta \\
 \min \quad & \sum_{q=1}^{P_1^\beta} x_{q1}^\beta + \dots + \sum_{q=1}^{P_J^\beta} x_{qJ}^\beta \\
 \text{s. a} \quad & \sum_{p=1}^{P_1^\alpha} \alpha_{p1} x_{p1}^\alpha + \dots + \sum_{p=1}^{P_J^\alpha} \alpha_{pJ} x_{pJ}^\alpha + \sum_{q=1}^{P_1^\beta} \beta_{q1} x_{q1}^\beta + \dots + \sum_{q=1}^{P_J^\beta} \beta_{qJ} x_{qJ}^\beta = d \\
 & \sum_{p=1}^{P_j^\alpha} x_{pj}^\alpha + \sum_{q=1}^{P_j^\beta} x_{qj}^\beta \leq e_j, \quad j = 1, \dots, J \\
 & x_{pj}^\alpha, x_{qj}^\beta \geq 0 \text{ e inteiros, } p = 1, \dots, P_j^\alpha, q = 1, \dots, P_j^\beta, j = 1, \dots, J
 \end{aligned}$$

Propostas de Resolução do PCES

As diferenças entre os modelos do PCER e PCES é que neste último são distinguidos dentre os padrões viáveis, aqueles que geram sobras.

Em alguns casos, pode não ser possível efetuar reaproveitamento das sobras se a soma das perdas da solução do PCER for inferior ao comprimento mínimo $S > 0$ que representa uma sobra. Também, esta solução é solução ótima para o PCES. Esta propriedade mostra que seria muito útil, se possível, obter informações das perdas do PCER *a priori*. Porém, informações desta natureza não são simples de serem obtidas. Ainda, se estiverem disponíveis informações sobre as soluções alternativas do PCER, boas informações iniciais para resolver o PCES poderão ser obtidas.

Estas observações sugerem estratégias para obter um conjunto das soluções ótimas ou quase-ótimas do PCER e, em seguida, buscar neste conjunto soluções que maximizem as sobras, distribuindo-as no menor número de objetos possível. Neste sentido, três estratégias de resolução para o PCES são propostas. Estas, num primeiro estágio, encontram o menor comprimento que deve ser cortado de objetos para o atendimento dos itens demandados e, num segundo estágio, buscam concentrar as perdas no menor número de objetos possível. Além destas, são propostas duas outras estratégias visando resolver o problema em um único estágio.

Estratégias em Dois Estágios

Estas estratégias desenvolvidas para resolução do PCES possuem objetivos diferentes em cada estágio. O primeiro estágio consiste em selecionar os objetos que deverão ser cortados para atender os itens demandados, minimizando o comprimento total a ser cortado. No segundo estágio, os objetos selecionados são cortados de maneira a gerar sobras, agrupando as perdas num número reduzido de objetos.

Nas três estratégias desenvolvidas, o primeiro estágio é sempre igual, definido pelo seguinte modelo matemático:

$$\min \sum_{j=1}^J y_j L_j \quad (E.1)$$

$$\text{s. a } \sum_{j=1}^J a_{ij} = d_i, \quad i = 1, \dots, m \quad (E.2)$$

$$\sum_{i=1}^m l_i a_{ij} \leq L_j y_j, \quad j = 1, \dots, J \quad (E.3)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i = 1, \dots, m, \quad j = 1, \dots, J \quad (E.4)$$

$$y_j \in \{0, 1\}, \quad j = 1, \dots, J \quad (E.5)$$

Onde: L_j representa o comprimento do objeto de índice j ; d_i é a demanda e l_i o comprimento do item de índice i ; a_{ij} são as variáveis que representam número de itens de índice i cortados do objeto de índice j ; as variáveis y_j assumem valores binários que indicam se o objeto de índice j é utilizado ou não; $i = 1, \dots, I$; $j = 1, \dots, J$.

A função objetivo (E.1) minimiza o comprimento total utilizado para atender a demanda. A restrição (E.2) garante que a quantidade de itens produzidos seja igual à demanda. A restrição (E.3) funciona da seguinte forma: se o objeto de índice j está sendo usado ($y_j = 1$) a mesma equivale a uma restrição de mochila, por outro lado, se o objeto de índice j não está sendo usado ($y_j = 0$) a restrição obriga que $a_{ij} = 0$, $i = 1, \dots, m$. Por fim, as restrições (E.4) e (E.5) definem as variáveis do problema.

