



UNIVERSIDADE ESTADUAL DE
CAMPINAS

Instituto de Matemática, Estatística e
Computação Científica

ANGELO ALIANO FILHO

**Novas Extensões de Técnicas de Escalarizações
no Problema de Corte Unidimensional Inteiro
Multiobjetivo**

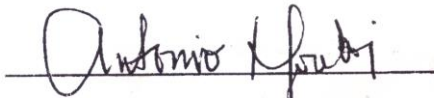
Campinas

2016

Angelo Aliano Filho

**Novas Extensões de Técnicas de Escalarizações no
Problema de Corte Unidimensional Inteiro Multiobjetivo**

Tese apresentada ao Instituto de Matemática,
Estatística e Computação Científica da Uni-
versidade Estadual de Campinas como parte
dos requisitos exigidos para a obtenção do
título de Doutor em Matemática Aplicada.



Orientador: Antonio Carlos Moretti

Este exemplar corresponde à versão
final da Tese defendida pelo aluno
Angelo Aliano Filho e orientada pelo
Prof. Dr. Antonio Carlos Moretti.

Campinas

2016

Agência(s) de fomento e nº(s) de processo(s): FAPESP, 2013/06035-0; CAPES

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

AL41n Aliano Filho, Angelo, 1988-
Novas extensões de técnicas de escalarizações no problema de corte unidimensional inteiro multiobjetivo / Angelo Aliano Filho. – Campinas, SP : [s.n.], 2016.

Orientador: Antonio Carlos Moretti.
Tese (doutorado) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Otimização multiobjetivo. 2. Problema do corte de estoque. 3. Heurística.
I. Moretti, Antonio Carlos, 1958-. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: New extensions of the scalarizations techniques in the multiobjective one-dimensional cutting stock problem

Palavras-chave em inglês:

Multi-objective optimization

Cutting stock problem

Heuristic

Área de concentração: Matemática Aplicada

Titulação: Doutor em Matemática Aplicada

Banca examinadora:

Antonio Carlos Moretti [Orientador]

Helenice de Oliveira Florentino Silva

Carla Taviane Lucke da Silva Ghidini

Antonio Roberto Balbo

Luiz Leduino de Salles Neto

Data de defesa: 17-03-2016

Programa de Pós-Graduação: Matemática Aplicada

Tese de Doutorado defendida em 17 de março de 2016 e aprovada

Pela Banca Examinadora composta pelos Profs. Drs.

Prof(a). Dr(a). ANTONIO CARLOS MORETTI

Prof(a). Dr(a). HELENICE DE OLIVEIRA FLORENTINO SILVA

Prof(a). Dr(a). CARLA TAVIANE LUCKE DA SILVA GHIDINI

Prof(a). Dr(a). ANTONIO ROBERTO BALBO

Prof(a). Dr(a). LUIZ LEDUINO DE SALLES NETO

A Ata da defesa com as respectivas assinaturas dos membros encontra-se no processo de vida acadêmica do aluno.

Aos meus pais...
Angelo e Loide Aliano...

Agradecimentos

A Deus pelo dom da vida eterna, saúde concedida, disposição, força e fé necessários para concluir mais uma jornada na vida.

Aos meus queridos pais Angelo e Loide Aliano, pelo apoio, educação, carinho, estímulo à vida acadêmica, pois sem eles não seria possível iniciar tampouco concluir este trabalho e aos demais familiares que sempre torceram por meu sucesso.

Ao meu caro orientador, o prof. Antônio Moretti, pela paciência em sua orientação, competência, incentivo, valiosos conselhos, sugestões, confiança dispensada e grande amizade que foram fundamentais para a elaboração deste trabalho.

Agradeço também à querida co-orientadora, a profa. Margarida Pato da Universidade de Lisboa/Portugal, em que seus conselhos, sugestões e correções foram fundamentais para o aperfeiçoamento deste trabalho. Também menciono o CIO - Centro de Investigação Operacional - por me amparar e dar todo suporte técnico para meu estágio em Lisboa.

À profa. Helenice da UNESP, ex-orientadora e amiga que sempre me apoiou e deu conselhos valiosíssimos em minha formação. Também não posso deixar de agradecer aos professores que tive ao longo deste doutorado, principalmente ao prof. Washington de Oliveira (FCA-UNICAMP) e prof. Leduíno Salles Neto (UNIFESP-São José dos Campos), que opinaram para o melhoramento desta tese no exame de qualificação realizado.

Ao Instituto de Matemática, Estatística e Computação Científica e a UNICAMP que forneceu todo o suporte técnico e computacional para a realização deste trabalho.

Agradeço a todos os meus amigos, companheiros e irmãos na fé da Congregação Cristã que conquistei durante estes anos: Fábio, Larissa, Priscila, Silvanira, João Crisóstomo, Eliene, Luiz Passoni e muitos outros. Agradeço a essas amigas que ampararam e sustentaram a minha fé e minha manutenção na Graça, que está acima de tudo, sendo também irrigada com as “reuniõezinhas” de segunda-feira na UNICAMP. Posso atribuir que esta força “invisível” e “infinita” que me trouxe até aqui, foi fundamental para o meu sucesso neste árduo e difícil trabalho.

À CAPES pelos quatro meses apoio financeiro.

À FAPESP processo 2013/06035-0, pela confiança depositada em mim e pelo

apoio financeiro concedido nos demais períodos da realização desta tese de doutorado. A FAPESP também me financiou, dentre tantos eventos e congressos, o meu estágio em Lisboa/Portugal pela Bolsa de Estágio de Pesquisa no Exterior, pelo processo 2014/22665-7.

À todas as pessoas que, direta ou indiretamente, contribuíram para a realização deste trabalho.

*“Não ajunteis tesouros na terra, onde a traça e a ferrugem tudo consomem,
e onde os ladrões minam e roubam,
mas ajuntai tesouros no céu, onde nem a traça nem a ferrugem consomem,
e onde os ladrões não minam nem roubam.
Porque onde estiver o vosso tesouro, aí estará também o vosso coração.
(Bíblia Sagrada, Mateus 6, 19-21)*

Resumo

O presente trabalho trata do Problema de Corte Unidimensional Inteiro Multiobjetivo (PCUIM). Este problema possui uma importância prática enorme e a sua abordagem multiobjetiva foi pouco reportada na literatura. O modelo biobjetivo considerado visa minimizar a soma das frequências dos padrões de corte para atender à mínima demanda e ao número de diferentes padrões a serem usados (*setup*), sendo estas metas conflitantes entre si. Neste caso, o PCUIM possui um conjunto não unitário de soluções, ditas de *soluções eficientes*, todas igualmente importantes para o problema. A geração de cada solução eficiente necessita a otimização de um Problema de Programação Linear Inteiro. Isso significa que obtenção de todas estas soluções pode ser uma tarefa cara, principalmente quando os padrões de corte não são fornecidos pelo usuário a priori. Neste trabalho, utilizamos sete métodos distintos de escalarizações, que transformam o PCUIM em vários subproblemas de otimização escalares, que por sua vez, geram as soluções eficientes. Seis métodos foram adaptados da literatura e um foi originalmente desenvolvido. A fim de acelerar a obtenção do conjunto de soluções eficientes, no caso com os padrões fornecidos pelo usuário, adotamos uma estratégia que relaxa as condições de integralidade das variáveis do problema e posteriormente, cada solução eficiente produzida é integralizada por meio de uma heurística que nós também propomos. Os extensos testes computacionais presentes no Capítulo 6, comprovaram que estas ideias foram adequadas e eficazes. Além disso, a nova técnica de escalarização se mostrou muito competitiva com as demais consagradas na literatura, possibilitando um crescimento e um avanço na área de Problemas de Corte bem como na Otimização Combinatória Multiobjetivo.

Palavras-chave: Métodos Clássicos de Otimização Multiobjetivo. Problema de Corte. Heurística de Arredondamento.

Abstract

The present work deals with the Multiobjective One-Dimensional Cutting Stock Problem (MODCSP). This problem has an enormous practical importance and the multiobjective approach has been little reported in the literature. The biobjective model considered aims to minimize the sum of the frequency of cutting patterns to meet minimal demand and the number of different cutting patterns to be used (setup), being these objectives conflicting. In this case, the MODCSP has a non-unitary set of solutions, said *efficient solutions*, equally important for the problem. The generation of each efficient solution requires the optimization of an Integer Programming Linear Problem. So, the complete enumeration of these solutions can be an expensive task, especially when cutting patterns are not provided by the user. In this work, we applied seven different methods that transform the MODCSP on scalar optimization problems, where each problem provide an efficient solution. Six scalarization methods were adapted from literature and one was unprecedentedly developed. In the case of the cutting patterns be provided a priori, we used a relaxation strategy (heuristic) to accelerate obtaining of the set efficient solutions. In this approach, we relaxed the integrality condition of the variables and each efficient solution was rounded by a specially developed heuristic. The extensive results in Chapter 6 validated that this idea was adequate and effective. Furthermore, the new scalarization technique proved to be very competitive with other established in the literature, enabling growth and advancement in the area of the Cutting Problems and in Multiobjective Combinatorial Optimization.

Keywords: Classical Methods of Multiobjective Optimization. Cutting Stock Problems. Rounding Heuristics.

Lista de ilustrações

Figura 1 – Possíveis alternativas para compra de um carro - Exemplo 1.1.14	32
Figura 2 – Região admissível para o Exemplo 1.1.15	33
Figura 3 – Soluções suportadas, não suportadas e fracamente não dominadas	34
Figura 4 – Pontos notáveis no espaço \mathcal{Z}	35
Figura 5 – Metas em OM	36
Figura 6 – Exemplo de padrões de corte	40
Figura 7 – Exemplo de aplicação da Soma Ponderada para problemas não convexos	61
Figura 8 – Exemplo de aplicação da Soma Ponderada para problemas convexos	62
Figura 9 – Ilustração de soluções dominadas obtidas com alguns pesos nulos	63
Figura 10 – Bases e soluções eficientes em um Problema de Programação Linear biobjetivo	64
Figura 11 – Triângulo inicial que contém todos os pontos não dominados para (3.2)	65
Figura 12 – Estratégia de ponderação do método NISE	66
Figura 13 – Novas soluções eficientes sendo obtidas com o NISE	67
Figura 14 – Todas soluções eficientes de um POM detectadas	67
Figura 15 – Interpretação geométrica da Métrica de Tchebycheff Ponderada	69
Figura 16 – Pontos na parte não convexa determinados pelo ε -R	72
Figura 17 – Interpretação do Problema Restrito Modificado	73
Figura 18 – Interpretação geométrica do método de Benson	76
Figura 19 – Área determinada pelas soluções não dominadas	81
Figura 20 – Tamanho de passo limite para satisfazer a restrições de não negatividade	97
Figura 21 – Tamanho de passo limite para satisfazer a restrição H	98
Figura 22 – Tamanho de passo limite para satisfazer as restrições $Ax \geq d$	99
Figura 23 – Tamanho de passo sendo reduzido com a atualização de H	101
Figura 24 – Região admissível e o primeiro hiperplano para o Exemplo 4.2.1	101
Figura 25 – Direção δ^1 sendo explorada no Exemplo 4.2.1	103
Figura 26 – Direção δ^2 sendo explorada e soluções inteiras no Exemplo 4.2.1	105
Figura 27 – Intervalos de mínimo e máximo no espaço \mathcal{Z}^* das soluções eficientes	110
Figura 28 – Exemplo de aplicação da Soma Ponderada	112
Figura 29 – Método Métrica de Tchebycheff no Problema de Corte multiobjetivo	114

Figura 30 – Pontos de referência e nadir modificados no Problema de Tchebycheff Sem Ponderação	116
Figura 31 – Novo ponto não dominado obtido com o Problema de Tchebycheff Sem Ponderação	117
Figura 32 – Soluções eficientes sendo obtidas pelo método de Tchebycheff Sem Ponderações	117
Figura 33 – Fronteira de Pareto para o problema do Exemplo 5.3.4	121
Figura 34 – ε -Restrito no Problema de Corte multiobjetivo para $\varepsilon = \varepsilon_1$	126
Figura 35 – Interpretação do Teorema de Benson Modificado	130
Figura 36 – Ilustração de pontos dominados no primeiro subproblema	135
Figura 37 – Ilustração das soluções excluídas com o segundo subproblema	136
Figura 38 – Ilustração das soluções excluídas do terceiro subproblema	137
Figura 39 – Comparação dos tempos computacionais (em seg.) utilizados para obter a fronteira de Pareto para o PCUIM - Fase 1	153
Figura 40 – Exemplo de uma fronteira de Pareto para o PCUIM para um problema da classe 8	155
Figura 41 – Tempo computacional demandado por cada subproblema em termos de w	164
Figura 42 – Comparação dos tempos computacionais demandados por cada subproblema no ε -Restrito e Benson	165
Figura 43 – Tempo computacional demandado por cada subproblema de Sylva & Crema em função do número de soluções eficientes	166
Figura 44 – Comparação dos tempos computacionais (em seg.) utilizados para obter \mathcal{Z}^*	167
Figura 45 – Comparação dos tempos computacionais (em seg.) utilizados para obter $\bar{\mathcal{Z}}$	167
Figura 46 – Comparação dos tempos computacionais (em seg.) utilizados para obter $\bar{\mathcal{Z}}$ e \mathcal{Z}^*	168
Figura 47 – Fronteiras de Pareto obtidas pela Soma Ponderada (esq.), Métrica de Tchebycheff (centro) e Tchebycheff Sem Ponderações (dir.)	170
Figura 48 – Comparação dos tempos computacionais levados pelo ε -Restrito para $p = 80, 120, 160$ e 200 colunas	172
Figura 49 – Comparação dos tempos computacionais levados pelo Tchebycheff Sem Ponderações para $p = 80, 120, 160$ e 200 colunas	172
Figura 50 – Comparação dos tempos computacionais (em seg.) utilizados para obter $\bar{\mathcal{Z}}$ e \mathcal{Z}^*	177
Figura 51 – Comparação dos custos-benefícios dos métodos de escalarização: $ \bar{\mathcal{Z}} \times \text{Tempo}^{-1}$	178
Figura 52 – Fronteira de Pareto \mathcal{Z}^* , $\tilde{\mathcal{Z}}$ e $\bar{\mathcal{Z}}$ para um exemplar da classe 9	180
Figura 53 – Fronteira de Pareto \mathcal{Z}^* , $\tilde{\mathcal{Z}}$ e $\bar{\mathcal{Z}}$ para um exemplar da classe 11	180

Figura 54 – Fronteiras de Pareto típicas determinadas pela Soma Ponderada e Tchebycheff Sem Ponderações para um exemplar com $m = 200$	181
Figura 55 – Diferentes aproximações da Fronteira de Pareto para $\theta = 50$ (18 soluções) à esq., $\theta = 75$ (27 soluções) ao centro e $\theta = 100$ (31 soluções) à dir. . .	185
Figura 56 – Diferentes aproximações da Fronteira de Pareto conforme m é modificado: $m = 20$ (15 soluções) à esq., $m = 40$ (31 soluções) ao centro e $m = 60$ (33 soluções) à dir.	187
Figura 57 – Diferentes fronteiras de Pareto conforme p varia	188
Figura 58 – Relação do número de soluções obtidas por segundo - Categoria 1 . . .	189
Figura 59 – Relação do número de soluções obtidas por segundo - Categoria 2 . . .	190
Figura 60 – Relação do número de soluções obtidas por segundo - Categoria 3 . . .	191

Lista de tabelas

Tabela 1 – Número de combinações de corte e padrões usados na solução ótima conforme ℓ varia	42
Tabela 2 – Dados para dois exemplares para o PCUI	50
Tabela 3 – Número de variáveis e restrições para o PCUI para dois exemplos numéricos	50
Tabela 4 – Dados para dois exemplares para o PCUI	51
Tabela 5 – Número de variáveis e restrições para o PCUI para dois exemplos numéricos	51
Tabela 6 – Três soluções eficientes obtidas para o Exemplo 5.8.1 e seus respectivos valores objetivos	144
Tabela 7 – Classes para o PCUIM - Fase 1	149
Tabela 8 – Valores médios para a Métrica σ^4 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1	150
Tabela 9 – Valores médios para a Métrica σ^5 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1	150
Tabela 10 – Valores médios para a Métrica σ^6 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1	151
Tabela 11 – Tempos computacionais médios (em seg.) para resolver cada subproblema para os métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1	151
Tabela 12 – Tempos computacionais médios (em seg.) para os métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - fase 1	152
Tabela 13 – Número médio de variáveis e restrições do PCUIM - Fase 1	153
Tabela 14 – Valores médios para o <i>gap</i> e número de nós para os subproblemas no ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1	154
Tabela 15 – Quatro diferentes soluções para o PCUIM com os padrões não fornecidos a priori	156
Tabela 16 – Classes para o PCUIM - Fase 2	157
Tabela 17 – Resultados computacionais para σ^1 - Fase 2	158
Tabela 18 – Resultados computacionais para σ^2 - Fase 2	159
Tabela 19 – Resultados computacionais para σ_{11}^3 , σ_{12}^3 , σ_{21}^3 e σ_{22}^3 - Fase 2	160

Tabela 20 – Resultados computacionais para $\sigma_1^4 \times 10^{-4}$ - Fase 2	161
Tabela 21 – Resultados computacionais para $\sigma_2^4 \times 10^{-4}$ - Fase 2	162
Tabela 22 – Resultados computacionais para σ_1^5 - Fase 2	162
Tabela 23 – Resultados computacionais para σ_2^5 - Fase 2	163
Tabela 24 – Tempos computacionais médios (em seg.) para obtenção de \mathcal{Z}^* - Fase 2	164
Tabela 25 – Tempos computacionais médios (em seg.) para obtenção de $\tilde{\mathcal{Z}}$ - Fase 2	166
Tabela 26 – Resultados computacionais comparando todos os métodos e com mais padrões de corte	169
Tabela 27 – Comparação das técnicas ε -Restrito e Tchebycheff Sem Ponderações nas abordagens exata e heurística com $p = 80, 120, 160$ e 200 padrões de corte	171
Tabela 28 – Classes para o PCUIM - Fase 3	173
Tabela 29 – Resultados computacionais para σ^1 - Fase 3	174
Tabela 30 – Resultados computacionais para σ_{12}^3 e σ_{22}^3 - Fase 3	175
Tabela 31 – Resultados computacionais para $\frac{\sigma_2^4}{\sigma_1^4}$ - Fase 3	175
Tabela 32 – Resultados computacionais para σ_1^5 - Fase 3	176
Tabela 33 – Tempos computacionais médios (em seg.) para obter as fronteiras $\tilde{\mathcal{Z}}$ - Fase 3	179
Tabela 34 – Tempos computacionais médios (em segundos) utilizados pela HADS - Fase 3	182
Tabela 35 – Resultados computacionais médios para a Categoria 1, com $\theta = 50, 75$ e 100 - Fase 4	185
Tabela 36 – Resultados computacionais médios para a Categoria 2, com $m = 20, 40$ e 60 - Fase 4	186
Tabela 37 – Resultados computacionais médios para a Categoria 3. com $p = m, 2 \cdot m$ e $3 \cdot m$ - Fase 4	187
Tabela 38 – Tempos computacionais médios (em segundos) para a Categoria 1. com $\theta = 50, 75$ e 100 - Fase 4	188
Tabela 39 – Tempos computacionais médios (em seg.) para a Categoria 2. com $m = 20, 40$ e 60 - Fase 4	189
Tabela 40 – Tempos computacionais médios (em seg.) para a Categoria 3, com $p = m, 2 \cdot m$ e $3 \cdot m$ - Fase 4	190
Tabela 41 – Resultados computacionais para o algoritmo Gilmore e Gomory - Fase 2	203
Tabela 42 – Resultados computacionais para o algoritmo Gilmore e Gomory - Fase 3	204

Lista de abreviaturas e siglas

PCUI:	Problema de Corte Unidimensional Inteiro
PCUIM:	Problema de Corte Unidimensional Inteiro Multiobjetivo
OM:	Otimização Multiobjetivo
POM:	Problema de Otimização Multiobjetivo
SP:	Soma Ponderada
Tch:	Método de Tchebycheff
Tch-SP:	Tchebycheff Sem Ponderações
ε -R:	ε -Restrito
NISE:	<i>Non Inferior Set Estimation</i>
Ben:	Método de Benson
S-C:	Método de Sylva & Crema
GC:	Geração de Colunas
HADS:	Heurística de Arredondamento por Direções Simplex

Lista de símbolos

\mathbb{N} :	Conjunto dos números naturais
\mathbb{Z} :	Conjunto dos números inteiros
\mathbb{R} :	Conjunto dos números reais
\mathbb{R}_+ :	Conjunto dos números reais positivos
\mathbb{B} :	Conjunto binário, isto é, $\mathbb{B} = \{0, 1\}$
X^n :	Produto cartesiano, isto é, $X^n = X \times \dots \times X$
$\ x\ _q$:	Norma- q do vetor x , isto é, $\ x\ _q = \sqrt[q]{\sum_{i=1}^n x_i ^q}$
$\ x\ _\infty$:	Norma- ∞ (ou de Tchebycheff) do vetor x , isto é, $\ x\ _\infty = \max_{1 \leq i \leq n} \{ x_i \}$
$x = (x_1, \dots, x_n)^T \in \mathbb{Z}_+^n$:	vetor cuja entrada j indica a frequência do j -ésimo padrão de corte
$\tilde{x} = (\tilde{x}_1, \dots, \tilde{x}_n)^T \in \mathbb{R}^n$	vetor proveniente da solução de um problema de programação matemática relaxado
$\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^T \in \mathbb{Z}_+^n$	vetor integralizado por um procedimento heurístico, a partir de \tilde{x}
$x^* = (x_1^*, \dots, x_n^*)^T \in \mathbb{Z}_+^n$	vetor proveniente da solução de um problema de programação matemática inteiro
$y = (y_1, \dots, y_n)^T \in \mathbb{B}^n$:	vetor binário cuja entrada j indica se o j -ésimo padrão de corte é usado ou não
$f = (f_1, \dots, f_r)^T$:	Vetor função objetivo r -dimensional
$w = (w_1, \dots, w_r)^T$:	Vetor de pesos
$a_{jk} \in \mathbb{N}^m$:	Vetor m -dimensional representando o padrão de corte j da bobina mestre k

m :	Número de itens demandados
n_k :	Número total de padrões de corte factíveis segundo a bobina mestre k
$p < n_k$:	Número de padrões de corte factíveis a ser considerado no PCUI com os padrões dados a priori
b :	Número de bobinas mestres com larguras distintas disponíveis
q :	Número máximo de facas permitido para se construir um padrão de corte
$I = \{1, \dots, m\}$:	Conjunto dos itens demandados
$J_k = \{1, \dots, n_k\}$:	Conjunto total de padrões de corte segundo a bobina mestre k
J_B :	Conjunto de índices de padrões de corte básicos
$P = \{1, \dots, p\}$:	Conjunto de padrões de corte
$K = \{1, \dots, b\}$:	Conjunto das bobinas mestres
L_k :	Largura da bobina mestre $k \in K$
ℓ_i :	Largura do item $i \in I$
$\lceil a \rceil$:	Maior inteiro menor ou igual à a
$\lfloor a \rfloor$:	Menor inteiro maior ou igual à a
$ X $:	Cardinalidade do conjunto X
$\min(\cdot)$:	Função mínimo
$\max(\cdot)$:	Função máximo
B :	Base factível para um Problema de Programação Linear
$\pi^T = c_B^T B^{-1}$:	Multiplicador Simplex
$\mathbf{1}^n = (1, \dots, 1)^T$:	Vetor unitário
$\mathcal{X} \subseteq \mathbb{R}^n$:	Conjunto factível para um problema de otimização
$\mathcal{Z} = f(\mathcal{X}) \subseteq \mathbb{R}^r$:	Imagem do conjunto \mathcal{X} ou espaço de critério
$\mathcal{X}^* \subseteq \mathbb{R}^n$:	Conjunto eficiente
$\mathcal{Z}^* = f(\mathcal{X}^*) \subseteq \mathbb{R}^r$:	Imagem do conjunto \mathcal{X}^* ou conjunto não-dominado
$\tilde{\mathcal{X}} \subseteq \mathbb{R}^n$:	Conjunto eficiente relaxado

$\tilde{\mathcal{Z}} \subseteq \mathbb{R}^r$:	Imagem de $\tilde{\mathcal{X}}$, isto é, $\tilde{\mathcal{Z}} = f(\tilde{\mathcal{X}})$
$\bar{\mathcal{X}} \subseteq \mathbb{Z}_+^n$:	Conjunto eficiente obtido integralizado a partir de $\tilde{\mathcal{X}}$
$\bar{\mathcal{Z}} \subseteq \mathbb{R}^r$:	Imagem de $\bar{\mathcal{X}}$, isto é, $\bar{\mathcal{Z}} = f(\bar{\mathcal{X}})$
$z^I \in \mathbb{R}^r$:	Vetor ideal
$z^N \in \mathbb{R}^r$:	Vetor nadir
$z^U \in \mathbb{R}^r$:	Vetor utópico
$x < y$:	$\Leftrightarrow x_i < y_i$ para todo $i = 1, \dots, n$
$x \leq y$:	$\Leftrightarrow x_i \leq y_i$ para todo $i = 1, \dots, n$
$x \leq y$:	$\Leftrightarrow x_i \leq y_i$ e $x_j < y_j$ para algum $j = 1, \dots, n$
$\rho > 0$:	constante positiva
Δ :	Tamanho de passo nas técnicas de escalarizações
θ :	Parâmetro que controla o tamanho de passo
δ^j :	j -ésima direção simplex
σ^i :	i -ésima métrica comparativa
cv :	Coefficiente de variação.

Lista de algoritmos

Algoritmo 1 – Método primal simplex	87
Algoritmo 2 – Método Geração de Colunas de Gilmore e Gomory	89
Algoritmo 3 – Heurística de Arredondamento por Direções Simplex	106
Algoritmo 4 – O Método Soma Ponderada para o PCUIM	110
Algoritmo 5 – O Método da Métrica de Tchebycheff para o PCUIM	114
Algoritmo 6 – O Método Tchebycheff Sem Ponderações para o PCUIM	120
Algoritmo 7 – O Método ε -Restrito para o PCUIM	126
Algoritmo 8 – O Método NISE híbrido com ε -Restrito para o PCUIM	128
Algoritmo 9 – O Método de Benson para o PCUIM	131
Algoritmo 10 – O Método de Silva & Crema para o PCUIM	138
Algoritmo 11 – Filtro de Pareto	202
Algoritmo 12 – Construção de um Padrão de Corte Admissível	202

Sumário

Introdução	23
0.1 Breve revisão de trabalhos correlatos na área	24
I REFERENCIAIS TEÓRICOS	27
1 A OTIMIZAÇÃO MULTIOBJETIVO	28
1.1 Conceitos e definições	28
2 MODELAGEM MATEMÁTICA DO PROBLEMA DE CORTE	37
2.1 Abordagem mono-objetiva do problema de corte	39
2.2 Técnicas de linearização	46
2.2.1 Linearização usando as variáveis x_{jk}	46
2.2.2 Linearização usando as variáveis a_{ijk}	48
2.3 Abordagem multiobjetiva do problema de corte	51
2.3.1 Motivação	51
2.3.2 Modelo multiobjetivo sem os padrões de corte a priori	53
2.3.3 Modelo multiobjetivo com os padrões de corte a priori	55
3 MÉTODOS CLÁSSICOS DE OTIMIZAÇÃO MULTIOBJETIVO	58
3.1 O Método da Soma Ponderada	58
3.2 O Método <i>Non Inferior Set Estimation</i>	63
3.2.1 Preliminares	63
3.2.2 Funcionamento do procedimento	64
3.3 O Método da Métrica de Tchebycheff	68
3.4 O Método do ε -Restrito	71
3.4.1 Conexões da Soma Ponderada com o ε -Restrito	74
3.5 O Método de Benson	75
3.6 O Método de Silva & Crema ((SYLVA; CREMA, 2004))	77
3.7 Métricas de Comparação	79
II DESENVOLVIMENTO DA PESQUISA	82
4 UMA NOVA HEURÍSTICA DE ARREDONDAMENTO PARA O PCUI	83
4.1 Revisão de métodos heurísticos para o PCUI	84

4.1.1	O método de Geração de Colunas	84
4.1.2	Heurística de arredondamento simples e melhorado	91
4.1.3	Heurística de um método exato nos padrões básicos	92
4.2	Heurística de Arredondamento por Direções Simplex	94
4.2.1	Inicialização	95
4.2.2	Identificação do tamanho de passo	96
4.2.2.1	Restrições de não negatividade	97
4.2.2.2	Restrição do hiperplano H	97
4.2.2.3	Restrições $Ax \geq d$	99
4.2.3	Busca na direção δ^j e atualização de α^{max}	100
4.2.4	Limitantes para a função-objetivo	106
5	OS MÉTODOS DE ESCALARIZAÇÕES PROPOSTOS NO PROBLEMA DE CORTE MULTIOBJETIVO	108
5.1	O Método Soma Ponderada	110
5.2	O Método da Métrica de Tchebycheff	112
5.3	O Método Tchebycheff Sem Ponderações	115
5.4	O Método ε -Restrito	125
5.5	O Método NISE Híbrido com ε -Restrito	127
5.6	O Método Benson	128
5.7	O Método de Silva & Crema	133
5.8	Relaxação com os padrões básicos	138
6	TESTES COMPUTACIONAIS	145
6.1	Resultados computacionais sem os padrões de corte a priori - Fase 1	148
6.2	Resultados computacionais com os padrões de corte a priori - Fase 2	155
6.2.1	Testes com colunas adicionais	168
6.3	Resultados computacionais com os padrões de corte a priori - Fase 3	172
6.4	Resultados computacionais com os coeficientes nas funções-objetivo diferente de 1 - Fase 4	183
III	CONSIDERAÇÕES FINAIS	192
7	CONCLUSÕES E PERSPECTIVAS	193
7.1	Principais contribuições deste trabalho	195
7.2	Trabalhos futuros	195
	REFERÊNCIAS	197

Introdução

A otimização é um processo de busca que visa encontrar uma ou mais soluções factíveis que correspondem a pontos extremos de uma ou mais funções-objetivos. Tendo em vista as excelentes qualidades dos otimizadores de um problema de programação matemática, os métodos de otimização exercem uma enorme função prática nos mais variados e tipos de problemas de cunho prático e econômico.