Estratégia 1

O modelo proposto é dado por:

$$\min \sum_{j=1}^N (L_j - \sum_{i=1}^m l_i a_{ij}) \quad (E.6)$$

$$\text{s. a } \sum_{j=1}^N w_j = R \quad (E.7)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in I \quad (E.8)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq w_j L_j + (1 - w_j)(S - 1), \quad j \in J \quad (E.9)$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq w_j S, \quad j \in J \quad (E.10)$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i \in I, \quad j \in J \quad (E.11)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (E.12)$$

Onde: N representa o número de objetos escolhidos no primeiro estágio; R indica o número de objetos usados para concentrar perdas não aproveitáveis; L_j é o comprimento do objeto de índice j escolhido no primeiro estágio; d_i é a demanda do item de índice i ; l_i indica o comprimento do item de índice i ; S é o comprimento mínimo aceito para sobra; a_{ij} são as variáveis que representam o número de itens de índice i cortados do objeto de índice j ; as variáveis w_j assumem valores binários que indicam se o objeto de índice j é escolhido para ter sobra ou não; $i = 1, \dots, I$; $j = 1, \dots, J$.

A função objetivo (E.6) minimiza as perdas e sobras do processo. A restrição (E.7) garante que o número de objetos utilizados para concentrar as perdas seja igual ao número R previamente fixado. A restrição (E.8) corresponde ao atendimento exato das demandas dos itens. As restrições (E.9) e (E.10) juntas, indicam para cada objeto, que o comprimento remanescente do corte dos itens deve estar no intervalo $[S, L_j]$ quando $w_j = 1$ (o objeto deve concentrar perdas), ou deve estar no intervalo $[0, S)$ quando $w_j = 0$ (o objeto não deve concentrar perdas). Como as variáveis do problema são inteiras, na restrição (E.9), $(1 - w_j)$ foi multiplicado por $(S - 1)$ para que o comprimento remanescente do corte dos itens esteja no intervalo $[0, S)$ quando $w_j = 0$. Por fim, as restrições (E.11) e (E.12) definem as variáveis do problema.

Esta estratégia minimiza as perdas visando a geração de sobras.

Estratégia 2

Nesta estratégia, é introduzido $M \geq \max L_j | j = 1, \dots, N$, e novas variáveis são consideradas, como mostra o modelo:

$$\min \sum_{j=1}^N T_j \quad (\text{E.13})$$

$$\text{s. a } \sum_{j=1}^N w_j = R \quad (\text{E.14})$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in m \quad (\text{E.15})$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq L_j + (1 - w_j)M, \quad j \in N \quad (\text{E.16})$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq S - (1 - w_j)M, \quad j \in N \quad (\text{E.17})$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \leq T_j + w_j M, \quad j \in N \quad (\text{E.18})$$

$$L_j - \sum_{i=1}^m l_i a_{ij} \geq T_j - w_j M, \quad j \in N \quad (\text{E.19})$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in m, j \in N \quad (\text{E.20})$$

$$T_j \geq 0 \text{ e inteiro, } j \in N \quad (\text{E.21})$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (\text{E.22})$$

Onde: T_j são as variáveis que indicam o comprimento de perda no objeto de índice j ; as variáveis y_j assumem valores binários que indicam se o objeto de índice j é utilizado ou não; $j = 1, \dots, N$.

A função objetivo (E.13) minimiza as perdas do processo. As restrições (E.14) e (E.15) são semelhantes às (E.7) e (E.8), respectivamente. A restrição (E.16) em conjunto com a restrição (E.17), visa gerar sobras quando $w_j = 1$ e ficam inativas quando $w_j = 0$. As restrições (E.18) e (E.19) juntas, definem perdas quando $w_j = 0$ e ficam inativas quando $w_j = 1$. Por fim, as restrições (E.20)-(E.22) definem as variáveis do problema.

Esta estratégia minimiza as perdas, na forma de uma variável, visando novamente a geração de sobras.

Estratégia 3

O modelo desenvolvido na terceira estratégia é mostrado a seguir:

$$\max \sum_{j=N-(R-1)}^N (L_j - \sum_{i=1}^m l_i a_{ij}) \quad (\text{E.23})$$

$$\text{s. a } \sum_{j=1}^N a_{ij} = d_i, \quad i = 1, \dots, m \quad (\text{E.24})$$

$$\sum_{i=1}^m l_i a_{ij} \leq L_j, \quad j = 1, \dots, J \quad (\text{E.25})$$

$$a_{ij} \geq 0 \text{ e inteiro, } i = 1, \dots, m, j = 1, \dots, J \quad (\text{E.26})$$

Observe que ao fazer $R \geq 1$, as perdas se concentram nos R últimos objetos. Para que o modelo priorize cortar objetos que possuam os menores comprimentos (estes são possivelmente sobras de outros processos de corte realizados anteriormente), uma ordenação não-crescente de seus comprimentos é efetuada, isto é, $L_1 \geq \dots \geq L_N$.

O objetivo (E.23) maximiza as perdas nos objetos de índices $j = N - (R - 1), \dots, N$. A restrição (E.24) garante que a quantidade de itens produzidos seja igual à demanda. A restrição (E.25) é uma restrição física de mochila e, a restrição (E.26) define a variável do problema. Observe que ao fazer $R = 1$, optou-se por concentrar as perdas no último objeto e, neste caso, foi feita uma ordenação decrescente dos comprimentos dos objetos, isto é, $L_1 \leq \dots \leq L_N$. Optou-se ordenar os objetos desta forma para que o modelo priorizasse cortar os objetos de comprimentos menores que s ao possivelmente sobras de outros processos de corte realizados anteriormente.

Esta estratégia maximiza as perdas em um número reduzido de objetos.

Estratégias em um Único Estágio

São propostas ainda duas outras estratégias para solução do PCES em um único estágio, cujas características são similares às de Abuabara (2006). Nestas, são exploradas as idéias contidas nos modelos das estratégias 1 e 2.

Estratégia 4

O modelo adotado na quarta estratégia é mostrado a seguir.

$$\min \sum_{j=1}^N y_j L_j \quad (\text{E.27})$$

$$\text{s. a } \sum_{j=1}^N w_j = R \quad (\text{E.28})$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in I \quad (\text{E.29})$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq w_j L_j + (1 - w_j)(S - 1), \quad j \in J \quad (\text{E.30})$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq w_j S, \quad j \in N \quad (\text{E.31})$$

$$a_{ij} \geq 0 \text{ e inteiro}, \quad i \in m, \quad j \in J \quad (\text{E.32})$$

$$y_j \in \{0, 1\}, \quad j \in J \quad (\text{E.33})$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (\text{E.34})$$

A função objetivo (E.27) minimiza o comprimento total cortado para atender a demanda. A restrição (E.28) garante que o número de objetos que o modelo usa para acumular perdas seja igual a R . A restrição (E.29) garante que a quantidade de itens produzidos seja igual à demanda. As restrições (E.30) e (E.31) são equivalentes às (E.9) (E.10) e, juntas indicam para cada objeto selecionado ($y_j = 1$), que o comprimento remanescente do corte dos itens, deve estar no intervalo $[S, L_j]$ quando $w_j = 1$ (o objeto deve concentrar perdas), ou deve estar no intervalo $[0, S)$ quando $w_j = 0$ (o objeto não deve concentrar perdas). Por fim, as restrições (E.32)-(E.34) definem as variáveis do problema.