Quando o problema de otimização analisado envolve apenas uma função-objetivo ou critério, ele é chamado de *problema de otimização mono-objetivo* ou *escalar*. Existem um número enorme de métodos na literatura que tratam especificamente estes problemas, desenvolvidos principalmente após a Segunda Guerra Mundial.

Contudo, existem problemas que podem levar em conta não apenas uma função-objetivo e sim múltiplas. A maioria dos problemas na vida prática são multiobjetivos, principalmente quando existe um certo grau de conflito entre os objetivos. As aplicações destes problemas são variadas, desde o campo da biologia, física, química, engenharia e assim, o desenvolvimento de técnicas mais robustas e eficazes são necessárias. Denominamos estes tipos específicos de problemas como *Problemas de Otimização Multiobjetivo* (POM), *Multicritério* ou até mesmo *Vetorial*, pois o ente matemático a ser otimizado é um vetor. A simplificação destes problemas utilizando-se um único critério (função), pode gerar perda de informações para o tomador de decisão.

A literatura específica na área de OM relata os métodos de Soma Ponderada e ε -Restrito como as abordagens mais utilizadas; no entanto, além da aplicação destas técnicas de escalarizações, pretendemos utilizar, testar e comparar outras abordagens, como a Métrica Tchebycheff, um método híbrido de Soma Ponderada Modificada com ε -Restrito, método de Benson e um método desenvolvido em (SYLVA; CREMA, 2004). Além destes seis métodos, propomos um novo procedimento de escalarização, advindo e aproveitando as ideias da Métrica de Tchebycheff, capaz de obter todas as soluções eficientes de um problema de otimização biobjetivo discreto. A pergunta que pode surgir é: “*Esta nova técnica é competitiva com as demais abordagens presentes na literatura, em termos de custo computacional e potencial para detectar todas as soluções do problema?*”. A resposta a esta e outras questões são dadas e convalidadas neste trabalho, atestando uma ideia inovadora e que mostrou uma contribuição na área de problemas de corte e em

OM.

0.1 Breve revisão de trabalhos correlatos na área

A grande parte dos métodos de OM para problemas combinatórios utilizam metaheurísticas, mais especificamente, os Algoritmos Genéticos e suas variações. A literatura é rica nessa área, e pontuamos a referência (DEB; KALYANMOY, 2001) como principal trabalho.

Quase todos os métodos foram pioneiramente desenvolvidos na década de 70, como os apresentados em (BENSON, 1978), (BOWMAN, 1976) e (COHON, 2003). Uma revisão de técnicas de programação multiobjetivo foi dada em (EVANS, 1984). Da década de 70 até os dias atuais, este ramo da pesquisa operacional tem recebido diversificadas atenções, com a descoberta e aprimoramento das mais variadas técnicas. Os trabalhos apresentados em (CLÍMACO; FERREIRA; CAPTIVO, 1997), (EHRGOTT; GANDIBLEUX, 2000) e (RUZIKA S.AND WIECEK, 2003) fazem uma revisão das pesquisas e avanços mais recentes na área.

Do ponto de vista matemático, interessamos estudar e aprofundar o uso de técnicas exatas para a caracterização de soluções eficientes. Temos muitos trabalhos nesta área. A classificação dos métodos é feita mediante os conceitos metodológicos que usam. Por exemplo, os métodos em (CLÍMACO; FERREIRA; CAPTIVO, 1997) e (KLEIN; HANNAN, 1982) especializaram-se em encontrar uma ramificação ideal no *Branch-and-Bound*. Outros, propuseram normas ponderadas, como em (CLÍMACO; FERREIRA; CAPTIVO, 1997), (DäCHERT; GORSKI; KLAMROTH, 2012), (KETTANI; AOUNIÏD; MARTEL, 2004), (NEUMAYER; SCHWEIGERT, 1994), (RALPHS; SALTZMAN; WIECEK, 2006), (SAYIN; KOUVELIS, 2005), (SCHANDL; KLAMROTH; WIECEK, 2001), (SOLANKI, 1991), e outros métodos utilizam restrições, como em (BÉRUBÉ; GENDREAU; POTVIN, 2009), (CHALMET; LEMONIDIS; ELZINGA, 1986), (MAVROTAS; FLORIOS, 2013), (OZLEN; BURTON; MACRAE, 2014) e (ÖZLEN; AZIZOGLU, 2009), para determinação das soluções eficientes.

O artigo proposto em (STEUER; CHOO, 1983) supera algumas dificuldades do método original apresentado em (BOWMAN, 1976) e desenvolve um método iterativo baseado na Métrica de Tchebycheff, onde no transcorrer das iterações, o decisor influencia determinadas regiões do espaço de busca onde queira enfatizar a busca por soluções eficientes.

O método proposto em (SYLVA; CREMA, 2004) foi desenvolvido para problemas combinatórios com múltiplos objetivos, que foi inspirado no trabalho pioneiro em (KLEIN; HANNAN, 1982). A partir de um conjunto de soluções eficientes (e dos pontos não dominados associados) a priori, outras soluções são obtidas, eliminando-se do

conjunto admissível dos subproblemas subsequentes as regiões do espaço de critério que são dominadas por estes pontos não dominados. O resultado é a obtenção de todas as soluções eficientes para o problema. Este método, foi implementado, analisado e comparado neste trabalho com as demais técnicas.

O artigo em (PRZYBYLSKI; GANDIBLEUX; EHRGOTT, 2009) pontua e compara quatro métodos distintos para o Problema da Atribuição Triobjetivo, traçando um paralelo com o mesmo método em (SYLVA; CREMA, 2004).

O método desenvolvido em (MIETTINEN et al., 2009) descreve um método para múltiplos objetivos que utiliza pontos de referência, níveis de aspirações e ajuste dos pesos nas funções-objetivos escalarizadas para determinação do conjunto eficiente. Similarmente, o trabalho em (LUQUE, 2015) emprega quase os mesmos conceitos, no entanto para problemas não lineares contínuos não convexos.

Ressaltamos, no entanto, que os trabalhos apresentados em (RALPHS; SALTZMAN; WIECEK, 2006) e (SOLANKI, 1991) são as pesquisas que mais influenciaram o presente estudo, no que tange ao novo método de escalarização que estamos propondo. O artigo em (SOLANKI, 1991) desenvolve um método de Tchebycheff para problemas discretos e biobjetivos que se inspira no método de *Non Inferior Set Estimation* (NISE), proposto em (COHON, 2003). Neste método, as soluções eficientes são calculadas por meio da Métrica de Tchebycheff, utilizando os pontos não dominados consecutivos e modificando iterativamente o ponto de referência (aspiração). A desvantagem desta técnica é que vários subproblemas podem fornecer a mesma solução eficiente, já obtida de iterações anteriores. Uma proposta de melhora foi encontrada recentemente no trabalho em (RALPHS; SALTZMAN; WIECEK, 2006) de 2015. Este método faz uso do mesmo princípio, gerando novas soluções eficientes, utilizando como guia os pontos consecutivos no espaço de critério. As diferentes soluções são obtidas ao modificar o ponto de referência. Além disso, uma estratégia de “precisão” para a fronteira de Pareto é empregada.

As principais diferenças entre o novo método aqui desenvolvido e o presente nestes dois trabalhos anteriores são (i) a ausência de pesos que controlam os desvios em relação ao ponto de referência e as funções-objetivos envolvidas, (ii) a forma de atualização dos pontos de referência e (iii) critério de parada. Embora fazem uso das mesmas ideias do ponto de vista matemático, estes três aspectos são fundamentais para caracterizar de inovador o método que propomos.

Este trabalho está organizado do seguinte modo:

- no Capítulo 1, damos as noções e terminologias básicas de Otimização Multiobjetivo.
- no Capítulo 2, apresentamos e descrevemos um clássico problema de otimização combinatória: o Problema de Corte. Neste problema iremos aplicar os métodos de

escalarização a serem desenvolvidos neste trabalho. Também destacamos a importância deste problema dentro da área de Pesquisa Operacional, bem como ressaltamos algumas ideias que tornam plausíveis a implementação dos métodos propostos neste problema.

- No Capítulo 3, apresentamos os métodos clássicos de OMs e os referenciais teóricos que garantem a eficiência das soluções geradas pelas técnicas de escalarizações.
- No Capítulo 4, revemos o método de Gilmore-Gomory (([GILMORE; GOMORY, 1961](#)) e ([GILMORE; GOMORY, 1963](#))), extensamente utilizado neste trabalho, bem como outras metodologias heurísticas para resolver o Problema de Corte mono-objetivo. Também propomos uma nova técnica heurística de arredondamento para resolver o Problema de Corte, ainda na versão mono-objetiva. Esta nova técnica será adaptada para gerar soluções eficientes para o Problema de Corte Multiobjetivo.
- No Capítulo 5, descrevemos as modificações realizadas nas técnicas apresentadas no Capítulo 3 para resolver o Problema de Corte Multiobjetivo. Também, apresentamos o novo método de escalarização - Tchebycheff Sem Ponderações - e seu funcionamento é ilustrado com um exemplo numérico.
- O Capítulo 6 apresenta todos os experimentos computacionais realizados, divididos em quatro fases distintas, necessários para convalidar nossas propostas apresentadas.
- No Capítulo 7, pontuamos algumas conclusões, inovações e futuras direções de pesquisas, a curto e longo prazo.

Parte I

Referenciais teóricos

Capítulo 1

A otimização multiobjetivo

No presente capítulo, revisamos os principais conceitos e definições da Otimização Multiobjetivo (OM), principal foco deste trabalho. Como nesses problemas a função a ser otimizada é um vetor, necessitamos redefinir a relação de ordem em \mathbb{R}^r para caracterizar o que vem a ser uma “solução ótima” nesse contexto. Assim, as noções de *soluções eficientes* e *não dominadas* amplamente utilizadas ao longo deste trabalho, serão apresentadas e exemplificadas. Todas as definições e conceitos da OM foram baseadas nas referências apresentadas em (EHRGOTT, 2000) e (MIETTINEN, 1999).

1.1 Conceitos e definições

Apresentamos algumas definições importantes para desenvolver os métodos de otimização que são propostos e também formalizamos os conceitos de otimalidade em um POM.

Definição 1.1 (Problema geral de otimização multiobjetivo). *Um POM tem um número $r > 1$ de funções-objetivos que devem ser minimizadas, sujeito a um conjunto viável $\emptyset \neq \mathcal{X} \subseteq \mathbb{R}^n$. A forma geral de um POM é*

$$\begin{aligned} \text{Minimize } & z = (f_1(x), \dots, f_r(x)) \\ \text{sujeito a } & x \in \mathcal{X} = \{x \in \mathbb{R}^n : g_j(x) \leq 0, j = 1, \dots, l\}, \end{aligned} \tag{1.1}$$

onde:

- $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$: vetor das variáveis decisórias;
- $z_k = f_k(x)$: k -ésima função-objetivo a ser minimizada, $k = 1, \dots, r$;
- $\emptyset \neq \mathcal{X} \subseteq \mathbb{R}^n$: espaço de decisão ou factível, gerado pelas l restrições $g_j(x) \leq 0$.

Em um POM, supomos que todas as funções-objetivo são conflitantes entre si, isto é, a minimização de uma acarreta o aumento de pelo menos uma outra função.

Definição 1.2 (Espaço Critério). *O espaço $\mathcal{Z} \subseteq \mathbb{R}^r$ gerado pela aplicação das r funções-objetivos no espaço \mathcal{X} é chamado espaço critério ou objetivo. Mais formalmente, $\mathcal{Z} = \{z \in \mathbb{R}^r : z = f(x), \forall x \in \mathcal{X}\}$.*

Definição 1.3 (Função convexa e côncava). *Uma função $\phi : \mathbb{R}^n \mapsto \mathbb{R}$ é chamada convexa se para quaisquer pares de vetores x^1 e $x^2 \in \mathbb{R}^n$ e para todo $\lambda \in [0, 1]$, a seguinte condição é verdadeira:*

$$\phi(\lambda \cdot x^1 + (1 - \lambda) \cdot x^2) \leq \lambda \cdot \phi(x^1) + (1 - \lambda) \cdot \phi(x^2).$$

Uma função é côncava se $-\phi$ é convexa.

Definição 1.4 (Conjunto convexo). *Um conjunto Ω é convexo se para quaisquer pontos x^1 e x^2 em Ω , a combinação convexa linear entre eles, $\bar{x} = \lambda \cdot x^1 + (1 - \lambda) \cdot x^2$ com $\lambda \in [0, 1]$, também pertencer ao conjunto Ω .*

Lema 1.1. *O conjunto $\mathcal{X} = \{x \in \mathbb{R}^n : g_j(x) \leq 0, j = 1, \dots, l\}$ é convexo se g_j for convexa para todo $j = 1, \dots, l$.*

Demonstração. Tome x^1 e x^2 em \mathcal{X} , logo $g_j(x^1) \leq 0$ e $g_j(x^2) \leq 0, j = 1, \dots, l$. Agora, para todo $\lambda \in [0, 1]$ e usando o fato de g_j ser convexa, tem-se que:

$$\begin{aligned} 0 &\geq \lambda \cdot g_j(x^1) + (1 - \lambda) \cdot g_j(x^2) \\ &\geq g_j(\lambda \cdot x^1 + (1 - \lambda) \cdot x^2) \\ &= g_j(\bar{x}), \quad j = 1, \dots, l, \end{aligned}$$

ou seja, $\bar{x} \in \mathcal{X}$ e assim é convexo. □

Definição 1.5 (Problema de otimização multiobjetivo convexo). *Um POM é dito convexo se $z_k = f_k(x)$ é convexa para todo $k = 1, \dots, r$ e o conjunto \mathcal{X} é um conjunto convexo (isto é, se $g_j(x)$ for convexa para todo $j = 1, \dots, l$).*

Um POM em que z_k e g_j são lineares definem um POM convexo se $x \in \mathbb{R}^n$. Além disso, \mathcal{Z} é convexo. No entanto, se $x \in \mathbb{Z}^n$, o conjunto \mathcal{Z} não necessariamente será convexo. Esta é uma grande dificuldade que encontramos quando vamos resolver um POM linear e combinatório.

Uma vez que temos r funções-objetivo conflitantes entre si, não existe um ponto $x^* \in \mathcal{X}$ que as minimizam simultaneamente. Caso contrário, o POM se transformaria em escalar. Nesse contexto, podemos pensar em minimizar individualmente cada uma destas r funções e construir um vetor muito utilizado em nossos propósitos, definido a seguir.

Definição 1.6 (Vetor ideal). *O vetor $z^I = (z_1^I, \dots, z_r^I)^T \in \mathcal{Z}$ é dito Ideal quando sua k -ésima componente é o valor mínimo do seguinte problema mono-objetivo:*

$$\begin{aligned} \text{Minimize} \quad & z_k = f_k(x) \\ \text{sujeito a} \quad & x \in \mathcal{X}. \end{aligned} \tag{1.2}$$

Em geral, a imagem inversa z^I é um ponto infactível.

O seguinte vetor também será importante para o Problema de Tchebycheff, que será apresentado nas próximas seções.

Definição 1.7 (Vetor utópico). *O vetor $z^U = (z_1^U, \dots, z_r^U)^T \in \mathcal{Z}$ é chamado Utópico e sua k -ésima componente é tal que $z_k^U = z_k^I - \epsilon_k$, para todo $k = 1, \dots, r$ e $\epsilon_k > 0$.*

A comparação e avaliação de soluções factíveis em um POM é feita utilizando as relações de dominância, apresentadas a seguir.

Definição 1.8 (Dominância). *Uma solução x^1 domina uma outra solução x^2 , ambas pertencentes ao conjunto \mathcal{X} , se as condições a seguir são satisfeitas:*

1. *A solução x^1 não é pior que a solução x^2 em todos os objetivos, isto é, $f_k(x^1) \leq f_k(x^2)$, $k = 1, \dots, r$.*
2. *A solução x^1 é estritamente melhor que a solução x^2 em pelo menos uma componente $\bar{k} \in \{1, \dots, r\}$, isto é, $f_{\bar{k}}(x^1) < f_{\bar{k}}(x^2)$.*

Nesse caso, a solução x^1 domina x^2 , ou x^2 é dominada por x^1 . Notação: $x^1 \leq x^2$. O caso em que x^1 não domina x^2 denotamos por $x^1 \not\leq x^2$.

Note que duas soluções são comparáveis quando uma domina a outra. No entanto, pode acontecer que $x^1 \not\leq x^2$ nem $x^2 \not\leq x^1$. Além disso, se $x^1 \not\leq x^2$ não implica que $x^2 \leq x^1$.

Definição 1.9 (Solução eficiente). *Uma solução $x^* \in \mathcal{X}$ é chamada eficiente ou ótima no sentido de Pareto¹, se não existe uma outra solução $x \in \mathcal{X}$, de maneira que $x \leq x^*$.*

Definição 1.10 (Ponto não dominado). *Um ponto z^* no espaço de critério \mathcal{Z} é chamado de não dominado se sua imagem inversa é uma solução eficiente, isto é, $z^* = f(x^*)$.*

Definição 1.11 (Conjunto eficiente). *O conjunto eficiente denotado por \mathcal{X}^* são todos os elementos de \mathcal{X} que não são dominados por nenhum membro de \mathcal{X} , isto é, $\mathcal{X}^* = \{x^* \in \mathcal{X} : x \not\leq x^*, \forall x \in \mathcal{X}\}$.*

Definição 1.12 (Conjunto não dominado). *A imagem do conjunto eficiente \mathcal{X}^* é o conjunto $\mathcal{Z}^* \subseteq \mathcal{Z}$ e denominamos por conjunto não dominado. Formalmente, $\mathcal{Z}^* = \{z^* \in \mathbb{R}^r : z^* = f(x^*), \forall x^* \in \mathcal{X}^*\}$. Para o caso $r = 2$ objetivos, o conjunto não dominado é nomeado de fronteira de Pareto*

¹ A ser definido a seguir.

Uma solução é *não eficiente* quando é dominada por alguma solução $x \in \mathcal{X}$. A imagem no espaço de critério de uma solução não eficiente resulta num ponto dominado e no interior do conjunto \mathcal{Z} .

Num problema biobjetivo onde as duas funções são minimizadas, a fronteira de Pareto é o conjunto de pontos (fronteira) mais inferior e esquerda do espaço \mathcal{Z} . Para o caso $r = 3$ funções-objetivos, o conjunto não dominado é a superfície mais inferior de \mathcal{Z} . Para $r > 3$, já não é mais possível ter uma visualização da fronteira.

Considere x^* qualquer solução eficiente. Então:

- Existe pelo menos um elemento $x \in \mathcal{X}$ tal que $x^* \leq x$.
- Para todo $x^* \neq y^* \in \mathcal{X}^*$, $x^* \not\leq y^*$ e $y^* \not\leq x^*$.

Na Figura (4), estas propriedades são ilustradas. Note que quaisquer dois membros do conjunto não dominado podem ser comparados de acordo com as definições aqui apresentadas. Além disso, para todo $x \in \mathcal{X}^*$, existe pelo menos uma solução no espaço \mathcal{X} de tal forma que a solução eficiente a domine.

A seguir, apresentamos dois exemplos de POMs para ilustrar estas definições.

Exemplo 1.1 (Problema da compra do carro). *Considere a decisão de se comprar um veículo. Como uma hipótese simplificadora, suponhamos que a pessoa interessada esteja avaliando apenas dois critérios: o preço e o nível de conforto. Temos à disposição sete modelos, cujos preços variam de R\$ 10.000,00 a R\$ 100.000,00 e o nível de conforto varia numa escala de 0-10. Considere estes sete modelos de veículos e as informações destes presentes na Figura (1).*

A figura ilustra as alternativas para a escolha dos modelos de carros, e esses pontos estão no espaço de critério, onde podemos avaliar o grau de qualidade de cada alternativa, sempre atentando-se em otimizar os dois critérios considerados. Observando para as alternativas extremas para ambos os objetivos, vemos que a melhor escolha para o primeiro critério (preço) é o carro A (mais barato), mas ele é a pior escolha para o outro objetivo (conforto). Por outro lado, para E, as situações se invertem. Levando em conta os dois objetivos simultaneamente, não podemos afirmar que A é superior à E ou contrário. Um é melhor num objetivo e pior no outro e não importa a amplitude dessa diferença. Da mesma forma, todas as alternativas B, C e D, pertencentes à linha contínua, possuem estas características e então, têm a mesma importância para o problema. Nesse caso, A, B, C, D e E formam o conjunto de pontos não dominados no espaço de critério, de sorte que nenhum veículo associado à estes pontos pode ser considerado melhor que o outro. Sempre associada à um ponto não dominado, existe uma solução eficiente no espaço

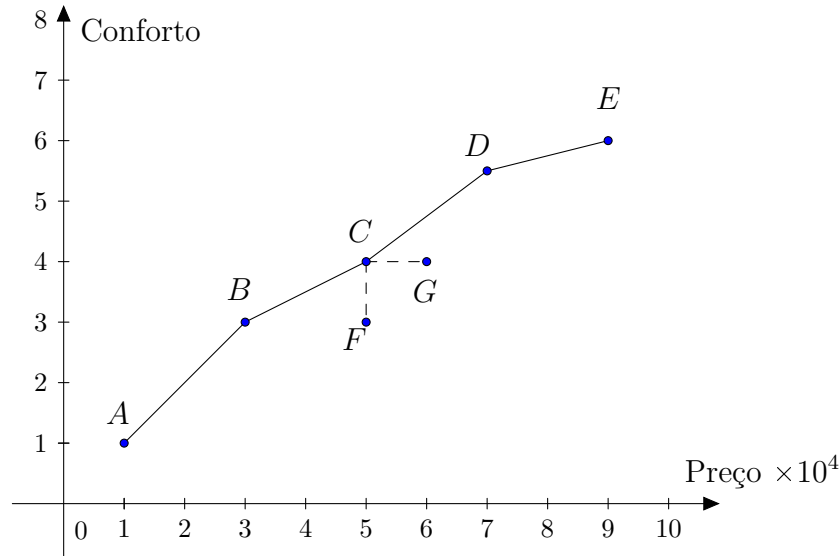


Figura 1 – Possíveis alternativas para compra de um carro - Exemplo 1.1.14

admissível de um POM. No presente exemplo, as soluções eficientes seriam associadas aos cinco veículos, cujos preços e níveis de conforto formariam estes cinco pontos não dominados contidos em \mathbb{R}^2 .

Por outro lado, os pontos F e G são pontos dominados para o problema. Por exemplo, com um preço de R\$ 50.000,00 temos duas alternativas: o carro com conforto de nível 3 (F) e o de nível 4 (C). Naturalmente, a melhor opção seria a C . Nesse caso, os pontos C e F são comparáveis. Do mesmo modo, para um mesmo nível 4 de conforto, a solução C é melhor que a G , pois custa menos.

No exemplo anterior, o conjunto de veículos à disposição é discreto, logo o conjunto de pontos não dominados também o será. No exemplo a seguir, temos um conjunto eficiente e não dominado contínuo.

Exemplo 1.2. Considere o seguinte problema biobjetivo contínuo e não linear:

$$\begin{aligned} \text{Minimize} \quad & z = (x_1, x_2) \\ \text{sujeito a} \quad & x_1^2 + x_2^2 \leq 1 \\ & (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1. \end{aligned}$$

Na Figura(2) ilustramos dois círculos cuja parte hachurada é a região admissível \mathcal{X} para este problema. Neste peculiar exemplo temos $\mathcal{X} = \mathcal{Z}$.

Nesta figura, é importante atentar para o arco circular EF em cor vermelha, descrita pela equação $(x_1 - 1)^2 + (x_2 - 1)^2 \leq 1$, com $0 \leq x_1 \leq 1$ e $0 \leq x_2 \leq 1$. Note que os pontos pertencentes a este arco têm uma propriedade comum: conforme uma componente destes pontos aumenta, a outra obrigatoriamente diminui, e vice-versa, de sorte que não é mais possível minimizar nenhuma delas sem que a outra seja aumentada. A mesma

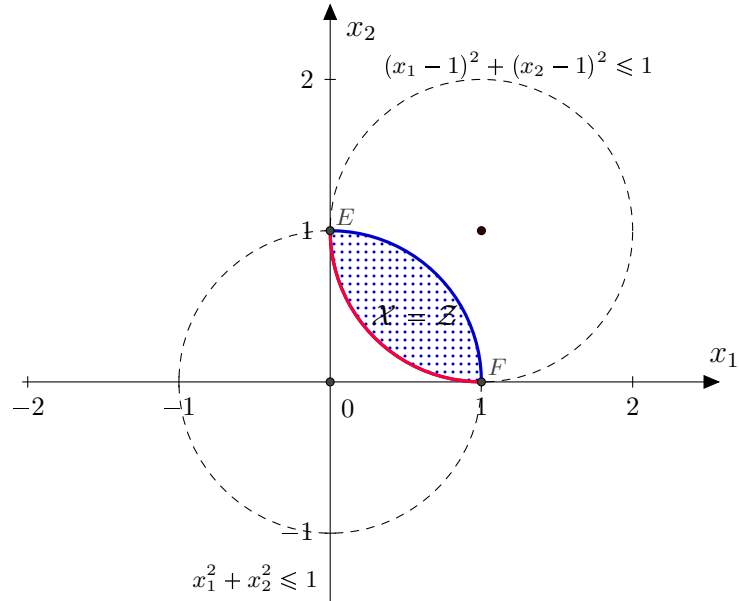


Figura 2 – Região admissível para o Exemplo 1.1.15

observação não é válida para os pontos no interior do conjunto hachurado. Assim, todas as soluções deste POM pertencem à curva (de cardinalidade infinita) em vermelho mais inferior e à esquerda.

Outras definições mais específicas também são apresentadas a seguir.

Definição 1.13 (Solução fracamente eficiente). *Uma solução \hat{x} é fracamente eficiente se não existe $x \in \mathcal{X}$ de modo que $f_k(x) < f_k(\hat{x})$ para todo $k = 1, \dots, r$.*

Definição 1.14 (Ponto fracamente não dominado). *Um ponto \hat{z} no espaço de critério é fracamente não dominado se sua imagem inversa é uma solução fracamente eficiente, isto é, $\hat{z} = f(\hat{x})$.*

De forma similar definimos os conjuntos *fracamente eficiente* e *fracamente não dominado*.

Pelas definições, as soluções eficientes também são fracamente eficientes, mas a recíproca é falsa. Na Figura (3) ilustramos dois pontos fracamente não dominados, A e B, ambos sobre os segmentos em azul. No entanto, eles são dominados. Dessa forma, o conjunto fracamente não dominado é a curva em vermelha unida com os segmentos em azul, ao passo que o conjunto não dominado restringe apenas à curva em vermelho.

Definição 1.15 (Solução localmente eficiente). *Uma solução $x^* \in \mathcal{X}$ é localmente eficiente se existe um $\epsilon > 0$ tal que x^* é eficiente em $\mathcal{X} \cap B(x^*, \epsilon)$, onde $B(x^*, \epsilon)$ é a bola centrada em x^* de raio ϵ .*

Definição 1.16 (Solução globalmente eficiente). *Uma solução $x^* \in \mathcal{X}$ é globalmente eficiente quando ela é eficiente em \mathcal{X} .*

Uma fronteira de Pareto localmente (globalmente) eficiente é a imagem de todos as soluções localmente (globalmente) eficientes.

A seguir, é fornecida uma importante definição com respeito à natureza de soluções eficientes.

Definição 1.17 (Soluções e pontos suportados). *Uma solução $x^* \in \mathcal{X}^*$ é eficiente e suportada, se existe algum $w = (w_1, \dots, w_r)^T > 0$ de maneira que x^* seja solução do problema escalar:*

$$\begin{aligned} \text{Minimize} \quad & z = w^T f(x) \\ \text{sujeito a} \quad & x \in \mathcal{X}. \end{aligned}$$

Além disso, o vetor objetivo resultante $z^* = f(x^*)$ é um ponto suportado. Caso contrário, x^* e z^* são denominados de solução eficiente não suportada e ponto não suportado, respectivamente.

Na Figura (3), C é um ponto suportado, ao passo que o ponto D é não suportado. Associado à estes pontos, existem no espaço \mathcal{X}^* as soluções eficientes suportadas e não suportadas, respectivamente. Note que C está na parte convexa da fronteira de Pareto, enquanto D na parte não convexa. Estes conceitos serão melhor justificados e trabalhados no Capítulo 3, quando o método da Soma Ponderada for apresentado.