Estratégia 5

A próxima estratégia consiste em explorar as condições do modelo da estratégia 2.

$$\min \sum_{j=1}^N (T_j + y_j L_j) \quad (E.35)$$

$$\text{s. a } \sum_{j=1}^N w_j = R \quad (E.36)$$

$$\sum_{j=1}^N a_{ij} = d_i, \quad i \in m \quad (E.37)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq L_j + (1 - w_j)M, \quad j \in N \quad (E.38)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq S - (1 - w_j)M, \quad j \in N \quad (E.39)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \leq T_j + w_j M, \quad j \in N \quad (E.40)$$

$$y_j L_j - \sum_{i=1}^m l_i a_{ij} \geq T_j - w_j M, \quad j \in N \quad (E.41)$$

$$a_{ij} \geq 0 \text{ e inteiro, } i \in m, j \in N \quad (E.42)$$

$$T_j \geq 0 \text{ e inteiro, } j \in N \quad (E.43)$$

$$y_j \in \{0, 1\}, \quad j \in J \quad (E.44)$$

$$w_j \in \{0, 1\}, \quad j \in J \quad (E.45)$$

A função objetivo (E.37) minimiza a perda e o comprimento total cortado para atender a demanda. A restrição (E.36) define o número de objetos onde serão acumuladas as perdas e a restrição (E.38) garante que a quantidade de itens produzidos seja igual à demanda. Para cada objeto selecionado ($y_j = 1$), a restrição (E.39) em conjunto com a restrição (E.40), visam gerar sobras quando $w_j = 1$ e ficam inativas quando $w_j = 0$, agindo de forma similar às restrições (E.16) e (E.17). Do mesmo modo que as restrições (E.18) e (E.19), as restrições (E.41) e (E.42) juntas, definem perdas quando $w_j = 0$ e ficam inativas quando $w_j = 1$. Por fim, as restrições (E.43)-(E.45) definem as variáveis do problema.

Simulações e Resultados

As cinco estratégias propostas foram implementadas, além do algoritmo iterativo proposto por Sinuany-Stern e Weiner (1994) e do modelo simplificado de Abuabara (2006). Foram feitas então simulações, utilizando um exemplo obtido de uma carteira da indústria de aeronaves, utilizado no trabalho de Abuabara (2006). Este exemplo também foi testado utilizando as heurísticas de Cherri (2006), as quais foram executadas pela própria autora: Heurística FFD Modificada, Heurística Gulosa Modificada, Heurística Residual FFD Modificada, Heurística Residual Gulosa Modificada e Heurística Nova Modificada (versões 1, 2 e 3).

Devido a um erro nos códigos implementados por Cherri (2006), algumas de suas heurísticas falharam ao obter as soluções dos exemplos selecionados. Desta forma, o termo “O código falhou” é utilizado na tabela para indicar que a solução não pode ser encontrada.

Todas as implementações foram realizadas no software Xpress-MP (versão 2008a1), utilizando a linguagem de modelagem Mosel. O teste foi realizado em um computador Intel Core 2 duo de 3 GHz e 4 GB de Ram.

Neste exemplo, o estoque é formado por 3 tipos de objetos, cujos dados estão na Tabela E.1. O comprimento de cada item e sua respectiva demanda são mostrados na Tabela E.2. Na Tabela E.3 são apresentadas as soluções obtidas por todas as estratégias.

Tabela E.1: Estoque do Exemplo 4.

Tipo de Objeto	Comprimento (u.c.)	Estoque
1	3000	3
2	3500	3
3	6000	3

Tabela E.2: Dados do Exemplo 4.

Tipo de Item	Comprimento (u.c.)	Demanda
1	80	20
2	30	10
3	50	10
4	70	5
5	10	20
6	12	20
7	14	20
8	15	40
9	20	10
10	28	10
11	30	35
12	31	20
13	32	10
14	33	20
15	35	15
16	36	5
17	39	10
18	40	5
19	43	20
20	65	5
21	75	5
22	90	25
23	100	5

No exemplo utilizado, o limite inferior para sobra é 10 u.c., assim, qualquer comprimento remanescente do objeto que não corresponda a um item demandado e que possua comprimento superior a 10 u.c. é considerado sobra.

Tabela E.3: Solução do Exemplo 4.