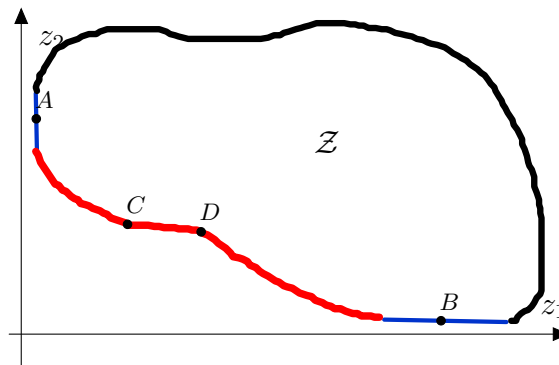


Figura 3 – Soluções suportadas, não suportadas e fracamente não dominadas

Definição 1.18 (Vetor nadir). *O vetor $z^N \in \mathcal{Z}$ é chamado de nadir e sua k -ésima componente é o maior valor da função-objetivo k no conjunto eficiente.*

Em geral, z^N é computacionalmente difícil de ser obtido. No entanto para $r = 2$ objetivos, ele é facilmente calculado, valendo-se da seguinte propriedade, provada em (BEENSON, 1971):

Teorema 1.1. *Dado um problema de otimização biobjetivo, no conjunto de soluções eficientes, o valor mínimo de uma função-objetivo é o valor máximo da outra e vice-versa.*

Nesse caso, sejam os pontos do espaço de critério $z^{*1} = (f_1(x^{*1}), f_2(x^{*1}))^T = (z_1^-, z_2^+)^T$ e $z^{*2} = (f_1(x^{*2}), f_2(x^{*2}))^T = (z_1^+, z_2^-)^T$ onde x^{*k} é o ótimo do objetivo $k = 1, 2$. Estes pontos são denominados de *end-points* ou *lexicográficos* da fronteira de Pareto. Logo, $z^I = (z_1^-, z_2^-)^T$ e $z^N = (z_1^+, z_2^+)^T$, à luz do teorema anterior.

Observação 1.1. O vetor z^N não deve ser confundido com o vetor cuja componente k é o maior valor para o k -ésimo objetivo no espaço \mathcal{Z} , como mostra a Figura (4) e as outras definições.

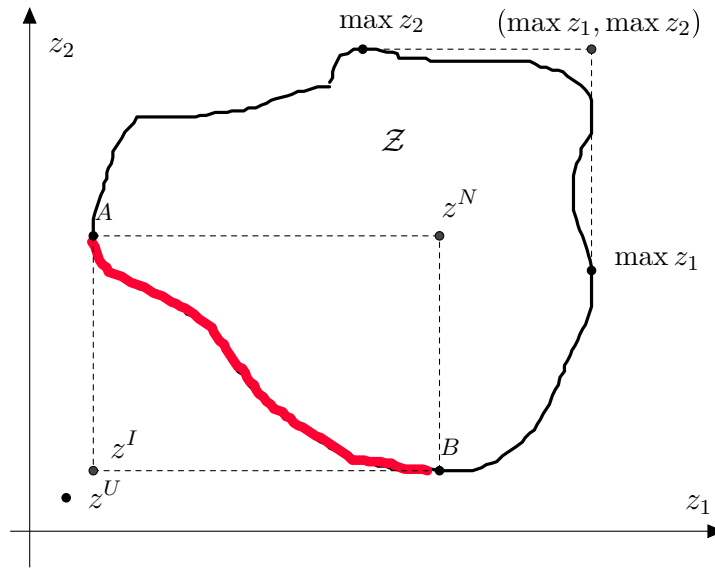


Figura 4 – Pontos notáveis no espaço \mathcal{Z}

Em alguns métodos de escalarização de OM, é muito conveniente que as funções-objetivos sejam normalizadas, pois, assim sendo, a magnitude das funções não afetam no processo de obtenção de soluções eficientes. Nesse caso, esta normalização é feita usando-se os pontos lexicográficos:

$$z_k^{\text{norm}} = \frac{z_k - z_k^I}{z_k^N - z_k^I}, \quad k = 1, \dots, r, \quad (1.3)$$

onde z_k^{norm} é k -ésima função-objetivo normalizada, isto é, $z_k^{\text{norm}} \in [0, 1]$.

Resolver um POM é encontrar *todas* as soluções eficientes, onde possuem igual importância. No caso de problemas com variáveis reais, todas as funções envolvidas contínuas e $z^I \neq z^N$, $|\mathcal{X}^*|$ é infinito e no caso de problemas com variáveis discretas, $|\mathcal{X}^*| < \infty$. Os métodos para resolver os POMs devem ser desenvolvidos de modo a produzir um conjunto eficiente que atentam à duas metas distintas e conflitantes: (i) produzir soluções globalmente eficientes e (ii) distribuir e diversificar estas soluções ao longo da fronteira, como pode ser visto na Figura (5). Um bom método de programação multiobjetivo deve atender a estes dois requisitos.

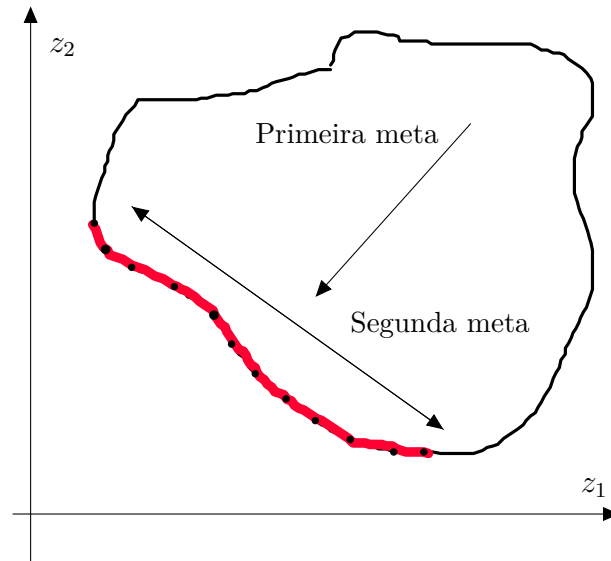


Figura 5 – Metas em OM

A geração de diferentes soluções eficientes pode ser feita por estratégias de *escalarizações*, isto é, consideramos uma sequência de subproblemas escalares, que ao serem otimizados e sob certas condições, produzem as soluções eficientes. Assim, o enfoque do nosso trabalho é fornecer, adaptar e expandir técnicas clássicas (exatas) de OM em problemas discretos. O uso desses métodos motivado pelo advento de computadores e pacotes computacionais de ótima qualidade, tornando plausível esta expansão. Além disso, outro foco está em obter o conjunto eficiente, contendo todas as soluções do problema, gerando o menor número possível de subproblemas escalares. Essa ideia pode tornar viável o tempo de resolução de um POM.

A literatura específica na área de OM relata os métodos de Soma Ponderada e ε -Restrito como as abordagens mais utilizadas; no entanto, além da aplicação destas técnicas de escalarizações, pretendemos utilizar, testar e comparar outras abordagens, como a Métrica Tchebycheff, um método híbrido de Soma Ponderada Modificada com ε -Restrito, método de Benson e um método desenvolvido em (SYLVA; CREMA, 2004). Além destes seis métodos, propomos um novo procedimento de escalarização, advindo e aproveitando as ideias da Métrica de Tchebycheff, capaz de obter todas as soluções eficientes de um problema de otimização biobjetivo discreto. A pergunta que pode surgir é: “*Esta nova técnica é competitiva com as demais abordagens presentes na literatura, em termos de custo computacional e potencial para detectar todas as soluções do problema?*”. A resposta a esta e outras questões são dadas e convalidadas neste trabalho, atestando uma ideia inovadora e que mostrou uma contribuição na área de problemas de corte e em OM.

Capítulo 2

Modelagem matemática do problema de corte

O Problema de Corte é um dos problemas combinatórios mais estudados dentro do campo da otimização combinatória, devido, principalmente, a sua aplicabilidade no mundo da engenharia de produção, fazendo parte do planejamento de uma diversidade de indústrias que cortam papel, madeira, vidro, plásticos, tecido, entre outras matérias-primas. Apesar de ser muito simples de ser entendido, este problema tem um elevado nível de complexidade, sendo classificado na literatura como \mathcal{NP} -Completo¹ ((COFFMAN JR.; GAREY; JOHNSON, 1997)).

O Problema de Corte Unidimensional Inteiro (PCUI), onde apenas uma dimensão no processo de corte é relevante, pode ser entendido da seguinte forma: as matérias-primas a serem utilizadas são estocadas em forma de grandes rolos ou bobinas-mestre de larguras padronizadas, e apenas mais tarde são reduzidas a tamanhos menores e variados (itens) para serem usadas pela indústria de acordo com as demandas externas. No entanto, esse procedimento causa o inconveniente de preparar as máquinas para a operação de corte destes materiais, ocasionando, inevitavelmente, perdas, ajustes de máquinas e um aumento no tempo de produção com sucessivas paradas no maquinário. Desse modo, há necessidade de planejar a maneira de efetuar estes cortes, a fim de minimizar o desperdício e efeitos negativos ocasionados com esta operação.

A tarefa de se obter uma solução ótima exata para o PCUI é desafiante. Isso se deve, principalmente, à integralidade e o elevado número de variáveis decisórias envolvidas neste problema. Alguns poucos métodos exatos para encontrar a solução ótima inteira do PCUI são conhecidos na literatura, como apresentados em (BELOV; SCHEITHAUER, 2002), (CARVALHO, 1999), (VANCE, 1998) e (VANDERBECK, 1996). No entanto, estes métodos podem ser aplicados apenas a problemas de pequeno e médio porte.

¹ Um problema é \mathcal{NP} -Completo quando não temos um algoritmo que o resolva em um tempo polinomial em função de seus dados.

Sendo assim, metodologias aproximativas têm sido desenvolvidas nas últimas cinco décadas para este problema. O primeiro trabalho a tratar o PCUI de maneira não exata foi proposto em (GILMORE; GOMORY, 1961), que propuseram uma técnica de Geração de Colunas (GC) para obtenção de uma solução ótima contínua. Outros métodos inexatos têm sido desenvolvidos e uma breve revisão é dada em (WASCHER; GAU, 1996). Ressaltamos também o uso de Heurísticas de Arredondamento, proposto em (POLDI; ARENALES, 2006), e algumas metaheurísticas específicas, desenvolvidas em (GEHRING; BORTFELDT, 1998), (GOLFETO; MORETTI; SALLES, 2008), (GOLFETO; MORETTI; SALLES, 2009), entre tantos.

Devido ao grande número de publicações de trabalhos com o Problemas de Cortes e Empacotamento e suas variantes nas últimas décadas, o trabalho apresentado em (DYCKHOFF, 1990) de 1990 mostrou uma classificação, organização e categorização desses problemas. Em 2006, essa tipologia foi modificada com artigo dado em (WASCHER; HAUBNER; SCHUMANN, 2007), amplamente utilizada na área. Resumidamente, essa classificação leva em conta os seguintes critérios:

- **Dimensão:** uni,bi,tri-dimensional, de acordo com a quantidade de dimensões levadas em conta.
- **Tipos de atribuição:** há dois casos distintos:
 - Muitos itens diferentes sendo retirados de objetos maiores, que não são suficientes para acomodar todos os itens. A intenção é minimizar o corte dos objetos maiores (L).
 - Muitos itens diferentes sendo retirados de objetos maiores, que são suficientes para acomodar todos os itens. A intenção é maximizar o uso de cada objeto maior (B).
- **Classificação dos itens:** temos três casos:
 - **tipo C:** todos os itens têm a mesma forma e tamanho,
 - **tipo R:** itens podem ser classificados em poucas classes de tamanhos variados,
 - **tipo F:** itens com muita variabilidade no tamanho e forma.
- **Objetos:** há duas maneiras de classificar os objetos maiores:
 - Um único objeto em estoque,
 - Vários objetos distintos em estoque.
- **Geometria dos itens:** aplicada apenas nos casos bi- e tri-dimensionais, distinguimos os itens em regulares (R) (retângulos, triângulos, círculos, bolas, cilindros, etc.) e irregulares (I).

De forma mais concisa, um Problema de Corte pode ser classificado segundo a quádrupla $\alpha/\beta/\gamma/\delta/\lambda$, onde α denota a dimensão (1, 2 ou 3), β é o tipo de atribuição (L ou B), γ denota a classificação dos itens (C, R ou F), δ é a quantidade de objetos disponíveis (b) e finalmente λ é a geometria dos itens (R ou I). Assim, o problema que estamos propostos a resolver é classificado como $1/B/\{C,R,F\}/\{b\}$.

A seguir, apresentamos a modelagem matemática utilizada para o problema de corte. Em um primeiro momento, consideramos a versão mono-objetiva para o problema. Os casos sem e com os padrões de corte dados a priori são tratados separadamente. Para o caso sem os padrões de corte a priori, a modelagem recai num Problema de Programação Não Linear e são desenvolvidas algumas estratégias de linearizações via decomposições por variáveis binárias. Na segunda parte, justificamos o uso de uma abordagem multiobjetiva e apresentamos as modelagens com e sem os padrões de corte a priori. Finalmente, um modelo parcialmente relaxado para o PCUIM é descrito e será extensamente utilizado ao longo desse trabalho.

2.1 Abordagem mono-objetiva do problema de corte

Considere b bobinas-mestre em estoque de larguras L_k (larguras padrão) com $k \in K = \{1, \dots, b\}$ e m o número de itens demandados. Cada item i tem largura $\ell_i < \min_{1 \leq k \leq b} \{L_k\}$ e ao menos d_i unidades precisam ser produzidas a fim de atender a demanda, $i \in I = \{1, \dots, m\}$. O objetivo consiste em minimizar a soma do número de padrões de corte a fim de atender à esta demanda mínima exigida. Do ponto de vista operacional, apenas faz sentido cortar um número inteiro de vezes destas bobinas-mestre, o que torna este problema combinatório.

Definição 2.1. *Uma instrução que lista os itens demandados a serem cortados da bobina-mestre é chamada de padrão de corte, que pode ser associado ao vetor m -dimensional*

$$a_{jk} = (a_{1jk}, a_{2jk}, \dots, a_{mjk})^T,$$

onde cada entrada $a_{ijk} \in \mathbb{N}$, denota a quantidade do item i presente no padrão $j \in J_k = \{1, \dots, n_k\}$ da bobina-mestre k . O número n_k denota o número total de combinações de corte que podem ser feitas usando a bobina-mestre k de largura L_k .

Definição 2.2. *Um padrão de corte $a_{jk} \in \mathbb{N}^m$, com $j \in J_k = \{1, \dots, n_k\}$, é admissível se satisfaz as seguintes condições:*

$$\begin{aligned} \sum_{i=1}^m \ell_i \cdot a_{ijk} &\leq L_k, \\ \sum_{i=1}^m \ell_i \cdot a_{ijk} &\geq L_k - \eta, \\ \sum_{i=1}^m a_{ijk} &\leq q, \end{aligned} \tag{2.1}$$

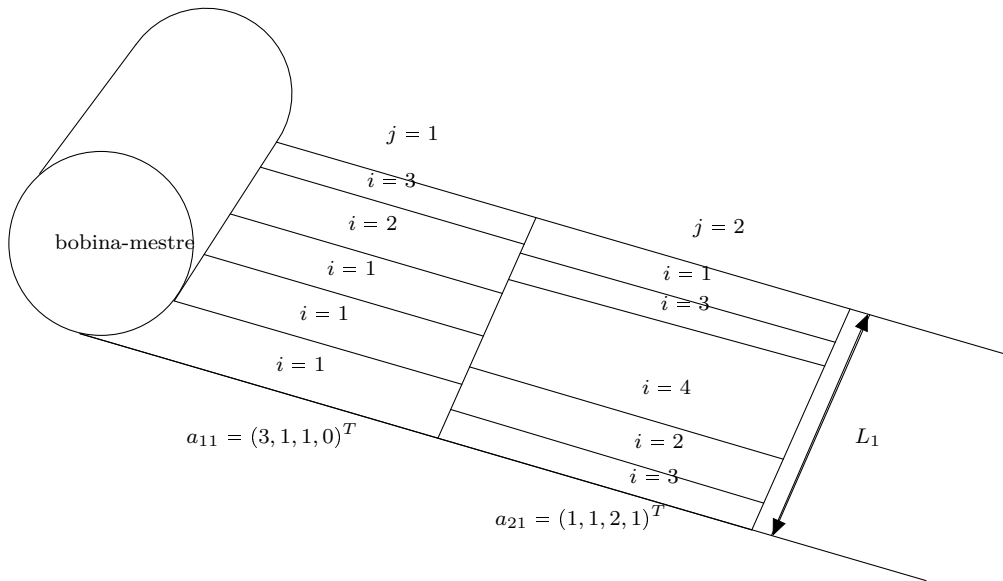


Figura 6 – Exemplo de padrões de corte

onde $\eta = \min_{1 \leq i \leq m} \{\ell_i\}$ e q é o número de facas máximo permitido no maquinário.

Definição 2.3. Um padrão de corte $a_{jk} \in \mathbb{N}^m$ é dito homogêneo (com respeito a bobina-mestre k) se ele contém apenas o item \bar{i} na sua confecção, isto é, $a_{\bar{i}jk} = \min \left\{ \left\lfloor \frac{L_k}{\ell_{\bar{i}}} \right\rfloor, q \right\}$ e $a_{ijk} = 0$ para $i \neq \bar{i}$, $i \in I$.

Na Figura (2.1), ilustramos uma bobina-mestre de largura L_1 , quatro itens demandados e dois padrões de cortes. No padrão $j = 1$, temos três vezes o item 1 sendo cortado, uma vez os itens 2 e 3 e nenhuma vez o item quatro, logo ele pode ser representado pelo vetor $a_{11} = (3, 1, 1, 0)^T$. Um outro padrão de corte poderia ser dado pelo vetor $a_{21} = (1, 1, 2, 1)^T$, em que os itens 1, 2 e 4 são cortados uma só vez e o item 3 sendo cortado 2 vezes. Note ainda, que se o número de facas fosse menor que quatro, ambos seriam infactíveis.

Quando há bobinas-mestre de várias larguras, é conveniente limitar a quantidade que cada uma é utilizada (um estoque máximo e_k para a bobina k). Geralmente as indústrias possuem uma quantidade limitada de espaço físico para estocagem e a compra das bobinas é efetuada conforme há solicitações de encomendas a serem cortadas. Porém esta restrição não é levada em consideração nos modelos de problema de corte neste trabalho, pois ela pode impossibilitar a geração de múltiplas soluções eficientes para o problema.

Nesta formulação inicial, partimos do princípio que *todas* as $\mu = \sum_{k=1}^b n_k$ combinações de corte da bobina k são conhecidas. As variáveis decisórias x_{jk} indicam a frequência

do padrão de corte j na bobina-mestre k . O modelo para este caso é:

$$\begin{aligned} \text{Minimize} \quad & z = \sum_{k=1}^b \sum_{j=1}^{n_k} x_{jk} \\ \text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^{n_k} a_{ijk} \cdot x_{jk} \geq d_i, \quad i \in I, \\ & x_{jk} \in \mathbb{N}, \quad j \in J_k, \quad k \in K. \end{aligned} \tag{2.2}$$

A função-objetivo para esta formulação visa minimizar a soma das frequências dos padrões de cortes a serem usados. O conjunto de restrições está associado ao atendimento mínimo da demanda d_i para cada item. É permitido excesso de produção nesse caso. O segundo grupo de restrições está associado com a integralidade de todas as variáveis deste problema, isto é, não faz nenhum sentido cortar um número não inteiro de vezes um padrão de corte, muito embora faremos uso de sua relaxação para obtenção de soluções inteiras com o uso de uma heurística propriamente desenvolvida.

O PCUI pode ter outros objetivos, como minimizar custos com a produção, maximização de lucros, minimização de perdas intrínsecas com os processos naturais de corte. A proposição a seguir mostra, para o caso de $k = 1$, que ao minimizar a função-objetivo anterior com as restrições de demanda na igualdade, equivale minimizar a soma das perdas $c_j = L - \sum_{i=1}^m \ell_i \cdot a_{ij}$ associada à cada padrão de corte j .

Proposição 2.1. *Considerando-se os objetivos*

$$\bar{z} = \min \left\{ \sum_{j=1}^n c_j \cdot x_j \right\} \quad e \quad z = \min \left\{ \sum_{j=1}^n x_j \right\},$$

então na solução ótima, $\bar{z}^* = z^*$, isto é, a mesma solução que minimiza \bar{z} também minimiza z .

Demonstração. Usando a definição do desperdício c_j e as restrições de demanda $\sum_{j=1}^n a_{ij} \cdot x_j = d_i$, na igualdade, temos que

$$\begin{aligned} \bar{z}^* &= \min \left\{ \sum_{j=1}^n c_j \cdot x_j \right\} = \min \left\{ \sum_{j=1}^n \left(L - \sum_{i=1}^m \ell_i \cdot a_{ij} \right) \cdot x_j \right\} \\ &= \min \left\{ L \cdot \sum_{j=1}^n x_j - \sum_{j=1}^n \sum_{i=1}^m \ell_i \cdot a_{ij} \cdot x_j \right\} \\ &= \min \left\{ L \cdot \sum_{j=1}^n x_j - \sum_{i=1}^m \ell_i \left(\sum_{j=1}^n a_{ij} \cdot x_j \right) \right\} \\ &= \min \left\{ L \cdot \sum_{j=1}^n x_j - \sum_{i=1}^m \ell_i \cdot d_i \right\} \\ &= \min \left\{ \sum_{j=1}^n x_j \right\} = z^*. \end{aligned}$$

□

Embora permitimos excesso de produção, o resultado anterior mostra que as funções-objetivo \bar{z} e z são não conflitantes, isto é, a minimização de uma acarreta na minimização da outra. Em todo este trabalho, utilizamos $z = z_1$ como um dos objetivos a ser minimizado.

O Problema (2.2) pode ser escrito de forma compacta como:

$$\begin{aligned} \text{Minimize} \quad & z = \|x\|_1 \\ \text{sujeito a} \quad & Ax \geq d, \\ & x \in \mathbb{N}^\mu, \end{aligned} \tag{2.3}$$

e será útil considerar esta forma matricial para futuros desenvolvimentos. Nesta notação, cada coluna j da matriz $A \in \mathbb{N}^{m \times \mu}$ contém o j -ésimo padrão de corte da bobina-mestre k .

A parametrização ou geração de todos os vetores possíveis a_{jk} torna-se uma técnica difícil e tediosa de ser aplicada, pois o número de padrões de corte pode ser extremamente grande mesmo para m da ordem de dezenas, desde que a razão $\min\{L_k/\ell_i\}$ fique pequena. A Tabela (1) ilustra alguns exemplos de como esse número de combinações de corte aumenta à medida que os itens ficam com as larguras menores. Considere $m = 4$, $b = 1$, $L_1 = 100$, $q = 100$ e os itens com larguras $\ell = c \cdot (1, 2, 3, 4)^T$.

Tabela 1 – Número de combinações de corte e padrões usados na solução ótima conforme ℓ varia

c	μ	$\# x_j^* > 0$
10	41	4
5	202	4
2	2.263	4
1	15.840	6
10^{-1}	75.755	6

Pode ser demonstrado ((GILMORE; GOMORY, 1961)) que

$$n_k > \frac{m!}{(m - \bar{q}_k)! \bar{q}_k!},$$

onde \bar{q}_k é o número médio de itens por padrão.

Felizmente, para resolver a relaxação linear de (2.3), é desnecessária a geração de todas as colunas da matriz A , uma vez que na solução ótima de um Problema de Programação Linear, apenas m variáveis serão não nulas (se não houver degenerescência). As $\mu - m$ (não básicas) serão nulas. No Problema de Programação Inteiro (PLI) original, o número de padrões efetivamente usados é ligeiramente diferente, pois a solução ótima pode não ser uma solução básica (no senso do Simplex), isto é, mais que m padrões podem ser utilizados, como comprova a terceira coluna da Tabela (1).

Um procedimento clássico que não enumera todas as colunas para resolver a relaxação linear, foi proposto no bem conhecido trabalho apresentado em (GILMORE; GOMORY, 1961), onde um método gera as m colunas ótimas para a relaxação linear do problema (2.3). Este método é conhecido como *Geração de Colunas de Gilmore-Gomory*. No entanto, estas m colunas podem não coincidir com as colunas presentes na solução ótima do problema original inteiro. Além disso, devemos adotar algum procedimento heurístico para obter as frequências $x_{jk} \in \mathbb{N}$ do problema original, tornando este método heurístico, isto é, sem garantia de otimalidade da solução.

Um caminho para obter uma aproximação para a solução ótima inteira para o PCUI seria considerar apenas um subconjunto $P = \{1, \dots, p\} \subset \cup_{k=1}^b \{J_k\}$ de todas as combinações de corte possíveis, com $p < \mu$. Nesse caso, temos uma aproximação para o modelo linear inteiro (2.2) dada por:

$$\begin{aligned} \text{Minimize} \quad & z = \sum_{k=1}^b \sum_{j=1}^p x_{jk} \\ \text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \geq d_i, \quad i \in I, \\ & x_{jk} \in \mathbb{N}, \quad j \in P, \quad k \in K. \end{aligned} \quad (2.4)$$

É útil incorporar ao PCUI o *setup* de máquina. Ele está associado a cada troca de um padrão de corte para outro. Do ponto de vista operacional, o *setup* envolve uma série de reparações e parada nas máquinas para que as facas possam ser trocadas para efetuar outros cortes e constitui um importante aspecto a ser levado em conta neste problema.

A variável binária:

$$y_{jk} = \begin{cases} 1, & \text{se } x_{jk} > 0 \\ 0, & \text{caso contrário,} \end{cases} \quad (2.5)$$

indica se o padrão $j \in J_k$ foi utilizado na bobina-mestre $k \in K$.

Considerando o modelo compacto (2.3) relaxado, temos que o problema resultante é um Problema de Programação Linear com apenas m restrições de demanda. Como comentamos anteriormente, o vetor com a solução ótima terá no máximo m componentes não nulas e as $\mu - m$ remanescentes serão nulas. Isto significa que o *setup* de máquina do problema relaxado será um número ε tal que $1 \leq \varepsilon \leq m$. Contudo, para o problema inteiro original, o *setup* pode ser ligeiramente diferente de m . Os exemplares considerados neste trabalho nunca geraram um *setup* maior que $2 \cdot m$.

Numa primeira abordagem, supomos que todas as combinações de corte $a_{jk} = (a_{1jk}, a_{2jk}, \dots, a_{mjk})^T$ para todo $j \in J_k$ e $k \in K$ são conhecidas. Nesse caso, o modelo linear

que minimiza a soma da frequência dos padrões de corte utilizados com um *setup* menor ou igual à ε e atendendo a demanda mínima, é apresentado a seguir:

$$\begin{aligned}
\text{Minimize} \quad & z = \sum_{k=1}^b \sum_{j=1}^{n_k} x_{jk} \\
\text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^{n_k} a_{ijk} \cdot x_{jk} \geq d_i, \quad i \in I, \\
& x_{jk} \leq N \cdot y_{jk}, \quad j \in J_k, \quad k \in K, \\
& x_{jk} \geq y_{jk}, \quad j \in J_k, \quad k \in K, \\
& \sum_{k=1}^b \sum_{j=1}^{n_k} y_{jk} \leq \varepsilon, \\
& y_{jk} \in \mathbb{B}, \quad x_{jk} \in \mathbb{N}, \quad j \in J_k, \quad k \in K,
\end{aligned} \tag{2.6}$$

onde N é um limitante superior para x_{jk} . Note que esta formulação tem $2 \cdot \mu$ variáveis inteiras, o dobro com relação problema sem o *setup*. Isso reforça ainda mais o argumento de que a parametrização de todos os vetores a_{jk} torna-se uma técnica inviável para resolver o PCUI.

Diante dessa situação, uma primeira estratégia para resolver o PCUI consiste em considerar uma aproximação para a formulação anterior, ao fornecer $p < \mu$ combinações de corte a priori. O modelo aproximado e linear é dado por:

$$\begin{aligned}
\text{Minimize} \quad & z = \sum_{k=1}^b \sum_{j=1}^p x_{jk} \\
\text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \geq d_i, \quad i \in I, \\
& x_{jk} \leq N \cdot y_{jk}, \quad j \in P, \quad k \in K, \\
& x_{jk} \geq y_{jk}, \quad j \in P, \quad k \in K, \\
& \sum_{k=1}^b \sum_{j=1}^p y_{jk} \leq \varepsilon, \\
& y_{jk} \in \mathbb{B}, \quad x_{jk} \in \mathbb{N}, \quad j \in P, \quad k \in K,
\end{aligned} \tag{2.7}$$

e envolve $2 \cdot p$ variáveis inteiras. É muito comum utilizar p um múltiplo de m colunas.