	Comp. total	N. de objetos (com)				Sobra	Perda
		cortados	aproveitamento total	sobra	perda		
M2P&R	16000	5	2	1	2	3190	5
M3P&R	15500	4	4	0	0	0	0
Estratégia1	13000	4	0	2	2	189	6
Estratégia2	13000	4	3	1	0	195	0
Estratégia3	13000	4	3	1	0	195	0
Estratégia4	13000	3	0	2	1	193	2
Estratégia5	13000	3	2	1	0	195	0
Sinuany	16000	5	4	1	0	3195	0
Abuabara	13000	4	3	1	0	195	0
FFD Mod	2	1	2	5	16000	5	3190
Gulosa Mod	0	1	4	5	16000	0	3195
FFD Res Mod			O código falhou				
Gulosa Res Mod			O código falhou				
Nova Mod 1			O código falhou				
Nova Mod 2			O código falhou				
Nova Mod 3			O código falhou				

Os dados da Tabela E.3 mostram que as soluções obtidas pelas Estratégias 4 e 5 são as que menos cortam objetos, sendo que a última não produz perda. Os resultados obtidos pelas Estratégias 2 e 3 são iguais aos obtidos pelo modelo simplificado de Abuabara (2006). Quanto às heurísticas propostas por Cherri (2008) e ao algoritmo iterativo proposto por Sinuany-Stern (1994), o comprimento total cortado foi superior ao obtido nas demais soluções, apesar de duas destas não gerarem perdas. Já a Estratégia 1 apresentou o pior resultado entre as abordagens que forneceram solução. Considerando que em situações práticas as indústrias privilegiam minimizar as perdas e gerar o máximo de sobras distribuídas no menor número de objetos, pode-se afirmar que, para este exemplo, as estratégias 2, 3 e 5 e a Abuabara tiveram êxito.

Discussão e Conclusão

Não existem métodos de soluções gerais e eficientes para os problemas de corte devido principalmente à complexidade computacional dos problemas e à diversidade de casos em que estes podem aparecer. Contudo, pode-se considerar que as estratégias de utilizar soluções ótimas ou quase-ótimas do PCER para obtenção de soluções para o PCES maximizando as sobras e distribuindo-as no menor número de objetos apresentaram bons resultados. Entretanto, tendo em vista o alto custo computacional envolvido no cálculo e análise de todos os padrões possíveis, uma nova abordagem está sendo desenvolvida, a qual utiliza uma técnica de geração de colunas para criação dos padrões de corte.

Nessa nova abordagem, os padrões de corte não precisam ser conhecidos a priori. Uma função objetivo é utilizada, a qual não faz distinção entre padrões de corte com perdas e com sobras. Esta distinção é feita nos geradores de colunas, os quais buscam utilizar o menor número possível de objetos. Então o modelo desenvolvido determina o momento mais adequado para utilizar um padrão com perda ou um padrão com sobra. O modelo sem limitação de estoque é apresentado a seguir:

$$\begin{aligned}
\min \quad & \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \\
\text{s. a} \quad & \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i \\
& x_{jp} \geq 0 \text{ e inteiro, } j \in J, p \in P(j)
\end{aligned}$$

Onde: L_j é o comprimento do objeto de índice j do estoque; $c_{jp} = c^{\text{processo}}(\sum_{i \in I} a_{ijp}) + c^{\text{retalho}}(L_j - \sum_{i \in I} a_{ijp} l_i)$, tal que c^{processo} é um custo associado ao processo de produção e c^{retalho} representa o custo de retalho, que pode ser c^{sobra} ou c^{perda} sendo c^{perda} e c^{sobra} o custo de perda e custo de sobra por unidade de comprimento, respectivamente; x_{jp} é a variável que representa o número de vezes que o padrão de corte de índice p no objeto de índice j foi aplicado; a_{ijp} é a quantidade de vezes que o item de comprimento l_i é cortado no objeto de índice j no padrão de corte de índice p ; d_i é a demanda de cada item de índice i .

A distinção entre os padrões é feita nos geradores de coluna. Os geradores de padrões que possuem perda e sobra são apresentados a seguir:

$$\begin{aligned}
\max Z_j^{\text{perda}} &= \sum_{i \in I} (\pi_i + c^{\text{perda}} l_i - c^{\text{processo}}) a_{ijp} - c^{\text{perda}} L_j \\
\text{sujeito a} \quad & \sum_{i \in I} a_{ijp} l_i \leq L_j \\
& \sum_{i \in I} a_{ijp} l_i \geq L_j - L_{\text{perda}_j} \\
& \sum_{i \in I} a_{ijp} \geq 1 \\
& a_{ijp} \geq 0 \text{ e inteiro, } p \in P(j)
\end{aligned}$$

$$\begin{aligned}
\max Z_j^{sobra} &= \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ijp} - c^{sobra} L_j \\
\text{sujeito a} & \sum_{i \in I} a_{ijp} l_i \leq L_j - L_{sobra_j} \\
& \sum_{i \in I} a_{ijp} l_i \geq L_j - L_{sobra_j}^{Max} \\
& \sum_{i \in I} a_{ijp} \geq 1 \\
& a_{ijp} \geq 0 \text{ e inteiro, } p \in P(j)
\end{aligned}$$