Uma segunda ideia mais sofisticada, consiste em considerar a_{ijk} como variáveis, ou seja, os padrões de corte são gerados pela própria formulação, assim como as respectivas frequências. Nesse caso, devemos inserir as condições (2.1) no modelo para atender a definição de um padrão de corte factível. O resultado é a formulação apresentada em (2.8) que por sua vez, tem a desvantagem de ser não linear, devido ao aparecimento de uma bilinearidade nas restrições de demanda. Além disso este conjunto de restrições é não

convexo, tornando o problema mais complicado de ser resolvido. Isto justifica o fato de se encontrar poucos trabalhos na literatura que abordam tal formulação.

$$\begin{aligned}
\text{Minimize} \quad & z = \sum_{k=1}^b \sum_{j=1}^p x_{jk} \\
\text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \geq d_i, \quad i \in I, \\
& \sum_{i=1}^m \ell_i \cdot a_{ijk} \leq L_k, \quad j \in P, \quad k \in K, \\
& \sum_{i=1}^m \ell_i \cdot a_{ijk} \geq L_k - \eta, \quad j \in P, \quad k \in K, \\
& \sum_{i=1}^m a_{ijk} \leq q, \quad j \in P, \quad k \in K, \\
& \sum_{i=1}^m a_{ijk} \geq q - \eta, \quad j \in P, \quad k \in K, \\
& x_{jk} \leq N \cdot y_{jk}, \quad j \in P, \quad k \in K, \\
& x_{jk} \geq y_{jk}, \quad j \in P, \quad k \in K, \\
& \sum_{k=1}^b \sum_{j=1}^p y_{jk} \leq \varepsilon, \\
& y_{jk} \in \mathbb{B}, \quad x_{jk} \text{ e } a_{ijk} \in \mathbb{N}, \quad j \in P, \quad k \in K.
\end{aligned} \tag{2.8}$$

O trabalho proposto em (HARJUNKOSKI et al., 1998) é o único a trabalhar com o formato do modelo acima. Essa não convexidade das restrições de demanda é tratada e uma série de mudanças de variáveis é apresentada, com a intenção de deixar o modelo convexo e posteriormente utilizar *solvers* de programação não linear inteira. No entanto, o modelo cresce demasiadamente em termos do número de variáveis e restrições, tornando sua solução praticamente impossível para os problemas prático.

É importante frisar que fornecemos ao modelo até p padrões de corte para cada bobina-mestre. O valor de p é uma escolha arbitrária e difícil de ser determinada. Usualmente, $p = 2 \cdot m$, pois nas classes de problemas de corte que consideramos, não foram encontrados exemplares onde na solução ótima utilizassem mais do que $2 \cdot m$ padrões por bobina-mestre.

Na próxima seção, nós apresentamos duas técnicas para linearizar o Modelo (2.8) e tornar possível a resolução do PCUI de maneira exata utilizando pacotes computacionais de PLI.

2.2 Técnicas de linearização

As ideias apresentadas nesta seção foram utilizadas em (HARJUNKOSKI et al., 1996) para fazer a linearização do Modelo (2.8). Duas formas similares são utilizadas, descritas detalhadamente a seguir.

2.2.1 Linearização usando as variáveis x_{jk}

Numa primeira abordagem, utilizamos as variáveis binárias auxiliares α_{jkh} para representar x_{jk} . Assuma que x_{jk} possua algum limitante superior N , isto é, $x_{jk} \leq N$ com $j \in P$ e $k \in K$.

Este limitante superior pode ser calculado utilizando os dados da demanda. Considerando a forma matricial do Problema (2.3), assumimos que uma linha i da matriz A seja nula, com exceção na entrada da coluna j . Além disso, suponhamos que nesta entrada $a_{ijk} > 0$. Neste caso, o menor valor de x_{jk} para atender a demanda d_i é $x_{jk} = d_i$. Então é razoável de afirmar que $x_{jk} \leq \max_{1 \leq i \leq m} \{d_i\}$ e assim, um limitante superior para esta variável seria dado por:

$$N = \max_{1 \leq i \leq m} \{d_i\}. \quad (2.9)$$

Escrevendo

$$x_{jk} = \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh}, \quad j \in P, \quad k \in K, \quad (2.10)$$

onde $\kappa = \lceil \log_2 N \rceil + 1$. Além disso, a restrição bilinear da Formulação (2.8) pode ser reescrita por:

$$\begin{aligned} d_i &\leq \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \\ &= \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} \\ &= \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot a_{ijk} \cdot \alpha_{jkh} \\ &= \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot r_{ijkh}, \quad i \in I, \end{aligned} \quad (2.11)$$

onde $\mathbb{R}_+ \ni r_{ijkh} = a_{ijk} \cdot \alpha_{jkh}$ para todo $i \in I$, $j \in P$, $k \in K$, $h \in H$.

Agora, no produto bilinear $a_{ijk} \cdot \alpha_{jkh}$, temos uma variável binária multiplicada por uma variável inteira que pode ser linearizada ao impor três restrições lineares a seguir, escritas para todo $i \in I$, $j \in P$, e $h \in H = \{1, \dots, \kappa\}$:

$$\begin{aligned} r_{ijkh} &\leq M_i \cdot \alpha_{jkh}, \\ r_{ijkh} &\leq a_{ijk}, \\ a_{ijk} - M_i \cdot (1 - \alpha_{jkh}) &\leq r_{ijkh}, \end{aligned} \quad (2.12)$$

onde M_i , $i \in I$, é um limitante superior para variável a_{ijk} , que pode ser calculado ao considerarmos um padrão de corte homogêneo, onde a_{ijk} assume o maior valor. Nesse caso,

$$M_i = \max_{1 \leq k \leq b} \left\{ \min \left\{ \left\lfloor \frac{L_k}{\ell_i} \right\rfloor, q \right\} \right\}, \quad i \in I. \quad (2.13)$$

Considerando o conjunto de restrições (2.12) que linearizam a forma bilinear, notamos que se $\alpha_{jkh} = 0$, então $r_{ijkh} = 0$. Assim, para este conjunto de restrições, temos o seguinte:

$$\begin{aligned} r_{ijkh} &\leq 0, \\ r_{ijkh} &\leq a_{ijk}, \\ a_{ijk} - M_i &\leq r_{ijkh}, \end{aligned}$$

isto é, $r_{ijkh} = 0$ pela primeira restrição. A segunda restrição é redundante e a terceira se transforma em $a_{ijk} \leq M_i$, que é sempre satisfeita. Por outro lado, se $\alpha_{jkh} = 1$, então pela definição de $r_{ijkh} = a_{ijk}$. As Restrições (2.12) resultarão em

$$\begin{aligned} r_{ijkh} &\leq M_i, \\ r_{ijkh} &\leq a_{ijk}, \\ a_{ijk} &\leq r_{ijkh}, \end{aligned}$$

isto é, $r_{ijkh} = a_{ijk}$ pela segunda e terceira restrições. A primeira será trivialmente satisfeita.

O modelo para o PCUI (2.8) não linear é reescrito substituindo a restrição bilinear pelo conjunto de restrições (2.10), (2.11) e (2.12). Obtemos assim, o seguinte modelo linear equivalente expresso como:

$$\begin{aligned} \text{Minimize } z &= \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} \\ \text{sujeito a } &\sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot r_{ijkh} \geq d_i, \quad i \in I, \\ &r_{ijkh} \leq M_i \cdot \alpha_{jkh}, \quad i \in I, \quad j \in P, \quad k \in K, \quad h \in H, \\ &r_{ijkh} \leq a_{ijk}, \quad i \in I, \quad j \in P, \quad k \in K, \quad h \in H, \\ &a_{ijk} - M_i \cdot (1 - \alpha_{jkh}) \leq r_{ijkh}, \quad i \in I, \quad j \in P, \quad k \in K, \quad h \in H, \\ &\sum_{i=1}^m \ell_i \cdot a_{ijk} \leq L_k, \quad j \in P, \quad k \in K, \end{aligned} \quad (2.14)$$

$$\begin{aligned}
\sum_{i=1}^m \ell_i \cdot a_{ijk} &\geq L_k - \eta, & j \in P, \quad k \in K, \\
\sum_{i=1}^m a_{ijk} &\leq q, & j \in P, \quad k \in K, \\
\sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} &\leq N \cdot y_{jk}, & j \in P, \quad k \in K, \\
\sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} &\geq y_{jk}, & j \in P, \quad k \in K, \\
\sum_{k=1}^b \sum_{j=1}^p y_{jk} &\leq \varepsilon, \\
\alpha_{jkh} \text{ e } y_{jk} &\in \mathbb{B}, \quad a_{ijk} \in \mathbb{N}, \quad r_{ijkh} \in \mathbb{R}_+, \quad i \in I, \quad j \in P, \quad k \in K, \quad h \in H.
\end{aligned}$$

É considerável aumento no número de variáveis e restrições na formulação (2.14) ao fazer esta linearização, isto é, temos $(\kappa + m \cdot \kappa + m + 1) \cdot m \cdot b$ variáveis e $m + 1 + m \cdot b \cdot (3 \cdot m \cdot \kappa + 5)$ restrições.

2.2.2 Linearização usando as variáveis a_{ijk}

Um outro modo de linearizar o Problema (2.8) seria utilizar a decomposição binária de a_{ijk} . Assumimos que esta variável possua um limitante superior M_i , obtido pela Expressão (2.13). Então, escrevemos a_{ijk} em função de sua decomposição binária β_{ijkh} , temos:

$$a_{ijk} = \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \beta_{ijkh}, \quad i \in I \quad j \in P, \quad k \in K, \quad (2.15)$$

com $\xi_i = \lceil \log_2 M_i \rceil + 1$ para todo $i \in I$. Assim, a restrição bilinear de (2.8) será reescrita por:

$$\begin{aligned}
d_i &\leq \sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \\
&= \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \beta_{ijkh} \cdot x_{jk} \\
&= \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\xi_i} 2^{h-1} \cdot s_{ijkh}, \quad i \in I.
\end{aligned} \quad (2.16)$$

onde $\mathbb{R}_+ \ni s_{ijkh} = x_{jk} \cdot \beta_{ijkh}$. Desse modo, ao linearizar a forma bilinear anterior, temos o seguinte:

$$\begin{aligned}
s_{ijkh} &\leq N \cdot \beta_{ijkh}, \\
s_{ijkh} &\leq x_{jk}, \\
x_{jk} - N \cdot (1 - \beta_{ijkh}) &\leq s_{ijkh},
\end{aligned} \quad (2.17)$$

e fazendo as adaptações necessárias no modelo bilinear (2.8), temos um outro modelo linearizado alternativo descrito a seguir:

$$\begin{aligned}
\text{Minimize } & z = \sum_{k=1}^b \sum_{j=1}^p x_{jk} \\
\text{sujeito a } & \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\xi_i} 2^{h-1} \cdot s_{ijkh} \geq d_i, & i \in I, \\
& s_{ijkh} \leq N \cdot \beta_{ijkh}, & i \in I, j \in P, k \in K, h \in H_i, \\
& s_{ijkh} \leq x_{jk}, & i \in I, j \in P, k \in K, h \in H_i, \\
& x_{jk} - N \cdot (1 - \beta_{ijkh}) \leq s_{ijkh}, & i \in I, j \in P, k \in K, h \in H_i, \\
& \sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \ell_i \cdot \beta_{ijkh} \leq L_k, & j \in P, k \in K, \\
& \sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \ell_i \cdot \beta_{ijkh} \geq L_k - \eta, & j \in P, k \in K, \\
& \sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \beta_{ijkh} \leq q, & j \in P, k \in K, \\
& x_{jk} \leq N \cdot y_{jk}, & j \in P, k \in K, \\
& x_{jk} \geq y_{jk}, & j \in P, k \in K, \\
& \sum_{k=1}^b \sum_{j=1}^p y_{jk} \leq \varepsilon, \\
& \beta_{ijkh} \text{ e } y_{jk} \in \mathbb{B}, x_{jk} \in \mathbb{N}, s_{ijkh} \in \mathbb{R}_+, i \in I, j \in P, k \in K, h \in H_i.
\end{aligned} \tag{2.18}$$

Assim como na primeira estratégia de linearização, há considerável aumento no número de variáveis e restrições, isto é, $2 \cdot m \cdot b \cdot (m \cdot \kappa + 1)$ variáveis e $m + 1 + m \cdot b \cdot (3 \cdot m \cdot \kappa + 5)$ restrições nesta abordagem.

Como observado na seção anterior, esse crescimento na formulação do problema está relacionado com os limitantes das variáveis x_{jk} e a_{ijk} . A primeira estratégia é mais adequada utilizá-la quando a demanda requisitada não for alta. Por outro lado, a segunda estratégia pode ser empregada em problemas que se tenha itens com larguras grandes em relação à bobina-mestre.

Exemplo 2.1. *A fim de melhor comparar estas abordagens e ter uma melhor compreensão de como estas linearizações afetam a dimensão do problema, consideremos dois tipos de dados em um PCUI, ambos com $m = 4$, $p = 4$ padrões de corte, $L = 100$, $q = 20$ e $b = 1$ bobina-mestre.*

Na Instância 1, temos itens pequenos (em relação à bobina-mestre) e uma demanda baixa; na Instância 2, temos itens grandes e demanda alta.

Na tabela a seguir ilustramos o número de variáveis para o modelo bilinear, o

Tabela 2 – Dados para dois exemplares para o PCUI

Item	Instância 1		Instância 2	
	ℓ_i	d_i	ℓ_i	d_i
1	2	10	20	100
2	4	12	40	120
3	6	14	60	140
4	8	16	80	160
M	$(20, 20, 16, 12)^T$		$(5, 2, 1, 1)^T$	
N	16		160	
κ	5		9	
ξ	$(6, 6, 5, 5)^T$		$(4, 2, 1, 1)^T$	

que considera todas as combinações de corte (parametrizado) e as duas abordagens com linearizações.

Tabela 3 – Número de variáveis e restrições para o PCUI para dois exemplos numéricos

Modelo	Variáveis								Restrições	
	Binárias		Inteiras		Contínuas		Total			
	Inst.1	Inst.2	Inst.1	Inst.2	Inst.1	Inst.2	Inst.1	Inst.2	Inst.1	Inst.2
Bilinear	0	0	20	20	0	0	20	20	16	16
Param.	0	0	145	11	0	0	145	11	435	37
Bin. x_j	20	36	20	20	80	144	120	200	261	453
Bin. a_{ij}	92	36	4	4	92	36	188	132	309	121

Como observado, a abordagem bilinear envolve poucas variáveis. O agravante é a não linearidade e a não convexidade das restrições de demanda, tornando a aplicação dos métodos de Programação Não Linear Inteira restrito.

A parametrização também torna-se uma técnica complicada de ser utilizada. Neste pequeno exemplar, o número de variáveis inteiras foi igual à 145. No entanto, este número pode ser ainda maior dependendo dos dados do problema.

Como discutido anteriormente, as linearizações provocam um forte aumento na quantidade de variáveis e restrições em relação à formulação original. A escolha de qual linearização utilizar está ligada àquela que gerar um menor número de variáveis inteiras. No nosso exemplo, é natural que a binarização em x_{jk} é mais adequada para a Instância 1, ao passo que na Instância 2 é mais conveniente o uso da binarização em a_{ijk} .

Exemplo 2.2. Consideramos os mesmos dados $m = 4$, $p = 4$, $L = 100$, $q = 20$ e $b = 1$ mas com as demandas em posições trocadas.

Na instância 3, temos itens pequenos (em relação à bobina-mestre) e uma demanda alta; na instância 4 temos itens grandes mas uma demanda baixa.

Na tabela a seguir resumimos o número de variáveis para o modelo bilinear, parametrizado e as duas abordagens com linearizações.

Tabela 4 – Dados para dois exemplares para o PCUI

Item	Instância 3		Instância 4	
	ℓ_i	d_i	ℓ_i	d_i
1	2	100	20	10
2	4	120	40	12
3	6	140	60	14
4	8	160	80	16
M	$(20, 20, 16, 12)^T$		$(5, 2, 1, 1)^T$	
N	160		16	
κ	9		5	
ξ	$(6, 6, 5, 5)^T$		$(4, 2, 1, 1)^T$	

Tabela 5 – Número de variáveis e restrições para o PCUI para dois exemplos numéricos

Modelo	Variáveis								Restrições	
	Binárias		Inteiras		Contínuas		Total		Inst.3	Inst.4
	Inst.3	Inst.4	Inst.3	Inst.4	Inst.3	Inst.4	Inst.3	Inst.4		
Bilinear	0	0	20	20	0	0	20	20	16	16
Param.	0	0	145	11	0	0	145	11	435	37
Bin. x_j	36	20	20	20	144	80	200	120	457	265
Bin. a_{ij}	92	36	4	4	92	36	188	76	309	121

A dimensão da formulação com a parametrização não foi alterada, pois o número de combinações de corte não depende da demanda.

A primeira estratégia pode ser considerada mais conveniente na instância 3, pois usa um menor número de variáveis inteiras (56 contra 96) mesmo gerando mais restrições. Para a instância 4, a segunda estratégia é mais recomendada pelo menor número de variáveis e restrições surgidas.

2.3 Abordagem multiobjetiva do problema de corte

2.3.1 Motivação

No Problema de Corte Unidimensional Inteiro Multiobjetivo (PCUIM), objetivamos minimizar, ao mesmo tempo, a soma das frequências dos padrões de corte para atender a demanda (z_1) e ao número de diferentes padrões de corte, denominado de *setup* (z_2). Este segundo objetivo tem a mesma importância que z_1 , de sorte que as mudanças de padrões de corte ocasionam sérios efeitos negativos no planejamento da produção, como custo e tempo para mudanças (isto é, ajustes) das facas nas máquinas que realizam os cortes na bobina-mestre. Ao longo deste texto, vamos evidenciar que existe um grau de conflito entre z_1 e z_2 , isto é, quando um objetivo é individualmente minimizado, o outro é piorado. Dito de outra forma, não existe uma solução que minimize z_1 e z_2 ao mesmo tempo. Nesse caso, a solução deste problema multiobjetivo será dada por um conjunto não unitário

de soluções, ditas *eficientes* no sentido de Pareto, de maneira que um objetivo não pode ser minimizado sem aumentar o outro e vice-versa. Como nós destacamos no Capítulo 1, estas soluções fornecem um compromisso entre os objetivos considerados e possuem igual importância para o problema, fornecendo ao gerente de produção, um conjunto de alternativas distintas para uma tomada de decisão mais fundamentada, abrangente e segura.

A título de ilustração, considere os dados do Exemplo (2.2). Se o objetivo fosse apenas minimizar z_1 , teríamos uma solução ótima com um *setup* de máquina igual à $z_2^* = 4$ e $z_1^* = 39$. Por outro lado, se minimizássemos z_2 , encontraríamos uma outra solução com *setup* de 1 mas um total de 140 a frequência dos padrões usados, muito acima do mínimo desejado, provocando um excesso de produção elevado. Contudo, no contexto de otimização multiobjetivo, estas duas soluções extremas que resultaram nesses valores objetivos possuem a mesma importância para o problema.

Uma maneira de resolver o PCUIM (ou qualquer outro problema multiobjetivo) seria atribuir um peso à cada função-objetivo e construir uma única função para ser minimizada. Por exemplo, se para um primeiro gerente for mais importante minimizar z_1 do que z_2 , então a otimização da função ponderada $z = w_1 \cdot z_1 + w_2 \cdot z_2$ com $w_1 > w_2$ seria razoável, pois enfatizaremos minimizar o termo z_1 . Se para um outro gerente, a economia na troca de padrões de corte pode ser muito mais importante do que minimizar a frequência dos mesmos, a função-objetivo ponderada cujos pesos satisfazem a relação $w_2 > w_1$ poderia ser empregada, de sorte que forneceria uma solução com mais compromisso em minimizar z_2 do que em z_1 . No entanto, não sabemos exatamente o valor desses pesos, que de um modo geral, são informações subjetivas, provenientes de quem está resolvendo o problema. Além disso, uma sensível modificação neles poderia produzir soluções muito diferentes entre si. Consequentemente, otimizar um problema com múltiplos objetivos adotando uma e particular atribuição de pesos, pode parecer um processo viciado e subjetivo.

Nesse contexto, a principal vantagem de uma abordagem multiobjetiva é dar um conjunto de *soluções eficientes*, que não são avaliadas por uma função escalar comum. O processo de otimização não é “tendencioso” para um determinado tipo de solução que depende destas ponderações. Ao final do processo da OM, a intenção é fornecer ao decisor um conjunto não unitário soluções com características distintas mas igualmente importantes ao problema. Após cuidadosa análise do gerente e baseado em sua escolha, ele será capaz de optar por aquela que mais o satisfaz, caracterizando uma tomada de decisão mais segura e confiável.

Esta maneira de resolver o PCUIM via métodos clássicos de OM é inédito. A maior parte dos métodos para solucionar este problema não geram um conjunto de soluções eficientes, e sim uma única solução. Além disso, os métodos em sua maioria são heurísticos e tentam encontrar uma solução aproximada usando a relaxação linear do modelo sem

o *setup* e depois empregando procedimentos para tentar reduzir z_2 sem aumentar z_1 . Os trabalhos propostos em (BELOV; SCHEITHAUER, 2002) e em (UMETANI; YAGIURA; IBARAKI, 2003) fazem uso desta ideia. O trabalho desenvolvido em (LEDUÍNO, 2005) usa um método diferente e propõe suavizar uma função não linear e não contínua que pondera os dois objetivos, adaptando o método de GC. Na literatura, encontramos apenas o trabalho proposto em (KOLEN; SPIEKSMAN, 2000) que explorou o aspecto biobjetivo do PCUIM. No entanto, o método exato desenvolvido é aplicável para instâncias com poucos itens, sendo restrita sua utilização para exemplares práticos de tamanho moderado.

As duas subseções tratam o PCUIM de maneira distinta, no que tange aos padrões de corte serem ou não fornecidos pelo usuário.

2.3.2 Modelo multiobjetivo sem os padrões de corte a priori

Nessa abordagem, propomos resolver um modelo para PCUIM onde as combinações de corte não sejam dadas pelo usuário a priori. Este modelo não linear para o PCUIM é apresentado a seguir:

$$\begin{aligned}
\text{Minimize } (z_1, z_2) &= \left(\sum_{k=1}^b \sum_{j=1}^p x_{jk}, \sum_{k=1}^b \sum_{j=1}^p y_{jk} \right) \\
\text{sujeito a } &\sum_{k=1}^b \sum_{j=1}^p a_{ijk} \cdot x_{jk} \geq d_i, & i \in I, \\
&\sum_{i=1}^m \ell_i \cdot a_{ijk} \leq L_k, & j \in P, \quad k \in K, \\
&\sum_{i=1}^m \ell_i \cdot a_{ijk} \geq L_k - \eta, & j \in P, \quad k \in K, \\
&\sum_{i=1}^m a_{ijk} \leq q, & j \in P, \quad k \in K, \\
&x_{jk} \leq N \cdot y_{jk}, & j \in P, \quad k \in K, \\
&x_{jk} \geq y_{jk}, & j \in P, \quad k \in K, \\
&y_{jk} \in \mathbb{B}, \quad x_{jk} \text{ e } a_{ijk} \in \mathbb{N}, & i \in I, \quad j \in P, \quad k \in K.
\end{aligned} \tag{2.19}$$

Como foi discutido nas seções anteriores, esta formulação tem sérios agravantes, principalmente por não ser convexa, portanto uma versão linear do problema deve ser construída. Além disso, há uma diferença entre a Formulação (2.8) e a Formulação (2.19), que consiste na inserção de mais um objetivo z_2 também a ser minimizado. Esta alteração faz com que o novo problema, tome outra dimensão, no tocante ao conjunto de técnicas e ferramentas específicas que necessitam ser empregadas para solucioná-lo. Como discutimos no Capítulo 1, uma das maneiras de resolver um POM consiste em transformá-lo em vários problemas mono-objetivo (subproblemas) e otimizá-los pelas técnicas de programação matemática inteira conhecidas como *Branch-and-Bound*. Sob certas condições cada um

desses subproblemas fornece uma solução eficiente para o PCUIM. A dificuldade reside no fato de que cada subproblema é um Problema de Programação Linear Inteiro. Dependendo do número de vezes e grau de dificuldade para otimizar cada um desses subproblemas, a obtenção de todas as soluções eficientes pode se tornar muito lento.

A fim de utilizar pacotes computacionais de programação linear mista para resolução de cada subproblema, é conveniente linearizar a Formulação (2.19), assim como fizemos para a versão com único objetivo. Adotando-se a formulação com binarização sobre as variáveis x_{jk} e usando a dedução que gerou o Modelo (2.14) mono-objetivo, é apresentado um modelo de biobjetivo linear a seguir:

$$\begin{aligned}
 \text{Minimize} \quad & (z_1, z_2) = \left(\sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh}, \sum_{k=1}^b \sum_{j=1}^p y_{jk} \right) \\
 \text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\kappa} 2^{h-1} \cdot r_{ijkh} \geq d_i, & i \in I, \\
 & r_{ijkh} \leq a_{ijk}, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H, \\
 & a_{ijk} - M \cdot (1 - \alpha_{jkh}) \leq r_{ijkh}, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H, \\
 & \sum_{i=1}^m \ell_i \cdot a_{ijk} \leq L_k, & j \in P, \quad k \in K, \\
 & \sum_{i=1}^m \ell_i \cdot a_{ijk} \geq L_k - \eta, & j \in P, \quad k \in K, \\
 & \sum_{i=1}^m a_{ijk} \leq q, & j \in P, \quad k \in K, \\
 & \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} \leq N \cdot y_{jk}, & j \in P, \quad k \in K, \\
 & \sum_{h=1}^{\kappa} 2^{h-1} \cdot \alpha_{jkh} \geq y_{jk}, & j \in P, \quad k \in K, \\
 & \alpha_{jkh} \text{ e } y_{jk} \in \mathbb{B}, \quad a_{ijk} \in \mathbb{N}, \quad r_{ijkh} \in \mathbb{R}_+, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H,
 \end{aligned} \tag{2.20}$$

Da mesma forma, o problema (2.19) pode ser linearizado utilizando-se as variáveis a_{ijk} . Nesse caso, a formulação alternativa e linearizada é a seguinte:

$$\begin{aligned}
 \text{Minimize} \quad & (z_1, z_2) = \left(\sum_{k=1}^b \sum_{j=1}^p x_{jk}, \sum_{k=1}^b \sum_{j=1}^p y_{jk} \right) \\
 \text{sujeito a} \quad & \sum_{k=1}^b \sum_{j=1}^p \sum_{h=1}^{\xi_i} 2^{h-1} \cdot s_{ijkh} \geq d_i, & i \in I, \\
 & s_{ijkh} \leq N \cdot \beta_{ijkh}, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H_i, \\
 & s_{ijkh} \leq x_{jk}, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H_i, \\
 & x_{jk} - N \cdot (1 - \beta_{ijkh}) \leq s_{ijkh}, & i \in I, \quad j \in P, \quad k \in K, \quad h \in H_i,
 \end{aligned} \tag{2.21}$$

$$\begin{aligned}
\sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \ell_i \cdot \beta_{ijkh} &\leq L_k, & j \in P, \quad k \in K, \\
\sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \ell_i \cdot \beta_{ijkh} &\geq L_k - \eta, & j \in P, \quad k \in K, \\
\sum_{i=1}^m \sum_{h=1}^{\xi_i} 2^{h-1} \cdot \beta_{ijkh} &\leq q, & j \in P, \quad k \in K, \\
x_{jk} &\leq N \cdot y_{jk}, & j \in P, \quad k \in K, \\
x_{jk} &\geq y_{jk}, & j \in P, \quad k \in K, \\
\beta_{ijkh} \text{ e } y_{jk} &\in \mathbb{B}, \quad x_{jk} \in \mathbb{N}, \quad s_{ijkh} \in \mathbb{R}_+, \quad i \in I, \quad j \in P, \quad k \in K, \quad h \in H_i.
\end{aligned}$$

O uso das Formulação (2.20) ou da Formulação (2.21) irá depender do tipo de dados do problema a ser resolvido. Dentre estas duas formas, aquela que gerar o menor número de variáveis e restrições será utilizada.

Contudo, independentemente do uso de uma ou outra linearização, ambas geram um excessivo aumento no número de variáveis inteiras e restrições, tornando sua aplicação apenas para instâncias de tamanho muito reduzido. A seção de resultados computacionais ilustrará esta situação e justificará o motivo de termos usado os padrões de corte gerados pelo método de Gilmore e Gomory, como vamos evidenciar na próxima subseção.

2.3.3 Modelo multiobjetivo com os padrões de corte a priori

Como discutido anteriormente, solucionar o PCUIM pode-se tornar uma tarefa custosa computacionalmente, principalmente quando os padrões de corte não são parâmetros. O principal agravante é o número de variáveis inteiras das formulações resultantes com o uso dessas linearizações. Outro ponto é a questão dos limitantes N e M_i para as variáveis, que necessitam ser considerados. Esses limitantes fazem a relaxação linear ficar muito fraca, tornando o método *Branch-and-Bound* lento e o esforço para otimizar cada subproblema ser grande. Somado a esse aspecto, vários subproblemas desta natureza devem ser resolvidos para garantir um conjunto de soluções não unitário, o que torna o procedimento de programação multiobjetivo custoso para o PCUIM.

O foco deste trabalho consiste em testar, comparar e desenvolver diferentes métodos de escalarização para o PCUIM. A aplicação dos modelos vistos na subseção anterior é restrita. A fim de usar a aplicação de técnicas de programação multiobjetivo neste problema, resolvemos utilizar apenas p padrões de corte a priori, simplificando a formulação de nosso problema. Parte dos padrões que vamos utilizar provém do método de Gilmore e Gomory, apresentado em (GILMORE; GOMORY, 1961) e são “bons” padrões², uma vez que a relaxação linear com eles ser forte. Com essa simplificação, conseguimos resolver PCUIMs de instâncias de moderado a grande escala num tempo computacional razoável, fornecendo um conjunto de soluções amplo e de boa qualidade ao decisor.

² Um bom padrão de corte é aquele onde nenhum item pode ser mais cortado.