Onde: a_{ij} são as variáveis que representam o número de itens l_i cortados do objeto de índice j ; L_{perda} representa um comprimento de perda definido como aceitável; π_i são as coordenadas do multiplicador simplex; $L_{sobra_j}^{Max}$ é o comprimento máximo possível definido como sobra; $i = 1, \dots, I$; $j = 1, \dots, J$.

A cada iteração, um novo padrão é escolhido para integrar o problema principal, comparando-se os valores obtidos para Z_j^{perda} e Z_j^{sobra} , $j = 1, \dots, J$, até que o máximo destes valores seja menor ou igual a zero.

Nesta nova abordagem para o problema de corte de estoque, espera-se reduzir o tempo necessário para obtenção da solução, e ainda resolver problemas de maior escala.

Abstract. This paper introduces five strategies for solving the One-dimensional Cutting Stock Problem with Usable Leftovers, using one-stage and two-stage approaches. Results on simulations made with these strategies and with other papers of literature are presented. In addition, a new approach involving a column generation technique is introduced.

$$\begin{aligned}
\max Z_j^{sobra} &= \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ijp} - c^{sobra} L_j \\
\text{sujeito a} & \sum_{i \in I} a_{ijp} l_i \leq L_j - L_{sobra_j} \\
& \sum_{i \in I} a_{ijp} l_i \geq L_j - L_{sobra_j}^{Max} \\
& \sum_{i \in I} a_{ijp} \geq 1 \\
& a_{ijp} \geq 0 \text{ e inteiro, } p \in P(j)
\end{aligned}$$

Onde: a_{ij} são as variáveis que representam o número de itens l_i cortados do objeto de índice j ; L_{perda} representa um comprimento de perda definido como aceitável; π_i são as coordenadas do multiplicador simplex; $L_{sobra_j}^{Max}$ é o comprimento máximo possível definido como sobra; $i = 1, \dots, I$; $j = 1, \dots, J$.

A cada iteração, um novo padrão é escolhido para integrar o problema principal, comparando-se os valores obtidos para Z_j^{perda} e Z_j^{sobra} , $j = 1, \dots, J$, até que o máximo destes valores seja menor ou igual a zero.

Nesta nova abordagem para o problema de corte de estoque, espera-se reduzir o tempo necessário para obtenção da solução, e ainda resolver problemas de maior escala.

Abstract. This paper introduces five strategies for solving the One-dimensional Cutting Stock Problem with Usable Leftovers, using one-stage and two-stage approaches. Results on simulations made with these strategies and with other papers of literature are presented. In addition, a new approach involving a column generation technique is introduced.

Referências Bibliográficas

A. Abuabara, "Otimização no Corte de Tubos Estruturais: Aplicação na Indústria Aeronáutica Agrícola", Dissertação de Mestrado, Departamento de Engenharia de Produção, UFSCAR, São Carlos, SP, 2006.

A. Abuabara, R. Morabito, Modelos de programação inteira mista para o planejamento do corte unidimensional de tubos metálicos na indústria aeronáutica agrícola, *Gest. Prod., São Carlos*, **15**, No. 3 (2008), 605-617.

A.C. Cherri, "O Problema de Corte de Estoque com Reaproveitamento das Sobras de Material", Dissertação de Mestrado, Instituto de Ciências e Matemática Computacional, EESC, USP, São Carlos, SP, 2006.

A.C. Cherri, M.N. Arenales, H.H. Yanasse, The one-dimensional cutting stock problem with usable leftover - A heuristic approach, *European Journal of Operational Research*, (2008).

M. Gradizar, J. Jesenko, C. Resinovic, Optimization of roll cutting in clothing industry, *Computers & Operational Research*, **10**, (1997), 945-953.

M. Gradizar, M. Kljajic, C. Resinovic, J. Jesenko, A sequential heuristic procedure for one-dimensional cutting, *European Journal of Operational Research*, **114** (1999a), 557-568.

L.J. Slater, A hybrid approach for optimization of one-dimensional cutting, *European Journal of Operational Research* **119** (1999b), 719-728.

M. Gradizar, P. Trkman, A combined approach to the solution to the general one-dimensional cutting stock problem, *Computers & Operations Research*, **32** (2005), 1793- 1807.

S. Koch, S. König, G. Wäscher, Linear programming for a cutting Problem in the wood processing industry - a case study, *Optimization*, **14**, julho (2008).

G. Scheithauer, A note on handling residual lengths, *European Journal of Operational Research*, **22**, No. 3 (1991), 461-466.

Z. Sinuany-Stern, I. Weiner, The one dimensional cutting stock problem using two objectives, *J. Oper. Res. Soc. (UK)*, **45**, No. 2 (1994), 231-236.

G.M. Roodman, Near-optimal solutions to one-dimensional cutting stock problems, *Computers and Operations Research*, **13**, (1986), 713-719.

APÊNDICE F – Trabalho submetido à XII ONPCE

A décima terceira Oficina Nacional de Problemas de Corte, Empacotamento, Dimensionamento de Lotes e Programação da Produção (ONPCE) trata-se de um evento que reúne pesquisadores, professores e alunos com interesse na área de problemas de corte, empacotamento, dimensionamento de lotes e programação da produção. O trabalho submetido a esta oficina é apresentado a seguir.

Trabalho: “Problema de Corte de Estoque Unidimensional com Reaproveitamento de Sobras: abordagens de resolução por meio de dois geradores de colunas”

P.C. FARIAS, PGMAC, UEL - Universidade Estadual de Londrina, 86051-980 Londrina, PR, Brasil, R.S.V. HOTO, Departamento de Matemática, UEL - Univ Estadual de Londrina, 86051-980 Londrina, PR, Brasil.

Este trabalho apresenta duas modelagens para o Problema de Corte de Estoque com Reaproveitamento de Sobras, que consiste em cortar peças menores (itens) de objetos em estoque, disponíveis em quantidade suficiente para atender aos pedidos de clientes (demandas), e obter sobras deste processo que possam ser reutilizadas futuramente, minimizando uma função objetivo. Essas modelagens consideram objetos de estoque com comprimentos diversos, e utilizam como abordagem de resolução uma técnica de geração de colunas. Em cada uma das modelagens, são resolvidos dois tipos de subproblemas: um gerador de padrões com perdas aceitáveis (perdas que serão permitidas na solução), e um gerador de padrões com sobras (em intervalos pré-definidos). Na primeira modelagem, são solucionados dois conjuntos de subproblemas (com perda e com sobra para cada um dos objetos distintos em estoque). Na segunda, existem apenas dois geradores, de forma que o tipo de objeto faz parte da solução. Em ambas busca-se minimizar o custo total da produção, que envolve custo de processo de corte, custo de perda ou sobra de material. O modelo sem limitação de estoque, para a primeira modelagem, é apresentado a seguir:

$$\begin{aligned} &\text{minimizar} && \sum_{j \in J} \sum_{p \in P(j)} c_{jp} x_{jp} \\ &\text{sujeito a} && \sum_{j \in J} \sum_{p \in P(j)} a_{ijp} x_{jp} \geq d_i, \quad i \in I \\ &&& x_{jp} \geq 0 \text{ e inteiro, } j \in J, p \in P(j) \end{aligned}$$

Onde: L_j é o comprimento do objeto de índice j do estoque; $c_{jp} = c^{\text{processo}} \left(\sum_{i \in I} a_{ijp} \right) + c^{\text{retalho}} \left(L_j - \sum_{i \in I} a_{ijp} l_i \right)$, tal que c^{processo} é um custo associado ao processo de produção e c^{retalho} representa o custo de retalho, que pode ser c^{sobra} ou c^{perda} , onde o primeiro representa o custo de sobra e o segundo denota o custo de perda, ambos por unidade de comprimento; x_{jp} é a variável que representa o número de vezes que o padrão de corte de índice p no objeto de índice j foi aplicado; a_{ijp} é a quantidade de vezes que o item de comprimento l_i é cortado no objeto de índice j no padrão de corte de índice p ; d_i é a demanda de cada item de índice i .