Nesse sentido é possível redefinir o PCUIM eliminando-se as Restrições (2.1) que impõem a definição de um padrão de corte. O modelo simplificado se torna o seguinte:

$$\begin{aligned}
\text{Minimize } (z_1, z_2) &= \left(\sum_{k=1}^b \sum_{j=1}^p x_{jk}, \sum_{k=1}^b \sum_{j=1}^p y_{jk} \right) \\
\text{sujeito a } \sum_{k=1}^b \sum_{j=1}^p a_{ijk}^* \cdot x_{jk} &\geq d_i, & i \in I, \\
x_{jk} &\leq N_j \cdot y_{jk}, & j \in P, \quad k \in K, \\
x_{jk} &\geq y_{jk}, & j \in P, \quad k \in K, \\
y_{jk} &\in \mathbb{B}, \quad x_{jk} \in \mathbb{N}, & j \in P, \quad k \in K,
\end{aligned} \tag{2.22}$$

onde N_j pode ser calculado levando-se em conta os dados dos padrões de corte que temos, isto é:

$$N_j = \max_{1 \leq k \leq b} \left\{ \max_{1 \leq i \leq m} \left\{ \left\lceil \frac{d_i}{a_{ijk}} \right\rceil \right\} \right\}. \tag{2.23}$$

Com esse limitante menor, é capaz de gerar uma formulação mais próxima do envoltório convexo do PLI, tornando o *Branch-and-Bound* mais rápido.

Esta abordagem leva a um modelo matemático biobjetivo com apenas $2 \cdot p$ variáveis e $2 \cdot m + p \cdot b$ restrições, menor e mais fácil de ser resolvido que as formulações apresentadas na subseção anterior.

Na perspectiva de fornecer os padrões de corte ao modelo, um outro aspecto é estudado nesta trabalho. Com a intenção de acelerar ainda mais o processo de obtenção de todas as soluções potencialmente eficientes para o PCUIM, adotamos um procedimento de relaxar parcialmente a Formulação (2.22) e em seguida utilizar uma heurística desenvolvida, que integraliza cada solução eficiente relaxada encontrada, capaz de gerar soluções inteiras para o problema. A pergunta que surge neste contexto é: “*Esta estratégia de relaxação e posteriormente um arredondamento de cada solução eficiente, produz soluções inteiras potencialmente eficientes de boa qualidade?*”. A resposta para esta pergunta é “sim”, e será respondida no capítulo dos resultados computacionais.

A formulação que relaxa a integralidade de x_{jk} e mantém o y_{jk} binário é expressa a seguir:

$$\begin{aligned}
\text{Minimize } (z_1, z_2) &= \left(\sum_{k=1}^b \sum_{j=1}^p x_{jk}, \sum_{k=1}^b \sum_{j=1}^p y_{jk} \right) \\
\text{sujeito a } \sum_{k=1}^b \sum_{j=1}^p a_{ijk}^* \cdot x_{jk} &\geq d_i, & i \in I, \\
x_{jk} &\leq N_j \cdot y_{jk}, & j \in P, \quad k \in K, \\
x_{jk} &\geq y_{jk}, & j \in P, \quad k \in K, \\
y_{jk} &\in \mathbb{B}, \quad x_{jk} \in \mathbb{R}_+, & j \in P, \quad k \in K.
\end{aligned} \tag{2.24}$$

As soluções provenientes desta formulação não são potencialmente inteiras nas componentes em x . Uma estratégia de arredondamento foi desenvolvida neste trabalho, consome um baixo tempo computacional e produz soluções integralizadas de ótima qualidade se comparada com a resultante da Formulação (2.22). No Capítulo 7 esta afirmação é sustentada e comprovada pelos testes numéricos.

Neste capítulo apresentamos diferentes modelos de problemas de corte, com ou sem os padrões de corte e nas abordagens mono e multiobjetiva. Também, para a versão não linear, deduzimos modelos equivalentes e lineares, que do ponto de vista teórico possuem características desejáveis. Vários aspectos matemáticos e computacionais destas formulações foram discutidos. No Capítulo 3, apresentaremos os diferentes métodos que resolvem um POM via escalarizações.

Capítulo 3

Métodos clássicos de otimização multiobjetivo

Neste capítulo, damos uma breve revisão teórica dos principais métodos *clássicos* de escalarização para geração de soluções de eficientes para o PCUIM. Estes métodos têm a propriedade de garantirem, matematicamente, que as soluções obtidas são eficientes. Como no presente caso \mathcal{X}^* é discreto, nossa intenção seria gerar todos os elementos desse conjunto. Contudo, algumas técnicas possuem a desvantagem de apenas gerar um subconjunto de \mathcal{X}^* , principalmente porque o PCUIM possui uma fronteira de Pareto não convexa.

Os métodos clássicos de escalarização apresentados transformam o problema multiobjetivo em vários subproblemas escalares, que sob certas condições, geram uma solução eficiente. Por vezes, as técnicas de escalarizações podem ser muito custosas computacionalmente, pois dependem fortemente do desempenho dos métodos que resolvem cada subproblema. Além disso, para um número grande de subproblemas, estes métodos podem gerar apenas uma quantidade pequena de soluções eficientes distintas assim como gerar uma má distribuição e pouca variabilidade dos pontos não dominados ao longo da fronteira de Pareto.

Neste estudo, aplicamos os métodos descritos a seguir e posteriormente realizaremos uma análise comparativa do desempenho entre eles. Este capítulo foi baseado nas referências clássicas de OM, como as apresentadas em (COHON, 2003), (DEB; KALYANMOY, 2001), (EHRGOTT, 2000) e (MIETTINEN, 1999).

3.1 O Método da Soma Ponderada

O Método da Soma Ponderada escala o conjunto das funções-objetivos do problema multiobjetivo original e o transforma num problema mono-objetivo ponderado. Isto é feito tomando uma combinação convexa de cada função-objetivo considerada, sendo w_k o peso para cada uma delas, que variado, permitirá a geração de outras soluções

eficientes distintas. Este método é o mais simples e fácil de ser implementado, por isso é o mais conhecido e difundido na literatura.

Embora a ideia seja muito simples, uma questão importante é mensurar estes pesos para cada objetivo. Obviamente, quanto mais (menos) importante for aquele critério no problema, maior (menor) deve ser o valor do peso w_k associado para aquele objetivo.

Atribuir estes pesos requer uma tarefa adicional de não permitir que a magnitude de cada função-objetivo não afete a geração de soluções eficientes. Por exemplo, em um problema biobjetivo onde uma função-objetivo visa minimizar o custo de produção que é da ordem de R\$ 10^6 e a outra é o desperdício que varia no intervalo de $[10^{-2}, 10^{-1}]$ m², é recomendável colocar estas funções numa mesma ordem de grandeza, mantendo-as numa mesma faixa de variação. Isto pode ser feito multiplicando a primeira função por 10^{-6} e a segunda por 10 (normalização).

Após normalizá-las, uma composição dos objetivos z^w é formada e temos um POM transformado em uma sequência de subproblemas de otimização mono-objetivo, como a seguir:

$$\begin{aligned} \text{Minimize} \quad & z^w = \sum_{k=1}^r w_k \cdot f_k(x) \\ \text{sujeito a} \quad & x \in \mathcal{X}, \end{aligned} \tag{3.1}$$

onde sem perda de generalidade $\sum_{k=1}^r w_k = 1$ e $w_k \in [0, 1]$. Assumimos que as funções $z_k = f_k(x)$ já estejam normalizadas.

Algumas propriedades e teoremas desta técnica são apresentados a seguir.

Teorema 3.1. *Se $w_k > 0$ para todo $k = 1, \dots, r$, então a solução do Problema (3.1) é eficiente para o POM original.*

Demonstração. Seja $x^* \in \mathcal{X}$ uma solução ótima para o Problema (3.1), com os pesos satisfazendo as hipóteses acima. Suponha que x^* não seja eficiente, logo existe uma solução $\bar{x} \in \mathcal{X}$ de modo que

$$f_k(\bar{x}) \leq f_k(x^*), \quad k = 1, \dots, r$$

e

$$f_i(\bar{x}) < f_i(x^*)$$

para ao menos um $i \in \{1, \dots, r\}$. Então,

$$\sum_{k=1}^r w_k \cdot f_k(\bar{x}) < \sum_{k=1}^r w_k \cdot f_k(x^*),$$

mas isto contradiz o fato de x^* ser solução ótima de (3.1), logo x^* é uma solução eficiente para o POM. \square

O teorema a seguir exige apenas que $w_i \geq 0$ sobre os pesos. Por outro lado, usa como hipótese uma condição mais forte sobre o tipo de solução do problema ponderado.

Teorema 3.2. *Se o Problema (3.1) tem solução uma única x^* , então x^* é eficiente.*

Demonstração. Seja x^* a única solução do problema ponderado. Suponha que ela não seja eficiente, logo, existe uma outra solução factível $\bar{x} \neq x^*$ de modo que

$$f_k(\bar{x}) \leq f_k(x^*), \quad k = 1, \dots, r$$

e

$$f_i(\bar{x}) < f_i(x^*)$$

para ao menos um $i \in \{1, \dots, r\}$. Como $w_k \geq 0$ para todo $k = 1, \dots, r$, temos que

$$\sum_{k=1}^r w_k \cdot f_k(\bar{x}) \leq \sum_{k=1}^r w_k \cdot f_k(x^*).$$

Como x^* é solução única, podemos escrever que

$$\sum_{k=1}^r w_k \cdot f_k(x^*) < \sum_{k=1}^r w_k \cdot f_k(\bar{x}),$$

mas isto contradiz a desigualdade imediatamente anterior, logo x^* é eficiente. \square

A recíproca do Teorema (3.1), em geral, não é verdadeira. Dito de outra forma, dada qualquer solução eficiente x^* , não existe um vetor de pesos w de modo que x^* possa ser obtido como solução ótima do Problema (3.1). Contudo, para problemas multiobjetivos convexos a recíproca do teorema anterior é verdadeira.

Teorema 3.3. *Se x^* é uma solução eficiente para o problema multiobjetivo (1.1) convexo, então existe um vetor $w = (w_1, \dots, w_r)^T \geq 0$ com $\sum_{k=1}^r w_k = 1$, tal que x^* é solução do Problema (3.1).*

Demonstração. Apresentada após o Teorema (3.14). \square

O Teorema (3.3) mostra que o Método da Soma Ponderada é capaz de encontrar *todas* as soluções eficientes para POM convexos (e fronteira de Pareto será convexa) desde que w seja adequadamente variado. Não há regras claras de como fazer esta variação, tampouco a unicidade deste vetor peso para cada solução eficiente encontrada. Isto quer dizer que vários vetores pesos distintos podem resultar numa mesma solução eficiente e uma variação uniforme de w , em geral, não resultar em pontos não dominados uniformemente espaçados na fronteira.

A Figura (7) ilustra a aplicação desta técnica para um POM não convexo com $r = 2$ objetivos. Note que a fronteira de Pareto (curva em vermelho), é não convexa.

Geometricamente, dados $w_1 \geq 0$ e $w_2 \geq 0$, a imagem da solução de (3.1) é um ponto do espaço \mathcal{Z} , que é obtido pela intersecção da reta $z^w = w_1 \cdot z_1 + w_2 \cdot z_2$ com o espaço \mathcal{Z} . Esta reta, com inclinação $-w_1/w_2$, nada mais é do que curva de nível da função ponderada. A minimização da função ponderada se transforma no problema de encontrar o ponto em que a reta com a dada inclinação tangencie o conjunto \mathcal{Z} .

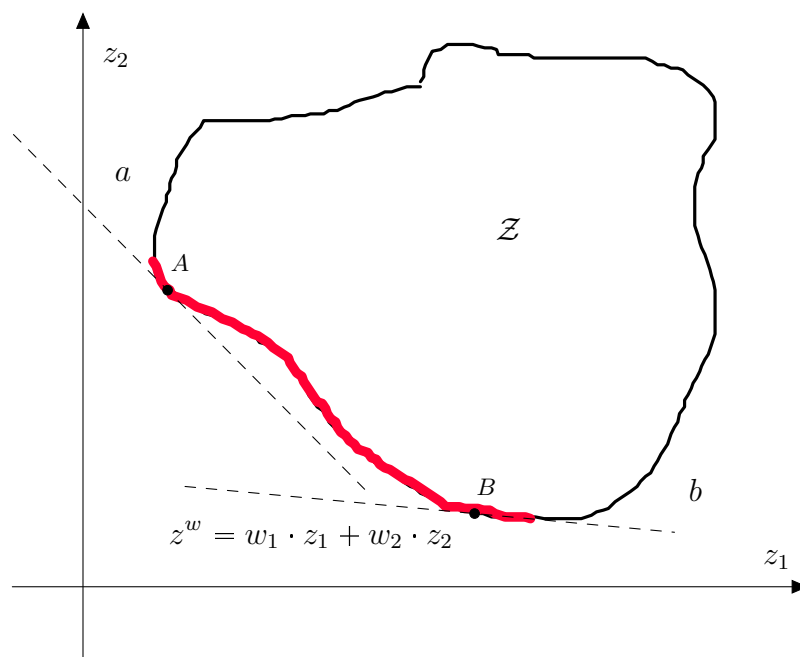


Figura 7 – Exemplo de aplicação da Soma Ponderada para problemas não convexos

Conforme os pesos w_1 e w_2 são variados, estas retas passam a ter declividades diferentes e outras soluções eficientes são obtidas. Note que as soluções contidas na região não convexa (entre os pontos A e B) da fronteira de Pareto nunca poderão ser obtidas por este procedimento.

A Figura (8) ilustra um POM convexo e a interpretação geométrica do Teorema (3.3), com diferentes soluções não dominadas sendo captadas pelo método Soma Ponderada. Note que para qualquer ponto não dominado na curva em vermelho, sempre existirá uma reta que o tangenciará. Isto é exatamente a existência destes vetores-pesos e a interpretação do teorema precedente.

Observação 3.1. *É necessário um cuidado adicional na resolução do problema ponderado quando um ou mais pesos são iguais a zero. Se ele tiver múltiplas soluções, então estas podem não ser eficientes. A eficiência só é garantida pelo Teorema (3.1) onde $w_k > 0$ para todo $k \in \{1, \dots, r\}$ ou, alternativamente, pelo Teorema (3.2) no caso do problema ponderado fornecer solução única. O teorema a seguir atesta que este método pode fornecer soluções fracamente eficientes.*

Teorema 3.4. *Se $w_k \geq 0$ para todo $k = 1, \dots, r$ e $w_i > 0$ para algum $i \in \{1, \dots, r\}$, então a solução do Problema (3.1) é fracamente eficiente para o POM original.*

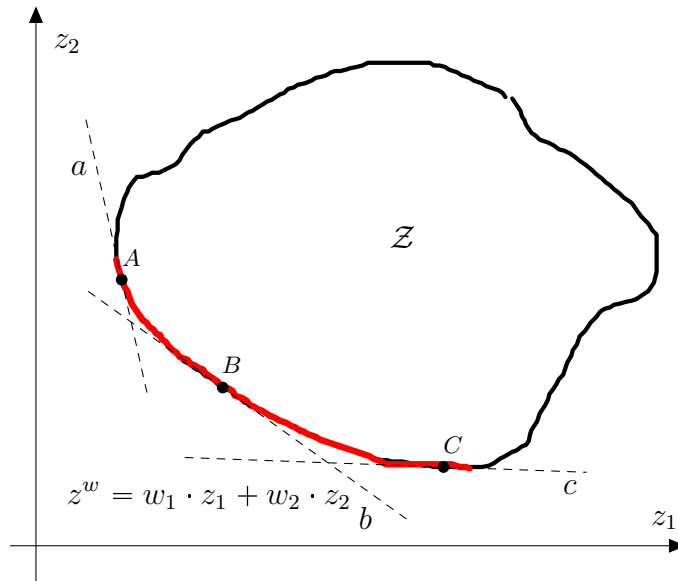


Figura 8 – Exemplo de aplicação da Soma Ponderada para problemas convexos

Demonstração. Seja $x^* \in \mathcal{X}$ uma solução ótima para o Problema (3.1), com os pesos satisfazendo as hipóteses acima. Suponha que x^* não seja fracamente eficiente, logo existe uma solução $\bar{x} \in \mathcal{X}$ de modo que

$$f_k(\bar{x}) < f_k(x^*), \quad k = 1, \dots, r.$$

Então como existe um $w_i > 0$,

$$\sum_{k=1}^r w_k \cdot f_k(\bar{x}) < \sum_{k=1}^r w_k \cdot f_k(x^*),$$

mas isto contradiz o fato de x^* ser solução ótima de (3.1), logo x^* é uma solução fracamente eficiente para o POM. \square

A Figura (9) ilustra o caso para $r = 2$ objetivos, onde $w_1 = 0$ e $w_2 > 0$. Nessa situação, a inclinação da curva de nível é igual à $-0/w_2 = 0$, uma reta paralela ao eixo z_1 . Suponha que o espaço \mathcal{Z} tenha como limitante inferior o segmento de reta BC , como na figura. Com estes pesos, o problema resultante fornecerá como solução todos os pontos no segmento BC (fracamente não dominados). No entanto, apenas B é não dominado.

Uma forma de contornar este inconveniente é sempre adotar $w_k \geq \rho$, $k = 1, \dots, r$, com ρ uma constante positiva.

A seguir, apresentamos um método que usa os princípios da Soma Ponderada, originalmente desenvolvido para Problema de Programação Lineares biobjetivos e que contorna alguns inconvenientes desta técnica.

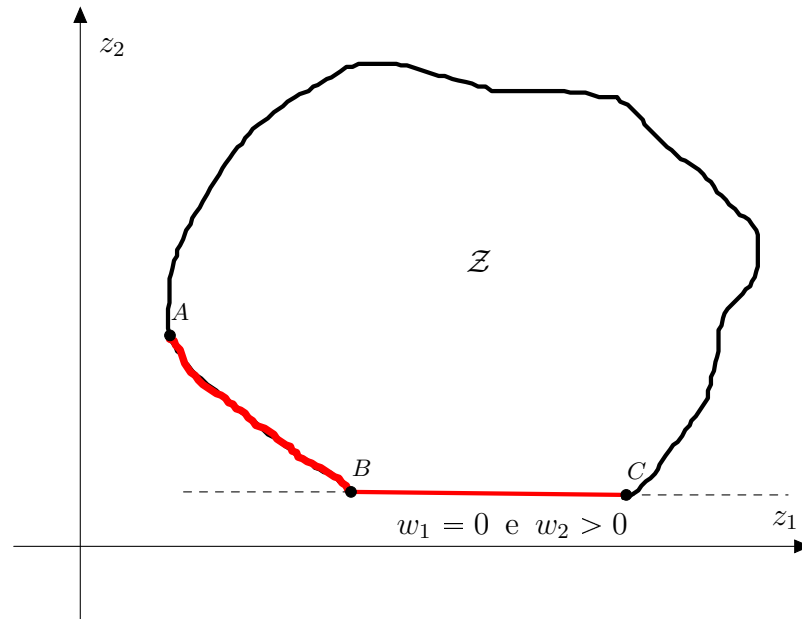


Figura 9 – Ilustração de soluções dominadas obtidas com alguns pesos nulos

3.2 O Método *Non Inferior Set Estimation*

O método *Non Inferior Set Estimation* (NISE) foi desenvolvido no trabalho em (COHON, 2003) e é especializado em resolver Problema de Programação Lineares biobjetivos

$$\begin{aligned} \text{Minimize} \quad & (z_1, z_2) = (c^T x, d^T x) \\ \text{sujeito a} \quad & Ax = b \\ & x \geq 0, \end{aligned} \tag{3.2}$$

em que $A \in \mathbb{R}^{m \times n}$, $\text{posto}(A) = m$, $b \in \mathbb{R}^m$, $c^T, d^T \in \mathbb{R}^n$, e $x \in \mathbb{R}^n$. Como a função-objetivo $z = (c^T x, d^T x)^T$ é convexa num conjunto de restrições convexo, a fronteira de Pareto para estes problemas será convexa.

3.2.1 Preliminares

Os seguintes conceitos são importantes para caracterizar soluções eficientes em um Problema de Programação Linear biobjetivo.

Definição 3.1 (Base eficiente). *Uma base $B \in \mathbb{R}^{m \times m}$ é eficiente, se é solução para o Problema de Programação Linear ponderado*

$$\begin{aligned} \text{Minimize} \quad & z = w_1 c^T x + w_2 d^T x \\ \text{sujeito a} \quad & Ax = b \\ & x \geq 0, \end{aligned}$$

com $0 \neq w = (w_1, w_2) \geq 0$.

Definição 3.2 (Bases adjacentes). *Uma base B é adjacente com uma outra base \tilde{B} se uma pode ser obtida da outra com um simples pivoteamento.*

Se B_1 e B_2 são eficientes e adjacentes, tendo como soluções x^1 e x^2 associadas à estas bases, respectivamente, então a combinação linear convexa destas soluções são eficientes. Baseada nestas duas definições, o seguinte teorema, demonstrado em (EHRGOTT, 2000), é muito útil:

Teorema 3.5. *Todas as bases de um Problema de Programação Linear multiobjetivo são conectadas, isto é, vizinhas duas a duas.*

Assim como em Programação Linear com única função-objetivo¹, as soluções básicas são caracterizadas geometricamente pelos vértices que definem o poliedro da região admissível. Dessa forma, resolver o Problema (3.2), resume-se em encontrar todos os vértices eficientes (bases eficientes), e assim, o conjunto eficiente será a união dos segmentos que ligam estes vértices. A Figura (10) ilustra três bases eficientes e \mathcal{X}^* sendo os segmentos conectados por estas bases (em vermelho).

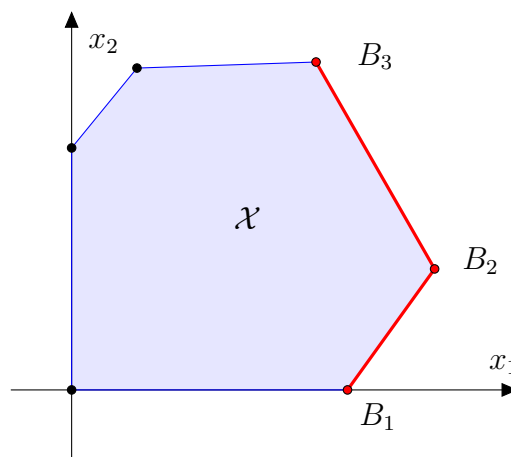


Figura 10 – Bases e soluções eficientes em um Problema de Programação Linear biobjetivo

3.2.2 Funcionamento do procedimento

Baseada nessas propriedades, o método é fundamentado na soma ponderada² dos objetivos, onde a cada iteração uma nova base eficiente é obtida. Ao final do método, temos a enumeração de todas as bases eficientes para este problema, uma vez que todas elas serão suportadas, e assim, todas as soluções eficientes para este problema também serão. O método é descrito com detalhes a seguir.

Inicialmente, os objetivos z_1 e z_2 são minimizados separadamente, resultando nas soluções x^{*1} e x^{*2} , respectivamente. Após isto, uma aproximação para a fronteira de Pareto

¹ Para mais detalhamento de Programação Linear, ver Capítulo 5, Seção 1.

² Não necessariamente os pesos estão entre 0 e 1.

pode ser obtida, traçando o segmento AB cujos extremos são os pontos lexicográficos, dados por $A = (z_1^-, z_2^+)^T$ e $B = (z_1^+, z_2^-)^T$. Se o POM presente tiver outras soluções eficientes, então estarão dentro do triângulo ABC , onde $C = z^I = (z_1^-, z_2^-)^T$ é o ponto ideal. A Figura (11) mostra esses pontos.

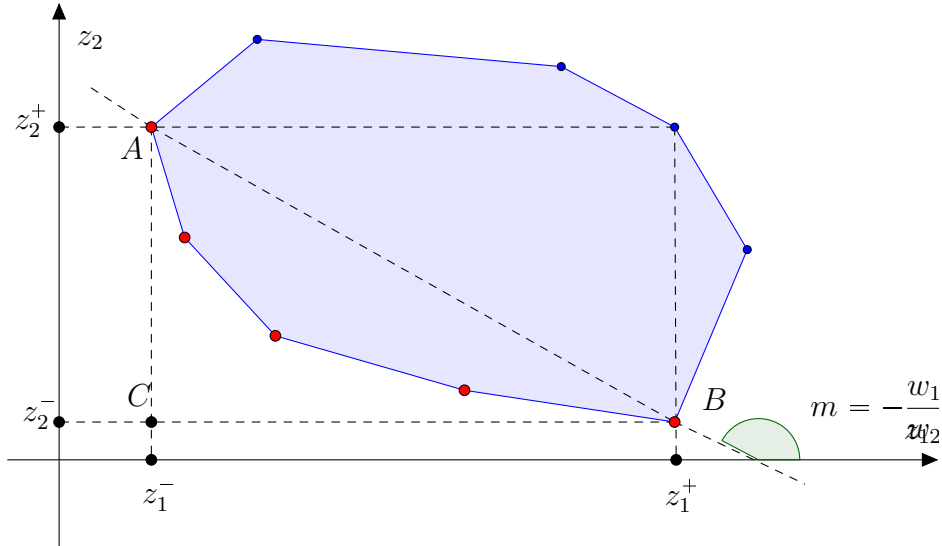


Figura 11 – Triângulo inicial que contém todos os pontos não dominados para (3.2)

A ideia do método NISE consiste em “arrastar” paralelamente o máximo possível, o segmento AB até que ele possa tangenciar o espaço \mathcal{Z} . Isto pode ser feito ao determinar a inclinação m deste segmento, que é dada por:

$$m = -\frac{w_1}{w_2} = \frac{z_2^- - z_2^+}{z_1^+ - z_1^-} < 0, \quad (3.3)$$

onde w_1 e w_2 são os pesos para as funções-objeto z_1 e z_2 . Sem perda de generalidade, podemos utilizar $w_2 = 1$. Se o Problema (3.2) possui mais que duas bases eficientes, ao resolver o problema ponderado:

$$\begin{aligned} \text{Minimize} \quad & z^w = w_1 c^T x + d^T x \\ \text{sujeito a} \quad & Ax = b \\ & x \geq 0, \end{aligned} \quad (3.4)$$

resultará em uma nova solução eficiente D no interior do triângulo ABC , tendo em vista a convexidade de \mathcal{Z}^* como pode ser visto na Figura (12).

O próximo passo é avaliar se a solução x^{w_1} do Problema (3.4) é um novo vértice no espaço objetivo, ou seja, se seu vetor objetivo está contido no segmento AB . Isso pode ser feito se analisarmos o valor ótimo z^w deste problema na solução x^{w_1} . Se esse valor é menor que o valor objetivo em qualquer solução cuja imagem é o segmento AB então teremos um novo vértice eficiente. Em outras palavras, x^{w_1} é eficiente se:

$$w_1 c^T x^{w_1} + d^T x^{w_1} < w_1 c^T x + d^T x,$$

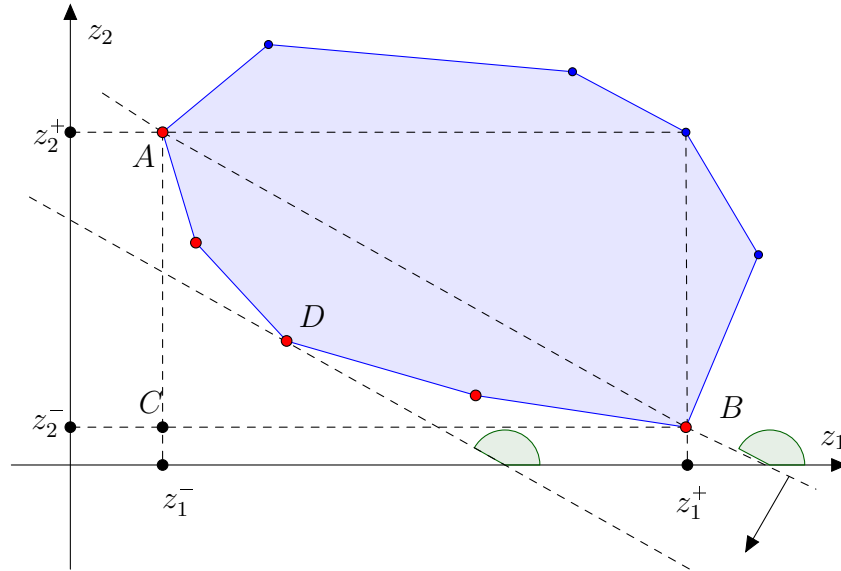


Figura 12 – Estratégia de ponderação do método NISE

onde x é qualquer solução cuja imagem pertença a AB . Em particular, esta solução pode ser os extremos deste segmento.

Se a igualdade anterior não for estrita, significa que não existe entre os pontos A e B outro ponto (vértice) não dominado, logo não existem outras soluções eficientes cuja imagem esteja entre A e B .

Caso a igualdade seja estrita, computamos o vetor objetivo de x^{w_1} e ordenamos os pontos não dominados a partir dos valores de z_1 em ordem crescente. Renomeamos os $t = 3$ pontos, temos o seguinte: $S^1 = A$, $S^2 = D$ e $S^3 = B$. Iteramos novamente utilizando dois vértices não dominados consecutivos. Após analisar os dois subproblemas e considerando os pares de pontos consecutivos S^1 e S^2 e depois S^2 e S^3 , atualizamos o valor de número de soluções eficientes, t .

As t soluções até então encontradas são organizadas em ordem crescente considerando-se a primeira função-objetivo, digamos, $A = S^1, S^2, \dots, S^t = B$, e o procedimento é novamente realizado, tomando os pares de pontos consecutivos $S^i S^{i+1}$, $1 \leq i \leq t-1$, e a inclinação do segmento que os une, que servirá de ponderação para construir os subproblemas (3.4) posteriores que serão re-otimizadas, resultando numa solução x^{w_i} . Se

$$w_1 c^T x^{w_i} + d^T x^{w_i} \leq w_1 c^T x + d^T x, \tag{3.5}$$

onde x é algum ponto do segmento $S^i S^{i+1}$ é satisfeita estritamente, então x^{w_i} é um novo vértice eficiente. Se for satisfeita na igualdade, então entre os pontos S^i e S^{i+1} não existe outra base eficiente cuja imagem esteja entre eles.

O método encerra a sua busca até a desigualdade (3.5) entre todos os t pontos consecutivos seja verificada na igualdade. A Figura (13) ilustra a ideia deste procedimento

e outras soluções sendo obtidas.

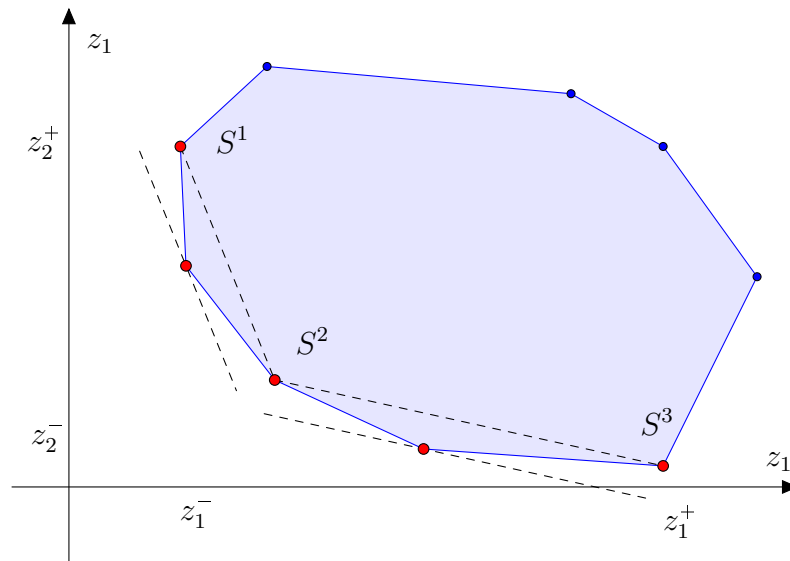


Figura 13 – Novas soluções eficientes sendo obtidas com o NISE

Na Figura (14), ilustramos S^i e S^{i+1} com $i = 1, \dots, 5$, e o Problema (3.4) não sendo capaz de “arrastar” o segmento formado por estes pontos. Isso significa que todas as bases eficientes foram encontradas para este problema.

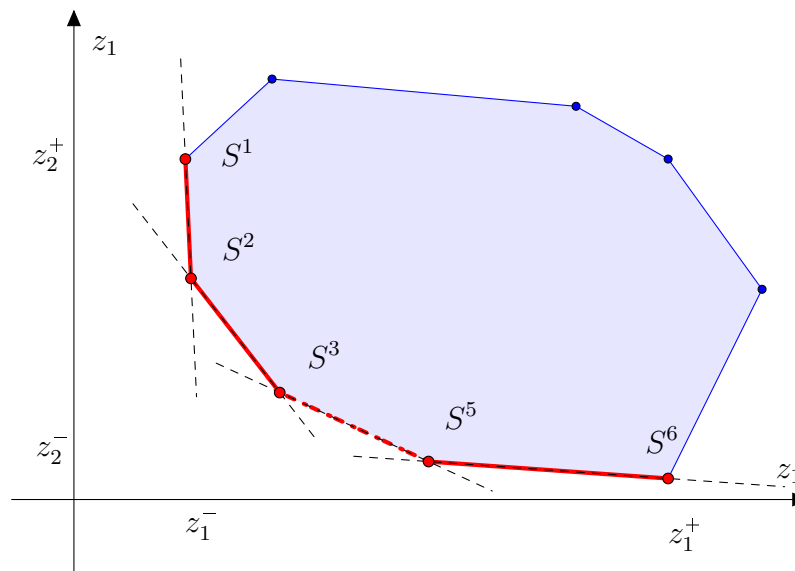


Figura 14 – Todas soluções eficientes de um POM detectadas

É válido ressaltar que este procedimento consome $2 \cdot t - 1$ subproblemas para calcular as t bases eficientes de um Problema de Programação Linear biobjetivo. Detectada quem são estas bases, o conjunto eficiente será aquele cuja imagem é a conexão destas bases (segmentos em vermelho).

3.3 O Método da Métrica de Tchebycheff

O uso da Norma de Tchebycheff em problemas de OM surgiu pioneiramente no trabalho proposto em (BOWMAN, 1976). Em vez de usar as ponderações convexas dos objetivos, outro meio de combinar múltiplos objetivos numa função escalar seria utilizar as normas ponderadas l_α em espaços euclidianos. Para pesos não negativos, um problema escalar alternativo poderia ser definido como o de minimizar, com respeito à norma ponderada α , a distância entre o vetor objetivo de uma solução x factível e o vetor z^I , isto é:

$$\begin{aligned} \text{Minimize } z^w &= \left[\sum_{k=1}^r w_k \cdot (f_k(x) - z_k^I)^\alpha \right]^{1/\alpha} \\ \text{sujeito a } x &\in \mathcal{X}, \end{aligned} \quad (3.6)$$

onde o parâmetro α pode ser escolhido pelo usuário, $1 \leq \alpha < \infty$ ou $\alpha = \infty$ e denota a métrica ponderada a ser utilizada. Assumimos que $\sum_{k=1}^r w_k = 1$, $w_k > 0$ e $f_k(x)$ estejam normalizadas, $k = 1, \dots, r$. Note que se $\alpha = 1$, o Problema (3.6) é equivalente ao problema ponderado. Neste trabalho, damos especial ênfase ao caso $\alpha = \infty$, pois as outras métricas (com exceção $\alpha = 1$) resultam em problemas escalares não lineares.

A métrica l_∞ recebe o especial nome de *Métrica de Tchebycheff*, em homenagem ao Matemático Russo Pafnuty Lvovich Tchebycheff (1821). Utilizando no contexto de OM, problema de Tchebycheff é definido da seguinte maneira:

$$\begin{aligned} \text{Minimize } z^w &= \max_{1 \leq k \leq r} \{w_k (f_k(x) - z_k^I)\} \\ \text{sujeito a } x &\in \mathcal{X}. \end{aligned} \quad (3.7)$$

No entanto, ele não é diferenciável. Inserindo a variável auxiliar $u \in \mathbb{R}_+$, podemos ter uma forma equivalente linear e diferenciável:

$$\begin{aligned} \text{Minimize } z^w &= u \\ \text{sujeito a } x &\in \mathcal{X}, \\ &w_k \cdot (f_k(x) - z_k^I) \leq u. \end{aligned}$$

Na Figura (15), ilustramos quatro soluções eficientes obtidas com diferentes ponderações nos desvios entre um ponto do espaço \mathcal{Z} e z^I usando a métrica l_∞ . Note que pontos pertencentes à região não convexa da fronteira podem ser gerados com esta técnica. Conforme os pesos variam, os contornos da métrica (em pontilhado, e que são retângulos com lados paralelos aos eixos coordenados) vão sendo modificados, de maneira que conseguem “tocar” qualquer parte da fronteira de Pareto.

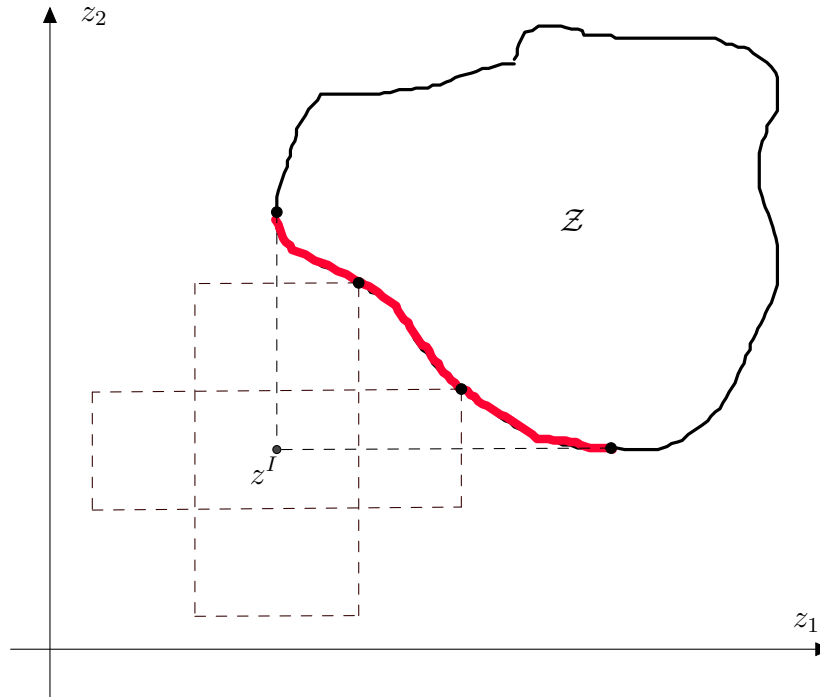


Figura 15 – Interpretação geométrica da Métrica de Tchebycheff Ponderada

Os teoremas a seguir asseguram resultados muito interessantes do ponto de vista teórico.

Teorema 3.6. *Se $w_k > 0$, $k = 1, \dots, r$, então solução do Problema (3.7) é eficiente.*

Demonstração. Seja $x^* \in \mathcal{X}$ uma solução ótima para o Problema (3.7), com os pesos satisfazendo as hipóteses acima. Suponhamos que x^* não seja eficiente, logo existe uma solução $\bar{x} \in \mathcal{X}$ de modo que $f_k(\bar{x}) \leq f_k(x^*)$, para todo $k = 1, \dots, r$ e $f_i(\bar{x}) < f_i(x^*)$ para ao menos um $i = \{1, \dots, r\}$. Então, para todo $k = 1, \dots, r$ podemos escrever que $w_k \cdot (f_k(\bar{x}) - z_k^I) \leq w_k \cdot (f_k(x^*) - z_k^I) \Leftrightarrow \max_{1 \leq k \leq r} \{w_k \cdot (f_k(\bar{x}) - z_k^I)\} < \max_{1 \leq k \leq r} \{w_k \cdot (f_k(x^*) - z_k^I)\}$, mas isto contradiz o fato de x^* ser solução ótima de (3.7), logo x^* é uma solução eficiente para o POM. \square

Teorema 3.7. *Se o Problema (3.7) tem solução única x^* , então x^* é eficiente.*

Demonstração. Seja x^* a única solução do Problema (3.7). Suponha que ela não seja eficiente, logo, existe uma outra solução factível $\bar{x} \neq x^*$ de modo que $f_k(\bar{x}) \leq f_k(x^*)$, $k = 1, \dots, r$ e $f_i(\bar{x}) < f_i(x^*)$ para ao menos um $i \in \{1, \dots, r\}$. Como $w_k \geq 0$ para todo $k = 1, \dots, r$, temos que $\max_{1 \leq k \leq r} \{w_k \cdot (f_k(\bar{x}) - z_k^I)\} < \max_{1 \leq k \leq r} \{w_k \cdot (f_k(x^*) - z_k^I)\}$. Como x^* é solução única, podemos escrever que $\max_{1 \leq k \leq r} \{w_k \cdot (f_k(x^*) - z_k^I)\} < \max_{1 \leq k \leq r} \{w_k \cdot (f_k(\bar{x}) - z_k^I)\}$ mas isto contradiz a desigualdade imediatamente anterior, logo x^* é eficiente. \square

O Teorema (3.7) alerta sobre o cuidado que devemos tomar quando o Problema de Tchebycheff Ponderado fornece múltiplas soluções; estas por sua vez podem não ser

eficientes (fracamente eficientes). No entanto, pelo Teorema (3.6) garantimos que se os pesos forem positivos, então a solução será eficiente. Não há garantia de eficiência quando temos pesos nulos. Assim, duas maneiras de evitar soluções não eficientes seria, numa primeira estratégia, adotar $w_k \geq \rho > 0$, ou considerar o Problema de Tchebycheff Aumentado:

$$\begin{aligned} \text{Minimize} \quad & z^w = u + \rho \cdot \sum_{k=1}^r (f_k(x) - z_k^I) \\ \text{sujeito a} \quad & x \in \mathcal{X}, \\ & w_k \cdot (f_k(x) - z_k^I) \leq u. \end{aligned} \tag{3.8}$$

A não convexidade de um POM é um fator importante a ser levado em consideração na escolha da técnica para resolvê-lo. A principal vantagem deste método é a capacidade de garantir *todas* as soluções eficientes para o problema, mesmo aquelas situadas na região não convexa da fronteira de Pareto, desde que os vetores pesos sejam variados de maneira adequada. O próximo teorema fornece esta garantia.

Teorema 3.8. *Se $x^* \in \mathcal{X}$ é uma solução eficiente, então existe um vetor peso $w = (w_1, \dots, w_r) > 0$ tal que x^* é solução para o Problema (3.7), onde a referência é o ponto utópico z^U .*

Demonstração. Seja x^* uma solução eficiente para o POM. Assumimos que não existe um vetor de pesos $w = (w_1, \dots, w_r) > 0$ tal que x^* não seja solução do Problema de Tchebycheff. Ora, $f_k(x) > z_k^U$ para todo $k = 1, \dots, r$ e para todo $x \in \mathcal{X}$. Logo é plausível escolhermos

$$0 < w_k = \frac{\beta}{f_k(x^*) - z_k^U}, \quad k = 1, \dots, r,$$

onde β é uma constante positiva de normalização. Como x^* não é solução para o Problema de Tchebycheff, então existe um $\bar{x} \in \mathcal{X}$ que deva ser, ou seja,

$$\begin{aligned} \max_{1 \leq k \leq r} \{w_k \cdot (f_k(\bar{x}) - z_k^U)\} &< \max_{1 \leq k \leq r} \{w_k \cdot (f_k(x^*) - z_k^U)\} \\ &= \max_{1 \leq k \leq r} \left\{ \frac{\beta}{f_k(x^*) - z_k^U} \cdot (f_k(x^*) - z_k^U) \right\} = \beta. \end{aligned}$$

Isso implica que $\max_{1 \leq k \leq r} \{w_k \cdot f_1(\bar{x}) - z_k^U\} < \beta \Leftrightarrow w_k \cdot f_1(\bar{x}) - z_k^U < \beta$, para todo $k = 1, \dots, r$, ou seja,

$$\frac{\beta}{f_k(x^*) - z_k^U} \cdot (f_1(\bar{x}) - z_k^U) < \beta \Leftrightarrow f_k(\bar{x}) < f_k(x^*)$$

para todo $k = 1, \dots, r$. Isto é uma contradição, pois assumimos que x^* era eficiente, o que completa a prova. \square

O teorema anterior fornece uma grande vantagem desta técnica em relação à Soma Ponderada, no sentido que ela é capaz de gerar soluções eficientes cuja imagem está na região não convexa da fronteira de Pareto. Em POM combinatórios, essas soluções

são as não suportadas, presentes com muita frequência em problemas desta natureza. A próxima metodologia também possui esta desejável propriedade, sendo bastante difundida na literatura especializada.

3.4 O Método do ε –Restrito

A fim de relevar as dificuldades do método da Soma Ponderada, principalmente para POM com fronteira de Pareto não convexa, a técnica ε –Restrito é um método alternativo e que supera algumas dificuldades dos métodos precedentes. Proposta em (HAIMES; LASDON; WISMER, 1971), este método escalariza um POM tomando uma função como objetivo e restringindo as demais com limitantes especificados. Quando esses limitantes variam de modo adequado, soluções eficientes podem ser obtidas. O Problema Restrito é definido como:

$$\begin{aligned} \text{Minimize} \quad & z^\varepsilon = f_i(x) \\ \text{sujeito a} \quad & x \in \mathcal{X}, \\ & f_k(x) \leq \varepsilon_k, \quad k = 1, \dots, r, \quad k \neq i, \end{aligned} \tag{3.9}$$

onde $i \in \{1, \dots, r\}$ é alguma função-objetivo previamente escolhida, ε_k é um limitante superior para a função-objetivo $f_k(x)$, $k \neq i$. Conforme estes limitantes variam, diferentes soluções eficientes podem ser obtidas com esta escalarização. Uma diferença crucial entre os três primeiros métodos aqui abordados é que a região admissível do problema escalarizado varia, enquanto nas outras abordagens apenas a função-objetivo sofre alterações.

A Figura (16) apresenta uma interpretação geométrica para este método para $r = 2$ objetivos e $i = 1$, significando que a segunda função-objetivo foi tomada como restrição.

Conforme o lado direito da restrição $f_2(x) \leq \varepsilon$ é modificado, a minimização de $f_1(x)$ leva à uma solução eficiente, desde que a solução do Problema Restrito seja única. Se ε assumir valores inteiros no intervalo $\mathcal{I} = [z_2^-, z_2^+]$, a projeção da fronteira de Pareto no eixo z_2 , então todas as soluções eficientes poderão ser obtidas, sejam elas suportadas ou não, como mostra a Figura (16). Note que se ε não estiver nesta imagem, o Problema Restrito pode ser infactível.

Os teoremas a seguir fornecem os suportes teóricos deste procedimento.

Teorema 3.9. *A solução do Problema (3.9) é fracamente eficiente.*

Demonstração. Seja $x^* \in \mathcal{X}$ uma solução para o Problema Restrito. Suponha que x^* não seja eficiente, logo existe uma outra solução $\bar{x} \neq x^*$, de modo que $f_k(\bar{x}) < f_k(x^*) \leq \varepsilon_k$, para todo $k = 1, \dots, r$. Logo \bar{x} é admissível para o Problema Restrito. Da última desigualdade, podemos afirmar em particular que $f_i(\bar{x}) < f_i(x^*)$, o que é uma contradição, pois x^* era solução ótima de (3.9). Assim, x^* é fracamente eficiente. \square

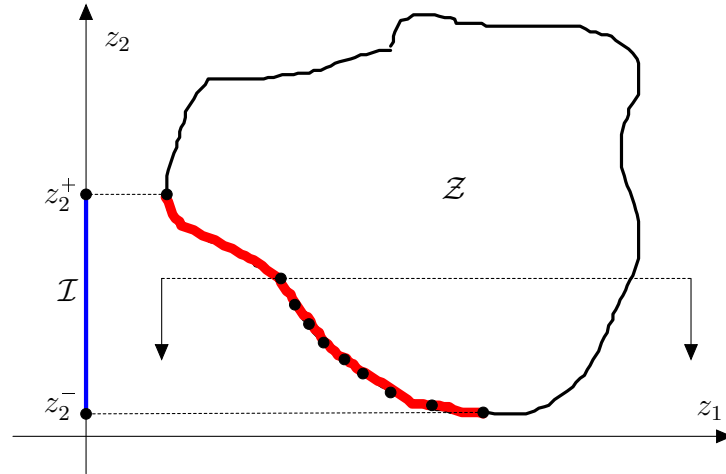


Figura 16 – Pontos na parte não convexa determinados pelo ε -R

Teorema 3.10. *Um vetor x^* é eficiente se, e somente se, é solução do Problema (3.9) para todo $i = 1, \dots, r$ e $\varepsilon_k = f_k(x^*)$, $k \neq i$.*

Demonstração. (\Rightarrow) Seja x^* eficiente, devemos mostrar que este vetor é solução do Problema Restrito para qualquer escolha para i . Para tanto, suponhamos que x^* não seja solução para este problema para algum $i \in \{1, \dots, r\}$ e $\varepsilon_k = f_k(x^*)$ para $k = 1, \dots, r$ e $k \neq i$. Então, deve existir uma outra solução $\mathcal{X} \ni \bar{x} \neq x^*$ tal que $f_i(\bar{x}) < f_i(x^*)$ e $f_k(\bar{x}) \leq f_k(x^*)$ para $k \neq i$. Isto contradiz a eficiência de x^* . Noutras palavras, x^* é solução para o Problema Restrito independente da função-objetivo tomada.

(\Leftarrow) Se $x^* \in \mathcal{X}$ é solução do Problema (3.9) para todo $i = 1, \dots, r$, então não existe um $\bar{x} \in \mathcal{X}$ tal que $f_i(\bar{x}) < f_i(x^*)$ e $f_k(\bar{x}) \leq f_k(x^*)$ para $k \neq i$. Isto é exatamente a definição de eficiência para x^* . \square

O Teorema (3.10) é útil porque não é necessário se preocupar qual função-objetivo considerar no Problema Restrito. O resultado a seguir alerta sobre o caso de quando temos múltiplas soluções neste procedimento.

Teorema 3.11. *Se o Problema (3.9) para algum $i = 1, \dots, r$ com $\varepsilon_k = f_k(x^*)$ para uma dada solução $x^* \in \mathcal{X}$ tiver única solução, então x^* é eficiente.*

Demonstração. Considere x^* única solução do Problema Restrito. Assumimos que ela não seja eficiente. Isto implica que existe uma outra solução $\mathcal{X} \ni \bar{x} \neq x^*$ tal que $f_k(\bar{x}) \leq f_k(x^*)$ para todo $k = 1, \dots, r$ e para ao menos um índice i , vale $f_i(\bar{x}) < f_i(x^*)$. No entanto, x^* é única solução, logo ela deve ser factível, de sorte que $f_k(\bar{x}) \leq \varepsilon_k = f_k(x^*)$, $k \neq i$ e também $f_i(x^*) < f_i(\bar{x})$. Tem-se aqui duas desigualdades conflitantes, logo x^* deve ser eficiente. \square

Assim, para se ter certeza que uma solução produzida por este procedimento seja eficiente, é necessário garantir a unicidade da solução fornecida pelo Problema Restrito

ou resolver r Problemas Restritos (cada um com uma função-objetivo diferente). Em geral, é difícil checar se a solução do subproblema é única. Do ponto de vista computacional, múltiplas soluções podem ser evitadas se o Problema Restrito for ligeiramente modificado da seguinte maneira:

$$\begin{aligned} \text{Minimize } z^\varepsilon &= f_i(x) + \rho \cdot \sum_{\substack{k=1 \\ k \neq i}}^r f_k(x) \\ \text{sujeito a } x &\in \mathcal{X}, \\ f_k(x) &\leq \varepsilon_k, \quad k = 1, \dots, r, \quad k \neq i, \end{aligned} \tag{3.10}$$

onde $\rho > 0$. Esta soma adicional na função-objetivo visa eliminar múltiplas soluções e “forçar” a eficiência da solução encontrada. A Figura (17) traz uma interpretação geométrica desta situação.

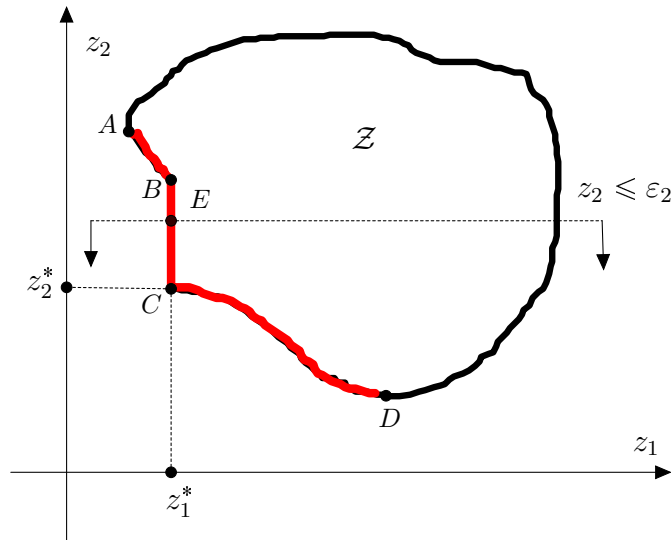


Figura 17 – Interpretação do Problema Restrito Modificado

Ao impormos a restrição $z_2 \leq \varepsilon_2$ há múltiplos valores de mínimo para z_1 . Todos os pontos do segmento vertical BC serão solução para o Problema Restrito original. No entanto, apenas o extremo inferior C é não dominado. Ao adicionarmos a parcela $\rho \cdot z_2$, a minimização do novo objetivo impedirá soluções fracamente dominadas.

Para finalizar os principais resultados desta técnica, enunciamos o teorema a seguir, que é uma versão mais geral do Teorema (3.11).

Teorema 3.12. *A única solução do Problema (3.9) é eficiente, para qualquer limite superior $\varepsilon = (\varepsilon_1, \dots, \varepsilon_{i-1}, \varepsilon_{i+1}, \dots, \varepsilon_r)^T$.*

Demonstração. Seja x^* a única solução do Problema (3.9). Isto significa que $f_i(x^*) < f_i(\bar{x})$ para todo $\bar{x} \in \mathcal{X}$. Assume-se que x^* não é eficiente, logo existe uma solução $\mathcal{X} \ni \bar{x} \neq x^*$

de maneira que $f_k(\bar{x}) \leq f_k(x^*)$, com $k = 1, \dots, r$ e $f_j(\bar{x}) < f_j(x^*)$ para ao menos um $j = \{1, \dots, r\}$.

- **Caso 1:** Se $j = i$, então $f_i(\bar{x}) < f_i(x^*)$ e $f_k(\bar{x}) \leq f_k(x^*) \leq \varepsilon_k$, $k \neq i$, o que é uma contradição, pois x^* resolve o Problema Restrito.
- **Caso 2:** Se $j \neq i$, então $f_j(\bar{x}) < f_j(x^*) \leq \varepsilon_j$, $f_k(\bar{x}) \leq f_k(x^*) \leq \varepsilon_k$, para $k \neq \{i, j\}$ e também $f_i(\bar{x}) \leq f_i(x^*)$. Esta última desigualdade é uma contradição, pois x^* é a solução única, e assim, x^* deve ser eficiente.

□

Um cuidado que devemos ter é impedir que $\varepsilon_k > z_k^N$ ou $\varepsilon_k < z_k^I$, pois o Problema Restrito ou será infactível ou a restrição $f_k(x) \leq \varepsilon_k$ será redundante.

3.4.1 Conexões da Soma Ponderada com o ε -Restrito

Embora pareçam técnicas bastante distintas, os métodos Soma Ponderada e ε -R possuem algumas relações e são apresentadas nos teoremas que seguem.

Teorema 3.13. *Seja $x^* \in \mathcal{X}$ uma solução para o problema ponderado com $w = (w_1, \dots, w_r) \geq 0$. Então:*

1. *Se $w_i > 0$, x^* é solução do Problema Restrito (3.9) utilizando $f_i(x)$ como função-objetivo e $\varepsilon_k = f_k(x^*)$, $k = 1, \dots, r$, $k \neq i$.*
2. *Se x^* é a única solução do problema ponderado, então x^* é uma solução do Problema Restrito com $\varepsilon_k = f_k(x^*)$, $k = 1, \dots, r$ e $k \neq i$, para toda a função-objetivo $f_i(x)$, $i = 1, \dots, r$.*

Demonstração. Seja x^* a solução do problema ponderado para algum vetor de pesos $w \geq 0$.

1. Seja $w_i > 0$. Então, para todo $x \in \mathcal{X}$ temos:

$$\sum_{k=1}^r w_k \cdot f_k(x) \geq \sum_{k=1}^r w_k \cdot f_k(x^*). \quad (3.11)$$

Agora, supomos que x^* não seja solução do Problema Restrito. Então existe um \bar{x} de modo que $f_i(\bar{x}) < f_i(x^*)$ e também $f_k(\bar{x}) \leq f_k(x^*) \leq \varepsilon_k$ para todo $k \neq i$. Agora, como $w_i > 0$, então:

$$0 > w_i \cdot (f_i(\bar{x}) - f_i(x^*)) - \sum_{\substack{k=1 \\ k \neq i}}^r w_k \cdot (f_k(\bar{x}) - f_k(x^*)) = \sum_{k=1}^r w_k \cdot (f_k(\bar{x}) - f_k(x^*)).$$

Esta última desigualdade é contraditória com relação à (3.11), logo x^* deve ser solução para o problema ponderado.

2. Se x^* é a única solução para o Problema Restrito, então:

$$\sum_{k=1}^r w_k \cdot f_k(x^*) < \sum_{k=1}^r w_k \cdot f_k(x),$$

para qualquer solução $x \in \mathcal{X}$. Suponha agora que existe algum i , para o qual o Problema Restrito com $f_i(x)$ como objetivo, não resulte na solução x^* . Nesse caso, deve existir um outro $\bar{x} \in \mathcal{X}$ de maneira que $f_i(\bar{x}) < f_i(x^*)$ e que $f_k(\bar{x}) \leq f_k(x^*) \leq \varepsilon_k$ para $k \neq i$. Logo, para qualquer $w \geq 0$, temos o seguinte:

$$\sum_{k=1}^r w_k \cdot f_k(\bar{x}) \leq \sum_{k=1}^r w_k \cdot f_k(x^*),$$

o que é uma contradição, pois x^* é solução única. Então, x^* deve ser solução para o Problema Restrito quando $f_i(x)$ é minimizada.

□

O teorema a seguir está relacionado para problemas convexos.

Teorema 3.14. *Seja um POM convexo. Se $x^* \in \mathcal{X}$ é uma solução para o Problema Restrito (3.9) para qualquer função-objetivo $f_i(x)$ e $\varepsilon_k = f_k(x^*)$, para $k = 1, \dots, r$ e $k \neq i$, então existe um vetor de pesos $w > 0$, com $w_k \in [0, 1]$, $k = 1, \dots, r$ e $\sum_{k=1}^r w_k = 1$, de modo que x^* é também solução para o problema ponderado.*

Demonstração. Não pertence ao escopo deste texto, e pode ser encontrada em (HAIMES; LASDON; WISMER, 1971). □

Nesse momento, temos condição para demonstrar o Teorema (3.3).