Devido ao alto tempo computacional envolvido na resolução de problemas com grande número de variáveis, a Técnica de Geração de Colunas torna-se viável já que poucas colunas (padrões de corte) são consideradas e as demais são geradas quando necessário a cada iteração.

O processo de geração de colunas é dividido em dois geradores, apresentados a seguir: um que produz padrões com perdas aceitáveis e o outro que produz padrões com sobras.

$$\begin{aligned}
 \text{maximizar } Z_j^{perda} &= \sum_{i \in I} (\pi_i + c^{perda} l_i - c^{processo}) a_{ijp} - c^{perda} L_j \\
 \text{sujeito a} & \sum_{i \in I} a_{ijp} l_i \leq L_j \\
 & \sum_{i \in I} a_{ijp} l_i \geq L_j - L_{perda} \\
 & \sum_{i \in I} a_{ijp} \geq 1 \\
 & a_{ijp} \geq 0 \text{ e inteiro, } p \in P(j)
 \end{aligned}$$

$$\begin{aligned}
 \text{maximizar } Z_j^{sobra} &= \sum_{i \in I} (\pi_i + c^{sobra} l_i - c^{processo}) a_{ijp} - c^{sobra} L_j \\
 \text{sujeito a} & \sum_{i \in I} a_{ijp} l_i \leq L_j - L_{sobra} \\
 & \sum_{i \in I} a_{ijp} l_i \geq L_j - L_{sobra}^{Max} \\
 & \sum_{i \in I} a_{ijp} \geq 1 \\
 & a_{ijp} \geq 0 \text{ e inteiro, } p \in P(j)
 \end{aligned}$$

Onde: a_{ijp} são as variáveis que representam o número de itens l_i cortados do objeto de índice j no padrão de corte de índice p ; L_{perda} representa um comprimento de perda definido como aceitável; L_{sobra} representa o comprimento mínimo aceito como sobra; π_i são as coordenadas do multiplicador simplex; L_{sobra}^{Max} é o comprimento máximo possível definido como sobra; $i = 1, \dots, I$; $j = 1, \dots, J$.

Na segunda modelagem, o modelo considerado utiliza novas restrições relacionadas à limitação de objetos em estoque e à produção de itens em excesso. Os problemas geradores de padrões são modificados para anexar em sua solução o objeto de estoque utilizado.

Os modelos propostos determinam quais objetos e padrões são mais vantajosos de acordo com os dados de entrada fornecidos pelo planejador de produção.

Referências Bibliográficas

A. Abuabara, R. Morabito, Modelos de programação inteira mista para o planejamento do corte unidimensional de tubos metálicos na indústria aeronáutica agrícola, *Gest. Prod., São Carlos*, **15**, No. 3 (2008), 605-617.

A.C. Cherri, "O Problema de Corte de Estoque com Reaproveitamento das Sobras de Material", Dissertação de Mestrado, Instituto de Ciências e Matemática Computacional, EESC, USP, São Carlos, SP, 2006.

- M. Gradizar, J. Jesenko, C. Resinovic, Optimization of roll cutting in clothing industry, *Computers & Operational Research*, **10**, (1997), 945-953.
- M. Gradizar, P. Trkman, A combined approach to the solution to the general one-dimensional cutting stock problem, *Computers & Operations Research*, **32** (2005), 1793- 1807.
- S. Koch, S. König, G. Wäscher, Linear programming for a cutting Problem in the wood processing industry - a case study, *Optimization*, **14**, julho (2008).
- G. Scheithauer, A note on handling residual lengths, *European Journal of Operational Research*, **22**, No. 3 (1991), 461-466.
- Z. Sinuany-Stern, I. Weiner, The one dimensional cutting stock problem using two objectives, . *J. Oper. Res. Soc. (UK)*, **45**, No. 2 (1994), 231-236.