Demonstração. (Teorema 3.3). Como x^* é eficiente, pelo Teorema (3.10), então é solução para o Problema Restrito, seja qual for a função-objetivo $f_i(x)$ considerada. Pela hipótese de convexidade do POM e com a ajuda do Teorema (3.14), a demonstração é finalizada. □

3.5 O Método de Benson

Este método foi proposto inicialmente no trabalho proposto em (BENSON, 1978) e é uma técnica alternativa de escalarização que pode gerar soluções eficientes para um dado POM.

Dada uma solução inicial factível x^0 , consideramos o desvio positivo (no espaço objetivo) entre outra solução factível x , isto é, $l_k = f_k(x^0) - f_k(x) \geq 0$. A ideia deste procedimento, consiste em maximizar a soma de todos os r desvios em todos os objetivos, obtendo o seguinte problema escalar:

$$\begin{aligned} \text{Maximize} \quad & z_0 = \sum_{k=1}^r l_k \\ \text{sujeito a} \quad & x \in \mathcal{X}, \\ & l_k = f_k(x^0) - f_k(x), \quad k = 1, \dots, r, \\ & l_k \geq 0, \quad k = 1, \dots, r. \end{aligned} \tag{3.12}$$

A ilustração a seguir mostra a ideia da formulação anterior para o caso $r = 2$ objetivos. As variáveis l_1 e l_2 são desvios positivos entre os vetores objetivos associados à uma solução admissível x^0 e x . Dado x^0 , de vetor objetivo dado pelo ponto $A = f(x^0)$, a função-objetivo do Problema (3.12) visa buscar um ponto $B = f(x)$ no espaço de critério de modo que a soma $l_1 + l_2$ seja a maior possível.

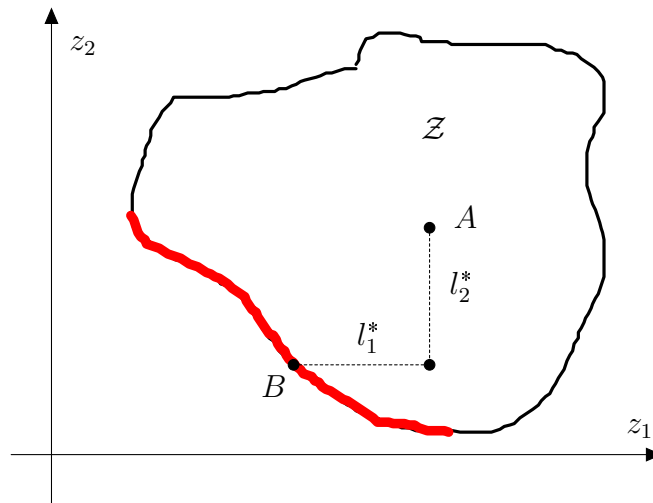


Figura 18 – Interpretação geométrica do método de Benson

Os teoremas a seguir apresentam algumas propriedades deste método.

Teorema 3.15. *A solução $x^0 \in \mathcal{X}$ é eficiente se, e somente se, $z_0^* = 0$.*

Demonstração. (\Rightarrow) Suponhamos que (x^*, l^*) seja solução do Problema (3.12) e que x^* seja eficiente. Então não há uma solução admissível x^0 , de modo que $f_k(x^0) \leq f_k(x^*)$ e que $f_k(x^0) < f_k(x^*)$. Em outras palavras, não existe um x^0 de maneira que $0 > -l^* = f(x^*) - f(x^0)$, isto é, $l^* = 0$.

(\Leftarrow) Seja (x^*, l^*) uma solução admissível para (3.12) e suponhamos que x^* não seja eficiente. Nesse caso, existe um $x^0 \in \mathcal{X}$, de modo que $f_k(x^0) \leq f_k(x^*)$ e $f_k(x^0) < f_k(x^*)$

para ao menos um $k \in \{1, \dots, r\}$. Isto significa que

$$\sum_{k=1}^r l_k = \sum_{k=1}^r (f_k(x^0) - f_k(x^*)) < 0,$$

contradizendo a hipótese. Logo x^* deve ser eficiente. \square

Teorema 3.16. *Se o Problema (3.12) tem solução ótima (x^*, l^*) com $z_0^* < \infty$, então x^* é eficiente.*

Demonstração. Suponha que x^* não seja eficiente. Logo, deve existir um $\mathcal{X} \ni \bar{x} \neq x^*$ de modo que $f_k(\bar{x}) \leq f_k(x^*)$, $k = 1, \dots, r$ e $f_k(\bar{x}) < f_k(x^*)$ para ao menos um $k = \{1, \dots, r\}$. Defina

$$\bar{l} = f(x^0) - f(\bar{x}).$$

Então (\bar{x}, \bar{l}) é factível para (3.12), pois

$$\bar{l} = f(x^0) - f(\bar{x}) > f(x^0) - f(x^*) = l^* \geq 0.$$

Agora,

$$\sum_{k=1}^r \bar{l}_k > \sum_{k=1}^r l_k^*,$$

pois $\bar{l}_k > l_k^*$. Mas isto é impossível, visto que (x^*, l^*) é solução ótima. Assim, x^* deve ser eficiente. \square

O método de Benson tem a vantagem, em relação à Soma Ponderada, de conseguir gerar soluções cuja imagem pertencem à região não convexa de \mathcal{Z}^* , desde que x^0 seja convenientemente escolhido. Por outro lado, duas soluções factíveis podem fornecer uma mesma solução eficiente. Isto faz com que a obtenção de soluções eficientes dependa fortemente das escolhas x^0 . Além disso, podemos ter uma distribuição de pontos não dominados muito irregular para um dado número de subproblemas com diferentes x^0 . Essa distribuição está ligada à curvatura da fronteira de Pareto do POM. No Capítulo 5 e Sub-seção 6, adaptamos uma maneira de obter todas as soluções eficientes para o PCUI em um número mínimo de subproblemas.

3.6 O Método de Silva & Crema ((SYLVA; CREMA, 2004))

O método proposto em (SYLVA; CREMA, 2004) é apresentado brevemente a seguir. Foi desenvolvido especificamente para tratar a seguinte problema,

$$\begin{aligned} \text{Minimize} \quad & z = Cx \\ \text{sujeito a} \quad & Ax = b, \\ & x \in \mathbb{N}^n, \end{aligned} \tag{3.13}$$

onde $C \in \mathbb{Z}^{r \times n}$, $A \in \mathbb{Z}^{m \times n}$ e $b \in \mathbb{Z}^m$.

Em seções anteriores, o Teorema (3.1) atestou que qualquer solução do problema ponderado resulta numa solução x^* eficiente. Além disso, essas soluções são suportadas. A grande desvantagem da técnica da Soma Ponderada está no fato de não gerar soluções não suportadas, que são muito comuns em problemas combinatórios multiobjetivos. O método apresentado nesta seção é uma técnica alternativa, bem diferente das abordagens apresentadas até este ponto e que também contorna essa desvantagem.

O teorema a seguir está demonstrado no artigo em (SYLVA; CREMA, 2004) e é a base do funcionamento desse método.

Teorema 3.17. *Sejam x^1, \dots, x^l soluções eficientes para o Problema (3.13). Seja $\mathcal{X}_s = \{x \in \mathbb{N}^n : Cx^s \leq Cx\}$, isto é, o conjunto das soluções que são dominadas por x^s . Seja x^* uma solução eficiente para o problema*

$$\begin{aligned} & \text{Minimize} && z = Cx \\ & \text{sujeito a} && x \in \mathcal{X} - \cup_{s=1}^l \mathcal{X}_s. \end{aligned} \tag{3.14}$$

Então

- (i) x^* é eficiente para o Problema (3.13).
- (ii) Além disso, se o Problema (3.14) é infactível, então $\cup_{s=1}^l \{Cx^s\}$ são todas as soluções não dominadas para o Problema (3.13).

Demonstração. (i) Suponhamos que existe um $\bar{x} \in \mathcal{X}$ tal que $C\bar{x} \leq Cx^*$ com ao menos uma desigualdade estrita. Como x^* é uma solução eficiente para (3.14), então \bar{x} não pode pertencer à região admissível do Problema (3.14), isto é, $\bar{x} \in \cup_{s=1}^l \mathcal{X}_s$. Isto implica que $\bar{x} \in \mathcal{X}_s$ para algum $s = \{1, \dots, l\}$. Pela definição de \mathcal{X}_s e $Cx^s \leq C\bar{x} \leq Cx^*$, temos que $x^* \in \mathcal{X}_s$, que é um absurdo, pois $x^* \in \mathcal{X} - \cup_{s=1}^l \mathcal{X}_s$.

(ii) Se o Problema (3.14) é infactível, então $\mathcal{X} \subseteq \cup_{s=1}^l \mathcal{X}_s$ e, para qualquer $x \in \mathcal{X}$, sempre existirá um x^s tal que $Cx^s \leq Cx$. Então devemos ter que $Cx = Cx^s$ (e daí $Cx \in \cup_{s=1}^l \mathcal{X}_s$) ou que $Cx^s \leq Cx$ com ao menos uma desigualdade estrita (nesse caso, Cx é dominado). □

Como consequência imediata do teorema anterior e do Teorema (3.1), temos o seguinte resultado:

Corolário 3.1. *Sejam x^1, \dots, x^l soluções eficientes para o Problema (3.13). Se x^* é uma solução ótima para o problema*

$$\begin{aligned} & \text{Minimize} && z = w^T Cx \\ & \text{sujeito a} && x \in \mathcal{X} - \cup_{s=1}^l \mathcal{X}_s, \end{aligned} \tag{3.15}$$

para $0 < w \in \mathbb{R}^r$, então x^* é uma solução eficiente para o Problema (3.13).

Assim, o método inicia resolvendo um problema ponderado com algum vetor peso $w > 0$ previamente especificado. Supondo que este problema ponderado é factível, então uma primeira solução eficiente x^1 foi encontrada. O próximo passo consiste em eliminar do conjunto \mathcal{X} todos os pontos que são dominados por x^1 . Uma maneira de isso ser feito consiste em inserir as restrições lineares

$$\begin{aligned} (Cx)_k &\leq ((Cx^1)_k - 1) \cdot t_k + M_k \cdot (1 - t_k), \quad k = 1, \dots, r, \\ \sum_{k=1}^r t_k &\geq 1, \\ t_k &\in \mathbb{B}, \quad k = 1, \dots, r, \end{aligned}$$

onde M_k é um limitante superior para o k -ésimo objetivo. Se, por exemplo, $t_k = 0$, então $(Cx)_k \leq M_k$ que é redundante. Se $t_k = 1$, então $(Cx)_k \leq Cx^1 - 1$, isto é, no k -ésimo objetivo a nova solução encontrada terá um valor objetivo inferior à $(Cx^1)_k$. Estas restrições geram um novo problema, mais restrito que o original, que deverá ser resolvido:

$$\begin{aligned} \text{Minimize} \quad & z = w^T Cx \\ \text{sujeito a} \quad & Ax = b, \\ & (Cx)_k \leq ((Cx^1)_k - 1) \cdot t_k + M_k \cdot (1 - t_k), \quad k = 1, \dots, r, \\ & \sum_{k=1}^r t_k \geq 1, \\ & x \in \mathbb{N}, \quad t_k \in \mathbb{B}, \quad k = 1, \dots, r. \end{aligned}$$

Mais genericamente, se as soluções x^1, \dots, x^l são eficientes e já são conhecidas previamente, o novo problema a ser construído será dado por:

$$\begin{aligned} \text{Minimize} \quad & z = w^T Cx \\ \text{sujeito a} \quad & Ax = b, \\ & (Cx)_k \leq ((Cx^s)_k - 1) \cdot t_k + M_k \cdot (1 - t_k), \quad k = 1, \dots, r, \quad s = 1, \dots, l, \\ & \sum_{k=1}^r t_k^s \geq 1, \quad s = 1, \dots, l, \\ & x \in \mathbb{N}^n, \quad t_k \in \mathbb{B}, \quad k = 1, \dots, r, \quad s = 1, \dots, l. \end{aligned} \tag{3.16}$$

Os dois conjuntos de restrições adicionais garantem a exclusão do conjunto \mathcal{X}_s do conjunto factível original \mathcal{X} . O procedimento é executado até que o Problema (3.16) fique infactível. Nesse caso todas as soluções eficientes para o problema original foram detectadas.

3.7 Métricas de Comparação

Para avaliar e comparar os métodos de escalarização desenvolvidos, aplicamos algumas métricas que comparam as soluções eficientes obtidas tomando por base as

soluções não dominadas. Embora as metodologias consideradas são determinísticas, isto é, potencialmente todas deveriam fornecer exatamente as mesmas soluções eficientes, pode acontecer que algumas soluções sejam perdidas (como por exemplo na Soma Ponderada e no Método de Tchebycheff) ou serem distintas umas das outras influenciadas pela da técnica de escalarização a ser adotada, tempo computacional máximo permitido para resolver cada subproblema, etc. No Capítulo 6 propomos duas maneiras distintas de se resolver o PCUIM, que podem gerar fronteiras de Pareto diferentes para o mesmo problema. Estas métricas visam quantificar tais disparidades e foram apresentadas na referência descrita em (DEB; KALYANMOY, 2001).

A seguir, apresentamos as métricas avaliativas que foram utilizadas neste trabalho e comparam dois conjuntos não dominados \mathcal{Z}_1 e \mathcal{Z}_2 no espaço de critério.

- **Métrica 1 - distanciamento médio** (σ^1): para cada ponto $z^j \in \mathcal{Z}_1$ esta métrica calcula a distância euclidiana entre z^j e o ponto de \mathcal{Z}_2 mais próximo. Em seguida fazemos a média de ϕ_j . Formalmente,

$$\sigma^1 = \frac{\sum_{j=1}^{|\mathcal{Z}_1|} \phi_j}{|\mathcal{Z}_1|},$$

onde

$$\phi_j = \min_{1 \leq i \leq |\mathcal{Z}_2|} \left\{ \sqrt{\sum_{k=1}^r (z_k^i - z_k^j)^2} \right\}, \quad (3.17)$$

onde z_k^j é o valor objetivo do j -ésimo ponto na k -ésima função-objetivo. Quanto menor σ_1 , mais próximos os pontos de \mathcal{Z}_1 estão de \mathcal{Z}_2 . Esta métrica pode ser mais útil quando temos a fronteira ótima de Pareto \mathcal{Z}^* e queremos comparar com uma fronteira \mathcal{Z}_1 obtida de maneira não exata. Para isso é só adotar $\mathcal{Z}_2 = \mathcal{Z}^*$.

- **Métrica 2 - distanciamento pontual** (σ^2): Esta métrica visa calcular a proporção dos pontos de \mathcal{Z}_1 que distam no mínimo $\delta \geq 0$ de \mathcal{Z}_2 . Para todo z^j de \mathcal{Z}_1 , a expressão (3.17) fornece a distância dele à \mathcal{Z}_2 . Calculamos

$$\sigma^2 = \frac{\sum_{j=1}^{|\mathcal{Z}_1|} e_j}{|\mathcal{Z}_1|}$$

onde

$$e_j = \begin{cases} 1, & \text{se } \phi_j > \delta \\ 0, & \text{caso contrário.} \end{cases}$$

Neste trabalho, adotamos $\delta = 0$ para esta métrica. O resultado deste cálculo fornece a proporção dos pontos de \mathcal{Z}_1 que coincidem com os pontos de \mathcal{Z}_2 . Quanto menor for σ^2 , mais pontos coincidentes existem entre estas fronteiras.

- **Métrica 3 - amplitude dos objetivos** (σ^3): calcula as amplitudes dos pontos lexicográficos. Se z^+ e z^- são os *lexicográficos* da fronteira \mathcal{Z} , então

$$\sigma_k^3 = \|z_k^+ - z_k^-\|_1, \quad k = 1, \dots, r.$$

- **Métrica 4 - hipervolume** (σ^4): esta métrica calcula a soma dos hipervolumes dos hipercubos cuja a diagonal é unida pelos pontos $z^j \in \mathcal{Z}$ e $u \in \mathbb{R}^r$ cuja coordenada k é dada por $\max z_k$, $k = 1, \dots, r$. Para o caso bidimensional assumimos que os pontos $z^1, \dots, z^{|\mathcal{Z}|}$ estejam ordenados em ordem crescente pelo valor da primeira função-objetivo. Esta métrica calcula a soma das áreas dos retângulos como na Figura (19), isto é:

$$\sigma^4 = (u_1 - z_1^1) \cdot (u_2 - z_2^1) + \sum_{j=2}^{|\mathcal{Z}|} (z_1^{j+1} - z_1^j) \cdot (z_2^j - z_2^{j+1}).$$

Quanto maior for o valor de σ^4 associado, mais inferior é a fronteira de Pareto.

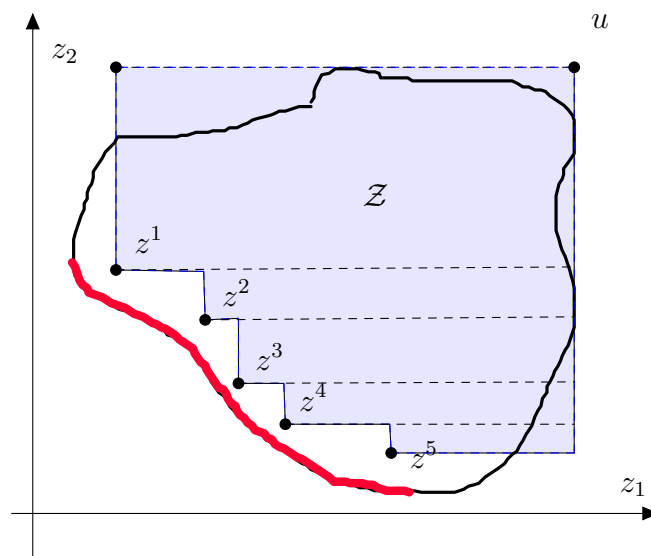


Figura 19 – Área determinada pelas soluções não dominadas

- **Métrica 5 - número de soluções não dominadas** (σ^5): determina o número de soluções não dominadas obtidas, isto é, $\sigma^5 = |\mathcal{Z}|$.

Este capítulo apresentou os principais métodos de escalarização clássicos da literatura para resolver um POM e as condições matemáticas para termos garantia de soluções eficientes com estas técnicas. Apresentamos alguns métodos utilizados para comparar duas fronteiras de Pareto, que serão fundamentais para o estudo realizado no Capítulo 6. O próximo capítulo apresenta algumas abordagens heurísticas para resolver o PCUI dando especial importância a um método original, capaz de produzir soluções inteiras para nosso problema, que por sua vez, será adaptado para o PCUIM.

Parte II

Desenvolvimento da pesquisa

Capítulo 4

Uma nova heurística de arredondamento para o PCUI

Neste capítulo, tratamos especificamente o PCUI e métodos heurísticos de arredondamento. Como destacamos no Capítulo 2, a solução exata deste problema é praticamente impossível de ser obtida para exemplares práticos. Isso motivou durante muitos anos o desenvolvimento de métodos heurísticos para gerar soluções aproximadas para este problema. Além dos trabalhos pioneiros apresentados em (GILMORE; GOMORY, 1961) e (GILMORE; GOMORY, 1963), outros artigos como em (BELOV; SCHEITHAUER, 2002), (FARLEY; RICHARDSON, 1984), (FOERSTER; WASCHER, 2000), (GOLFETO; MORETTI; SALLES, 2008), (HAESSLER, 1975), (POLDI; ARENALES, 2006), (POLDI; ARENALES, 2009), (WASCHER; GAU, 1996), utilizam diferentes abordagens aproximativas. A ideia chave destes procedimentos é produzir soluções admissíveis de boa qualidade num tempo computacional aceitável.

Neste capítulo, não é nossa intenção fornecer um estado-da-arte de todas as abordagens heurísticas presentes na literatura, e sim, apresentar as principais ideias de procedimentos mais utilizados e que possuem melhores performances e robustez neste problema. Em seguida, damos especial ênfase no método que estamos propondo, completamente novo e que será empregado e adaptado para o PCUIM.

Inicialmente, descrevemos brevemente o método Simplex que servirá de base para o clássico método Geração de Colunas (GC) proposto em (GILMORE; GOMORY, 1961) específico para o problema de corte, que terá importância fundamental na obtenção de “bons” padrões de cortes para o PCUIM. Em seguida, apresentamos algumas heurísticas de arredondamento, que buscam obter uma solução inteira para o PCUI baseada em sua relaxação linear. Finalmente, apresentamos, descrevemos e exemplificamos nossa heurística que gera uma solução inteira para o problema investigando nas direções simplex do ótimo do problema relaxado. Esta heurística será fundamental para que possamos gerar soluções eficientes para o PCUIM de maneira mais rápida.

4.1 Revisão de métodos heurísticos para o PCUI

4.1.1 O método de Geração de Colunas

Este método foi pioneiramente apresentado no trabalho apresentado em (GILMORE; GOMORY, 1961) em 1961 e depois aprimorado dois anos mais tarde em (GILMORE; GOMORY, 1963). A grande vantagem deste método é o fato de não ser necessário gerar todos os μ padrões de corte para resolver a relaxação linear do Problema (2.3), isto é, o Problema de Programação Linear:

$$\begin{aligned} \text{Minimize } & z = \|x\|_1 \\ \text{sujeito a } & Ax \geq d, \\ & x \geq 0, \end{aligned} \tag{4.1}$$

cujas colunas de $A \in \mathbb{N}^{m \times \mu}$ contém todos os padrões de corte possíveis. Em muitos casos, $\mu \gg m$, tornando inviável a aplicação direta do método Simplex nesta formulação. No entanto, da álgebra do método simplex, se a base ótima não for degenerada, a solução ótima relaxada, x^* , é um vetor com exatamente m componentes não nulas e as $\mu - m$ remanescentes serão nulas. Isso sugere a ideia de apenas considerar m padrões de cortes (colunas) e estabelecer um critério de identificação se estas m colunas são ótimas para a relaxação.

Antes de apresentar o método GC propriamente dito, fazemos uma breve revisão do método Primal Simplex e inserimos a notação empregada para futuros desenvolvimentos.

Considere o Problema de Programação Linear na forma canônica:

$$\begin{aligned} \text{Minimize } & z = c^T x \\ \text{sujeito a } & Ax = b, \\ & x \geq 0, \end{aligned}$$

onde $x \in \mathbb{R}^n$, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times m}$ e $b \in \mathbb{R}^m$ e $\text{posto}(A) = m$. O principal resultado dentro da teoria de Problema de Programação Linear, demonstrado em (BAZARAA; JARVIS; SHERALI, 2004), afirma o seguinte:

Teorema 4.1 (Teorema da Programação Linear). *Em um Problema de Programação Linear factível de n variáveis e m restrições, sua solução ótima x^* é um vetor com no máximo m variáveis não nulas.*

Baseado nesse teorema, uma solução ótima deve ser procurada anulando as $n - m$ variáveis e considerando as m remanescentes não nulas. Isto gera um total de C_n^{n-m} soluções a serem procuradas. Obviamente, este procedimento é inviável computacionalmente, pois o número de combinações aumenta exponencialmente com n . Contudo, o método simplex contorna este inconveniente, utilizando a estratégia Simplex descrita a seguir.

Considere as seguintes partições:

- $A = [B \ N]$, onde $B \in \mathbb{R}^{m \times m}$ é a matriz com as variáveis básica (não singular) e $N \in \mathbb{R}^{m \times n-m}$ a matriz com as variáveis não básicas.
- $x = [x_B \ x_N]$, onde x_B denota as variáveis básicas e x_N as não básicas. Denotamos por \mathcal{B} o conjunto dos índices das variáveis básicas e \mathcal{N} o conjunto dos índices das variáveis não básicas.
- $c = [c_B \ c_N]$, em que c_B e c_N denotam os custos básicos e não básicos, respectivamente.

Assim sendo, temos que:

$$Ax = b \Leftrightarrow [B \ N] \begin{bmatrix} x_B \\ x_N \end{bmatrix} = b \Leftrightarrow x_B = B^{-1}b - B^{-1}Nx_N.$$

Logo a solução básica primal é $x^0 = \begin{bmatrix} x_B^0 \\ x_N^0 \end{bmatrix}$ com $x_B^0 = B^{-1}b$ e $x_N^0 = 0$. Se $x_B^0 \geq 0$, então temos uma solução primal factível.

Substituindo x_B na função-objetivo, obtemos o seguinte:

$$z = c^T x = [c_B \ c_N]^T \begin{bmatrix} x_B \\ x_N \end{bmatrix} = c_B^T x_B + c_N^T x_N.$$

Agora, notamos que:

$$\begin{aligned} z &= c_B^T (B^{-1}x_B - B^{-1}Nx_N) + c_N^T x_N \\ &= (c_B^T B^{-1})b + (c_N^T - c_B^T B^{-1}N)x_N \\ &= \pi^T b + z_N^T x_N, \end{aligned}$$

onde $\pi^T = c_B^T B^{-1}$ é denominado de *multiplicador simplex* relativo à base \mathcal{B} e $z_N^T = c_N^T - c_B^T B^{-1}N$ é chamado custo relativo das variáveis não básicas. Notamos que se $z_j < 0$, para algum $j \in \mathcal{N}$, podemos inserir a variável não básica correspondente na base e o valor de z decrescerá. Alguma variável básica com valor positivo será levada no valor 0 e deverá dar lugar à esta variável de entrada, mantendo a factibilidade do problema ($x_B^0 \geq 0$). Caso contrário, se $z_N \geq 0$, temos uma solução ótima. As estratégias descritas anteriormente compõe o que se denomina *Estratégia Simplex*.

Mais formalmente, os passos desta estratégia são resumidos a seguir:

- Escolhemos uma variável de índice $k \in \mathcal{N}$ tal que $z_k < 0$.
- Façamos $\begin{cases} x_k = \alpha > 0 \\ x_j = 0, & j \in \mathcal{N} - \{k\}. \end{cases}$

- Nova solução: $x_N = \alpha e_k$, $e_k = (0, \dots, 1, \dots, 0)$, o k -ésimo vetor canônico de dimensão n .
- Atualização:

$$\begin{aligned} x_B &= B^{-1}b - B^{-1}Nx_N \\ &= B^{-1}b - B^{-1}N\alpha e_k \\ &= B^{-1}b - \alpha B^{-1}Ne_k \\ &= x_B^0 - \alpha B^{-1}a^k \\ &= x_B^0 - \alpha \delta \geq 0, \end{aligned}$$

onde $\delta = B^{-1}a^k$ é denominado *bloqueio*, a^k é a k -ésima coluna de N . Logo,

$$\alpha = \min_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\},$$

para manter a factibilidade primal. O cálculo anterior denomina *teste da razão*.

- Definimos

$$s = \arg \min_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\},$$

o índice da variável que deixa a base.

- Se $\delta < 0$ então a solução é ilimitada.
- Atualize a base atual, isto é, $\mathcal{B} = (\mathcal{B} \cup \{k\}) - \{s\}$.

O processo anterior se repete até que $z_j \geq 0$ para todo $j \in \mathcal{N}$. A cada iteração, uma nova base mais promissora é detectada.

O método primal simplex descrito anteriormente é esquematizado no Algoritmo (1).

Tendo por base o funcionamento do método simplex, podemos apresentar o método GC para resolver o problema relaxado (4.1), que é a forma compacta do PCUI com $b = 1$. Por simplicidade, consideramos que as restrições de demanda estejam na igualdade. Em um PCUI normalmente $\mu \gg m$, então supomos que todas as colunas de A não estejam disponíveis.

O método GC parte de uma base inicial formada por uma matriz B utilizando os m padrões de corte homogêneos, isto é:

$$B = \begin{bmatrix} \lfloor L/\ell_1 \rfloor & 0 & \cdots & 0 \\ 0 & \lfloor L/\ell_2 \rfloor & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lfloor L/\ell_m \rfloor \end{bmatrix}.$$

Algoritmo 1: Método primal simplex

- 1: Entrada: A , b e c
- 2: Considere B uma matriz básica inicial factível
- 3: Calcule $x_B^0 = B^{-1}b$
- 4: Calcule $\pi^T = c_B^T B^{-1}$
- 5: Calcule $z_N^T = c_N^T - \pi^T N$, o vetor dos custos reduzidos
- 6: Calcule $k = \operatorname{argmin}\{z_N : z_N < 0\}$ (critério de Dantzig)
- 7: **while** $z_{N_k} < 0$ **do**
- 8: Calcule o bloqueio $\delta^k = B^{-1}a^k$
- 9: **if** $\delta^k < 0$ **then**
- 10: *Pare, solução ilimitada*
- 11: **else**
- 12: Faça o teste da razão para obter α :

$$\alpha = \min_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\}$$

e obtenha a variável s a sair da base atual:

$$s = \operatorname{arg min}_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\}.$$

- 13: Atualize a base $\mathcal{B} = (\mathcal{B} \cup \{k\}) - \{s\}$ e $\mathcal{N} = (\mathcal{N} \cup \{s\}) - \{k\}$
- 14: **end if**
- 15: **end while**
- 16: Saída \mathcal{B} .

A partir desta base inicial, calculamos a solução primal correspondente, resolvendo o sistema linear $Bx = d$ e também o vetor multiplicador simplex pelo sistema $B^T \pi = \mathbf{1}$. O próximo passo consiste em determinar se a base atual é ou não ótima para o problema, determinando o vetor dos custos reduzidos, isto é, $z_N = c_N^T - \pi^T N$ e analisando a sua menor componente (critério de Dantzig).

No método primal simplex, é necessário ter, a priori, todas as colunas da matriz A explicitamente. A grande vantagem deste método é que ele dispensa esta exigência, associando um subproblema, denominado *price*, que computa a *menor* componente do vetor z_N . Noutras palavras, o subproblema *price* pesquisa no espaço \mathbb{N}^m , alguma coluna a_j (que é um padrão de corte factível que satisfaz as condições (2.1)) de tal maneira que o vetor $z_N = c_N^T - \pi^T a_j = 1 - \pi^T a_j < 0$.

O subproblema *price* é então definido da seguinte forma:

$$\begin{aligned}
 & \text{Minimize} && \phi = 1 - \pi^T a_j \\
 & \text{sujeito a} && \sum_{i=1}^m \ell_i \cdot a_{ij} \leq L, \\
 & && \sum_{i=1}^m \ell_i \cdot a_{ij} \geq L - \eta, \\
 & && \sum_{i=1}^m a_{ij} \leq q, \\
 & && a_{ij} \in \mathbb{N}, \quad i \in I.
 \end{aligned} \tag{4.2}$$

Se o valor ótimo do problema anterior é tal que $\phi^* < 0$, então a solução associada a_j^* é um padrão factível que deve ser inserido na matriz básica B atual. Isto é feito da mesma forma como no método primal simplex, determinando o bloqueio δ e realizando o teste da razão para calcular α e a coluna s a ser substituída. Este procedimento é repetido até que o valor objetivo do subproblema seja positivo, isto é, detectamos que o *menor* custo reduzido positivo, logo $z_N > 0$ e então, a base atual é ótima para a relaxação do PCUI.

As m colunas que este procedimento fornece são denominadas *padrões de Gilmore-Gomory* ou *Básicos*. Determinados esses padrões, o próximo passo consiste em obter o vetor inteiro das frequências x de norma-1 mínima que satisfaça a demanda. Nas próximas subseções, apresentamos algumas técnicas para tal obtenção. Com especial importância, apresentamos nosso próprio método de arredondamento, descrito na última seção a parte.

A seguir, organizamos em um pseudocódigo o Método de GC para PCUI.

Algoritmo 2: Método Geração de Colunas de Gilmore e Gomory

- 1: Entrada: dados para o PCUI
- 2: Considere B a matriz básica inicial com os padrões de cortes homogêneos
- 3: Seja $\phi^* = -1$
- 4: **while** $\phi^* < 0$ **do**
- 5: Calcule $x_B^0 = B^{-1}d$
- 6: Calcule $\pi^T = \mathbf{1}^T B^{-1}$
- 7: Resolva o subproblema *price* (4.2) e atualize ϕ^*
- 8: **if** $\phi^* < 0$ **then**
- 9: Calcule o bloqueio $\delta = B^{-1}a_j^*$
- 10: **if** $\delta^k < 0$ **then**
- 11: *Pare, solução ilimitada*
- 12: **else**
- 13: Faça o teste da razão para obter α :

$$\alpha = \min_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\},$$

e obtenha a coluna s de B a sair da base atual:

$$s = \arg \min_{i \in \mathcal{B}} \left\{ \frac{x_{Bi}^0}{\delta_i} : \delta_i > 0 \right\}$$

- 14: Seja B^s esta coluna. Então atualize a matriz básica $B = (B \cup \{a_j^*\}) - \{B^s\}$
 - 15: **end if**
 - 16: **end if**
 - 17: **end while**
 - 18: Saída B .
-

Exemplo 4.1. Consideramos uma instância para o PCUI, como $b = 1$, $L = 11$, $\ell = (3, 4, 5)^T$, $q = 20$ e $d = (1.000, 1.500, 2.000)^T$. Com esses dados, existem $n = 8$ padrões de cortes possíveis, dos quais apenas três serão utilizados na solução ótima da relaxação.

Inicialmente

$$B = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

- **Iteração 1:** Calcule $x_B = B^{-1}d = (333.3, 750, 1000)^T$ e calculamos o multiplicador

simplex $\pi^T = \mathbf{1}^T B^{-1} = (1/3, 1/2, 1/2)^T$. Resolvemos o subproblema price:

$$\begin{aligned} \text{Minimize } \phi &= 1 - \frac{1}{3}a_{1j} - \frac{1}{2}a_{2j} - \frac{1}{2}a_{3j} \\ \text{sujeito a } 3a_{1j} + 4a_{2j} + 5a_{3j} &\leq 11, \\ 3a_{1j} + 4a_{2j} + 5a_{3j} &\geq 8, \\ a_{1j} + a_{2j} + a_{3j} &\leq 20, \\ a_{ij} \in \mathbb{N}, \quad i &= \{1, 2, 3\}, \end{aligned}$$

cuja solução é $a_j^* = (1, 2, 0)^T$ e $1 - \phi^* = -0,3 < 0$. Logo a coluna obtida é promissora de entrar na base atual.

Calculamos o vetor bloqueio $\delta = B^{-1}a_j^* = (1/3, 1, 0)^T$ e o passo mínimo $\alpha = \min\{x_{B_i}/\delta_i : \delta_i > 0, i = \{1, 2, 3\}\} = \min\{1000, 750\} = 750$ e $s = 2$. Atualizamos a base trocando a 2ª coluna de B por a_j^* :

$$B = \begin{bmatrix} 3 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

- **Iteração 2:** Calculamos $x_B = B^{-1}d = (83.3, 750, 1000)^T$ e o multiplicador simplex $\pi^T = \mathbf{1}^T B^{-1} = (1/3, 1/3, 1/2)^T$. Resolvemos o subproblema price:

$$\begin{aligned} \text{Minimize } \phi &= 1 - \frac{1}{3}a_{1j} - \frac{1}{3}a_{2j} - \frac{1}{2}a_{3j} \\ \text{sujeito a } 3a_{1j} + 4a_{2j} + 5a_{3j} &\leq 11, \\ 3a_{1j} + 4a_{2j} + 5a_{3j} &\geq 8, \\ a_{1j} + a_{2j} + a_{3j} &\leq 20, \\ a_{ij} \in \mathbb{N}, \quad i &= \{1, 2, 3\}, \end{aligned}$$

cuja solução é $a_j^* = (2, 0, 1)^T$ e $1 - \phi^* = -0,06 < 0$. Logo a coluna obtida é também promissora para entrar na base atual.

Calculamos o vetor bloqueio $\delta = B^{-1}a_j^* = (2/3, 0, 1/2)^T$ e o passo mínimo $\alpha = \min\{x_{B_i}/\delta_i : \delta_i > 0, i = \{1, 2, 3\}\} = \min\{125, 2000\} = 125$ e $s = 1$. Atualizamos a base trocando a 1ª coluna de B por a_j^* :

$$B = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 2 & 0 \\ 1 & 0 & 2 \end{bmatrix}.$$

- **Iteração 3:** Calculamos $x_B = B^{-1}d = (125, 750, 937.5)^T$ e o multiplicador simplex

$\pi^T = \mathbf{1}^T B^{-1} = (1/4, 3/8, 1/2)^T$. Resolvemos o subproblema price:

$$\begin{aligned} \text{Minimize } \phi &= 1 - \frac{1}{4}a_{1j} - \frac{3}{8}a_{2j} - \frac{1}{2}a_{3j} \\ \text{sujeito a } 3a_{1j} + 4a_{2j} + 5a_{3j} &\leq 11, \\ 3a_{1j} + 4a_{2j} + 5a_{3j} &\geq 8, \\ a_{1j} + a_{2j} + a_{3j} &\leq 20, \\ a_{ij} \in \mathbb{N}, \quad i &= \{1, 2, 3\}, \end{aligned}$$

cuja solução é $a_j^* = (0, 0, 2)^T$ e $1 - \phi^* = 0$. Logo a coluna obtida não é promissora de entrar na base e o critério de parada para o GC foi ativado.

Logo, a solução relaxada \tilde{x} para este exemplo é

- $\tilde{x}_1 = 125$ vezes o padrão $a_1^* = (2, 0, 1)^T$;
- $\tilde{x}_2 = 750$ vezes o padrão $a_2^* = (1, 2, 0)^T$;
- $\tilde{x}_3 = 937.5$ vezes o padrão $a_3^* = (0, 0, 2)^T$.

Seja J_B o conjunto dos índices dos m padrões básicos gerados pelo método GC e $A^* = [a_1^* | \dots | a_m^*]$ a matriz que os contém. Assumimos que a solução para a relaxação do PCUI seja o vetor $\tilde{x} = (\tilde{x}_1, \dots, \tilde{x}_m)^T \in \mathbb{R}^m$ e que possui pelo menos uma componente fracionária.

Algumas heurísticas de arredondamento para Problema de Programação Linear Inteiro gerais podem ser encontradas no trabalho em (GLOVER, 1977). O grupo de heurísticas apresentados a seguir são as mais utilizadas na literatura e tem em comum o fato de utilizarem os padrões básicos para gerar uma solução inteira aproximada \bar{x} para o PCUI e são amplamente discutidas e comparadas em (WASCHER; GAU, 1996). A seguir iremos dar uma breve revisão dessas técnicas.

4.1.2 Heurística de arredondamento simples e melhorado

A primeira heurística de arredondamento é a mais simples forma de se obter uma solução inteira \bar{x} para o PCUI. O procedimento consiste em arredondar \tilde{x} para o menor inteiro superior, isto é, $\bar{x} \leftarrow \lceil \tilde{x} \rceil$. Note que

$$\sum_{j \in J_B} a_{ij}^* \cdot \bar{x}_j \geq \sum_{j \in J_B} a_{ij}^* \cdot \tilde{x}_j \geq d_i, \quad i \in I,$$

pois $\bar{x} \geq \tilde{x}$. Isto significa que o simples arredondamento preserva a factibilidade do problema.

Como principal vantagem desta técnica, é de fácil implementação e baixo custo computacional. Como desvantagem, há elevado excesso de produção e valor objetivo,

tornando esta solução bastante distante do valor ótimo em muitos exemplares. Contudo, o erro máximo na função-objetivo que pode ser cometido com esta aproximação é majorado por m .

Uma segunda versão, um pouco mais melhorada, consiste em fazer reavaliações sucessivas nas soluções resultantes com a intenção de melhorar a solução inteira. Primeiramente, o vetor \tilde{x} é arredondado para o menor inteiro superior, gerando a solução \bar{x} . Em seguida, cada componente k de \bar{x} é reduzida de uma unidade e a restrição de demanda

$$a_{ik} \cdot (\bar{x}_k - 1) + \sum_{\substack{j \in J_B \\ j \neq k}} a_{ij} \cdot \bar{x}_j \geq d_i, \quad i \in I,$$

é verificada. Se for atendida esta redução é permitida, caso contrário, não. O método termina quando todas as componentes de \bar{x} forem testadas.

Uma terceira versão melhorada consiste repetir novamente o procedimento da segunda versão do método descrito no parágrafo anterior, na intenção de reduzir ainda mais o valor de cada componente de \tilde{x} . O método termina quando nenhuma componente puder ser diminuída.

4.1.3 Heurística de um método exato nos padrões básicos

Esta metodologia também possui uma ideia muito simples. Dado o conjunto J_B e a matriz A^* que define p padrões de corte, consideramos o PLI:

$$\begin{aligned} \text{Minimize} \quad & z = \|x\|_1 \\ \text{sujeito a} \quad & A^*x \geq d, \\ & x \in \mathbb{N}^p, \end{aligned} \tag{4.3}$$

que uma vez otimizado por algum método exato, como o *Branch-and-Bound*, produz uma solução inteira. Note que este problema, para p da ordem de dezenas, pode ser resolvido otimamente com certa rapidez pelo CPLEX. Além disso, outra vantagem desta técnica, é uma solução inteira de “boa” qualidade. Essa afirmação é sustentada pelo seguinte fato: a relaxação linear para o problema (4.3) é forte, de acordo com (WASCHER; GAU, 1996). Neste caso, os padrões A^* são boas colunas para o PCUI original.

Exemplo 4.2. *Considere um exemplo para o PCUI com $b = 1$, $L = 100$, $m = p = 4$, $\ell = (3, 6, 9, 12)^T$ e $q = 20$. Se gerarmos todas as combinações de corte, teremos $\mu = 378$. Ao resolvermos o PLI utilizando todos esses padrões, temos como solução ótima a seguinte:*

- $x_1^* = 3$ vezes o padrão $a_1 = (33, 0, 0, 0)^T$;
- $x_2^* = 13$ vezes o padrão $a_2 = (0, 9, 5, 0)^T$;
- $x_3^* = 1$ vez o padrão $a_3 = (3, 0, 10, 0)^T$;

- $x_4^* = 20$ vezes o padrão $a_4 = (0, 0, 3, 6)^T$;
- $x_5^* = 6$ vezes o padrão $a_5 = (0, 1, 1, 7)^T$,

cujos valores ótimos são $z^* = 43$, um *setup* de 5. Cada padrão tem uma perda de 1, logo a perda total é igual à 43. Além disso, esta solução gera um excesso de produção relativo de 1,53%.

Aplicando o método de GC neste exemplo, temos como solução ótima relaxada:

- $\tilde{x}_1 = 2.476$ vezes o padrão $a_1^* = (15, 0, 6, 0)^T$;
- $\tilde{x}_2 = 17.142$ vezes o padrão $a_2^* = (3, 7, 0, 4)^T$;
- $\tilde{x}_3 = 11.376$ vezes o padrão $a_3^* = (0, 0, 11, 0)^T$;
- $\tilde{x}_4 = 11.428$ vezes o padrão $a_4^* = (1, 0, 0, 8)^T$;

com um valor ótimo igual à $\tilde{z}^* = 42.42$ e um *setup* de 4. Uma solução inteira poderia ser obtida aproveitando-se a relaxação ao arredondar as componentes de \tilde{x} para o menor inteiro superior, resultando numa solução inteira $\bar{x} = (3, 18, 12, 12)^T$, cujo valor objetivo é 45.

A técnica desta seção visa considerar os padrões $A^* = [a_1^* | a_2^* | a_3^* | a_4^*]^T$ e resolver o PLI resultante. A solução ótima deste problema (ótima com respeito à estas colunas, não a do PCUI inteiro original) é o vetor $\bar{x} = (3, 18, 12, 11)^T$ com valor objetivo igual à $\bar{z}^* = 44$. A perda em cada padrão é também 1, resultando em uma perda total de 44 e ocasionando um excesso de produção relativo de 5%.

Este simples exemplo comprova que as colunas de Gilmore-Gomory são boas aproximações para o ótimo original do PCUI e considerar o PLI (4.3) com estes padrões poder ser uma boa ideia na obtenção de soluções inteiras de boa qualidade para o problema original.

Uma forma de levar em conta o *setup* no PCUI usando esta abordagem, consiste em considerar o problema de corte modificado, o qual fornecerá uma solução inteira utilizando no máximo ε padrões de corte.

$$\begin{aligned}
 \text{Minimize} \quad & z = \|x\|_1 \\
 \text{sujeito a} \quad & A^* x \geq d, \\
 & x \leq N \cdot y, \\
 & x \geq y, \\
 & \|y\|_1 \leq \varepsilon, \\
 & y \in \mathbb{B}^p, x \in \mathbb{N}^p.
 \end{aligned} \tag{4.4}$$

Comparado com o (4.3), este PLI tem exatamente o dobro do número de variáveis ($2 \cdot p$). No entanto, para p da ordem das dezenas, ainda é viável a aplicação do *Branch-and-Bound* para sua resolução.

As Heurísticas Residuais são procedimentos também aproximativos que visam obter uma solução inteira para o PCUI. Trabalhos como em (POLDI; ARENALES, 2006), (POLDI; ARENALES, 2009) e (WASCHER; GAU, 1996) usaram extensivamente esse método com tais técnicas e mostraram ser poderosas metodologias para obter uma solução inteira de ótima qualidade para o PCUI. Para mais descrições desta abordagem, recomendamos os trabalhos que citamos neste parágrafo.

Na próxima seção, apresentamos um método inédito e heurístico para encontrar soluções inteiras para o PCUI.

4.2 Heurística de Arredondamento por Direções Simplex

Esta seção é dedicada em apresentar um método heurístico alternativo, responsável por gerar uma solução inteira para PCUI (4.3), com os p padrões de corte fornecidos a priori. Esses padrões podem ser dados pelo método de Gilmore-Gomory (onde $p = m$) ou por alguma escolha do usuário ($p \neq m$).

A ideia da Heurística de Arredondamento por Direções Simplex (HADS) é inovadora, e fundamenta na relaxação linear do problema de corte e na busca por uma solução inteira baseada nas “vizinhanças” das direções simplex do poliedro que define a região admissível e nos hiperplanos parametrizados da função-objetivo que passam por soluções inteiras. A solução inteira que estamos nos referindo é para o PCUI com respeito aos p padrões de cortes que estamos considerando.

Inicialmente, a solução $x^0 = \tilde{x}$ para relaxação linear do problema (4.3) é obtida pelo consagrado método de Gilmore-Gomory, apresentado no início deste capítulo. Uma vez que as restrições de demanda para este problema são sempre do tipo “ \geq ”, podemos dizer que esta solução arredondada ao menor inteiro superior também será factível. Consideramos um novo problema ao acrescentar uma nova restrição, denominada de agora em diante de H e definida por

$$\mathbf{1}^T x \leq \mathbf{1}^T \lceil x^0 \rceil,$$

é correto afirmar que a região admissível deste subproblema conterá a solução ótima do problema original, uma vez que $\mathbf{1}^T \lceil x^0 \rceil$ fornece um limitante superior para o valor objetivo z . Além disso, a região admissível desse novo problema se torna um politopo.

Assim sendo, a ideia da HADS consiste em iniciar um processo de busca por uma solução inteira mais promissora caminhando em cada direção simplex da relaxação, iniciando a partir do ponto ótimo relaxado. Escolhida uma direção, quando um vetor

solução tiver pelo menos uma de suas componentes inteiras, arredondamos esta solução para o menor inteiro superior e se verificamos se esta nova solução inteira satisfaz a desigualdade estrita da restrição H . Se isto acontecer, uma nova solução inteira mais promissora foi encontrada, e assim, uma nova restrição H foi detectada, que está relacionada ao hiperplano da função-objetivo que cruza este último vetor inteiro. Este procedimento é prosseguido até que não seja mais possível avançar na direção atual, devido a limitação das outras restrições do problema, como a não negatividade das variáveis e à própria restrição H mais atual. Esse processo é repetido em todas as direções extremas da relaxação, até que nenhuma solução inteira factível consiga ser obtida dentro do politopo delimitado pelas restrições H e $Ax \geq d$.

O detalhamento deste método é realizado nas subseções a seguir.

4.2.1 Inicialização

Primeiramente, a solução x^0 do Problema de Programação Linear relaxado é obtida. As direções extremas do Problema de Programação Linear, na qual serão realizadas as buscas, podem ser obtidas diretamente da base ótima do problema.

No método simplex, a busca pelo ótimo global é realizada pelas arestas do politopo que define a região admissível do problema. A partir de uma solução básica factível, que geometricamente é um vértice desde poliedro, uma nova solução básica factível é obtida, caminhando-se um tamanho de passo α numa dada direção δ^j , até que um outro vértice do politopo seja atingido. Algebricamente, isto é realizado ao incrementar uma variável não básica (que inicialmente é 0) para algum valor positivo α , na direção na qual nenhuma componente do vetor solução fique negativa.

Suponhamos o PCUI esteja na forma canônica, isto é, o vetor x contém também as variáveis de folgas e que a solução básica x^k é a atual solução no método simplex numa dada iteração k , não necessariamente a ótima. Se uma nova solução x^{k+1} é obtida perturbando-se a componente não básica j de x^k , temos que

$$x^{k+1} = x^k + \alpha\delta^j,$$

com $\alpha > 0$ e δ^j a serem determinados.

Considerando as partições básicas $A = [B \ N]$, $x = [x_B \ x_N]^T$ e $\delta^j = [\delta_B^j \ \delta_N^j]^T$, onde o vetor δ_B^j contém as componentes associadas à direção das variáveis básicas e $\delta_N^j = e^j$ é o j -ésimo vetor canônico. Assim,

$$\begin{aligned} d &= Ax^{k+1} \\ &= A(x^k + \alpha\delta^j) \\ &= Ax^k + \alpha A\delta^j \\ &= d + \alpha A\delta^j, \end{aligned}$$

e a última igualdade implica que $A\delta^j = 0$. Usando-se estas partições, temos que:

$$\begin{aligned} 0 &= A\delta^j \\ &= B\delta_B^j + N\delta_N^j \\ &= B\delta_B^j + Ne^j \\ &= B\delta_B^j + a^j, \end{aligned}$$

isto é, $\delta_B^j = -B^{-1}a^j$ é a direção a ser seguida pelas variáveis básicas considerando uma perturbação na variável não básica j . Logo, uma direção extrema associada à variável não básica j no método simplex é dada por

$$\delta^j = \begin{bmatrix} -B^{-1}a^j \\ e^j \end{bmatrix}.$$

O tamanho de passo α a ser tomado a fim de manter a factibilidade da nova solução, pode ser determinado pelo *teste da razão*, ilustrado no método Primal Simplex no início deste capítulo.

Esta heurística inicia ao calcular *todas* as direções simplex a partir do ótimo do problema relaxado, isto é, δ^j é calculado para todo $j \in \mathcal{N}$. Feito isto, armazenamos os custos de cada direção, dados por $\mathbf{1}^T \delta^j$ e o produto escalar da linha i da matriz A por δ^j , isto é, $a_i^T \delta^j$, para todo $i \in I$ e $j \in \mathcal{N}$.

4.2.2 Identificação do tamanho de passo

Uma vez que a direção extrema δ^j foi escolhida para ser explorada, necessitamos determinar o tamanho de passo α^{max} nesta direção de modo que $x^{k+1} = x^k + \alpha^{max} \delta^j$ e respeitar três tipos de restrições:

- (i) a não negatividade das variáveis;
- (ii) a restrição atual H ;
- (iii) as demais restrições do problema.

Dado o ponto inicial da relaxação x^0 e uma solução atual x^k sobre a direção δ^j , o passo α^{min}

$$\alpha^{min} = \min_{1 \leq i \leq p} \begin{cases} \frac{\lfloor x_i^k + 1 \rfloor - x_i^k}{d_i}, & \text{se } d_i > 0 \\ \frac{\lceil x_i^k - 1 \rceil - x_i^k}{d_i}, & \text{se } d_i < 0, \end{cases} \quad (4.5)$$

se menor ou igual à α^{max} , faz chegar a uma solução x^{k+1} com alguma componente inteira. Esta nova solução atual, terá todas suas componentes arredondadas para o menor inteiro

superior ao resultar na solução $\bar{x}^{k+1} = [x^{k+1}]$ e então será testada na função-objetivo para atualização ou não do hiperplano H mais recente.

A seguir, apresentamos um procedimento para computar α^{max} e sua atualização do decorrer da busca, tratando cada um dos três conjuntos de restrições separadamente.

4.2.2.1 Restrições de não negatividade

Para que a solução atual x^k não infrinja as restrições de não negatividade das variáveis, necessitamos estabelecer um tamanho de passo limite para α . Para visualizar isto, considere Figura (20), onde na direção δ^2 , podemos caminhar um tamanho de passo máximo de tal forma que o ponto x^{max} seja atingido.

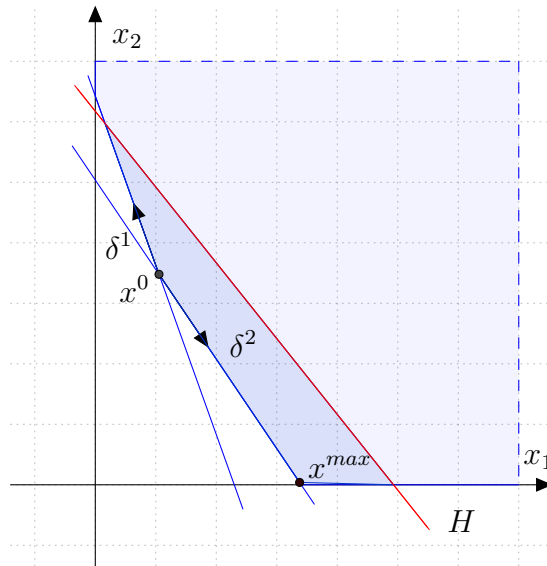


Figura 20 – Tamanho de passo limite para satisfazer a restrições de não negatividade

Assim, na solução x^k mais atual devemos impor a seguinte condição:

$$x^k = x^0 + \alpha\delta^j \geq 0 \Rightarrow x_i^k = x_i^0 + \alpha\delta_i^j \geq 0, \quad i = 1, \dots, p.$$

Se $\delta_i^j \geq 0$, não há restrição para α ; por outro lado se $\delta_i^j < 0$ temos que:

$$0 \leq \alpha \leq \frac{-x_i^0}{\delta_i^j}.$$

Então um limitante superior para α seria tomar α_1 como

$$0 \leq \alpha_1 = \min_{1 \leq i \leq p} \left\{ \frac{-x_i^0}{\delta_i^j} : \delta_i^j < 0 \right\}.$$

4.2.2.2 Restrição do hiperplano H

Para que a nova solução x^{k+1} seja inteira, é necessário estabelecer um tamanho de passo limite α de modo que não viole a restrição do hiperplano H atual. A Figura (21)

ilustra em duas dimensões, uma situação típica no PCUI, onde temos duas restrições de demanda, cuja região admissível é o polítopo em azul claro e cujo ótimo relaxado é em x^0 . Com este ponto, geramos o hiperplano H (em vermelho), delimitando um poliedro em azul escuro onde deverá estar o ótimo do problema inteiro.

No ponto x^0 existem duas direções simplex: δ^1 e δ^2 . Caminhando em direção à δ^1 , um tamanho de passo máximo igual à α_2 é permitido, o suficiente para ativar a restrição H no ponto x^{max} .

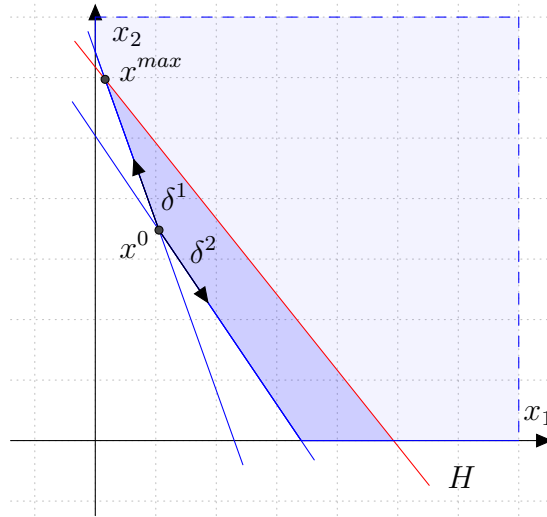


Figura 21 – Tamanho de passo limite para satisfazer a restrição H

Uma maneira de estimar um limitante superior para α_2 seria usar o fato de que, em x^0 , o valor da função-objetivo é menor ou igual em qualquer ponto sobre a direção δ^j escolhida. Como x^0 é o mínimo global do problema relaxado, então $\mathbf{1}^T \delta^j \geq 0$, pois caso contrário, ter-se-ia uma direção de recessão que melhoraria a função-objetivo no ponto x^0 .

Assim, necessitamos ter que o ponto atual $x^k = x^0 + \alpha \delta^j$ não viole a restrição H que passa pelo ponto inteiro \bar{x}^k atual. Pela definição do hiperplano H , como $\mathbf{1}^T x^l \leq \mathbf{1}^T \bar{x}^k$, então:

$$\mathbf{1}^T x^k = \mathbf{1}^T (x^0 + \alpha \delta^j) \leq \mathbf{1}^T \bar{x}^k$$

e isto implica que

$$\alpha \mathbf{1}^T \delta^j \leq \mathbf{1}^T (\bar{x}^k - x^0)$$

ou seja, se $\mathbf{1}^T \delta^j \neq 0$, então:

$$0 \leq \alpha_2 = \frac{\mathbf{1}^T (\bar{x}^k - x^0)}{\mathbf{1}^T \delta^j}$$

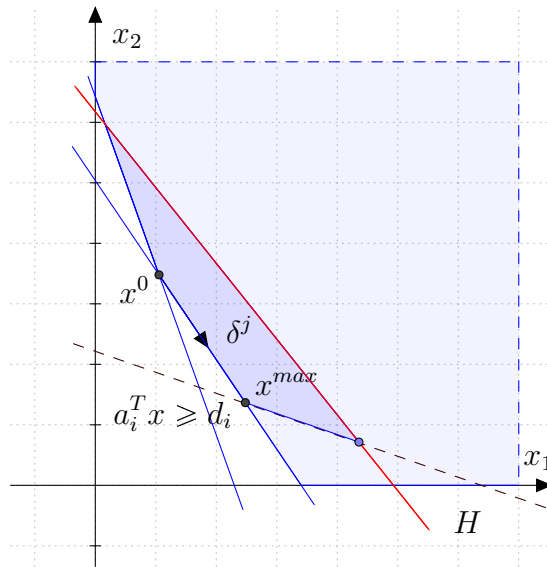
é o tamanho de passo a ser adotado de modo que a factibilidade da restrição H seja mantida.

4.2.2.3 Restrições $Ax \geq d$

A fim de limitar o tamanho de passo de modo a não violar as demais restrições do PCUI, procedemos de maneira análoga às deduções anteriores. Sendo $x^k = x^0 + \alpha\delta^j$, devemos impor a seguinte condição de modo a não infringir qualquer restrição do tipo $a_i^T x^l \geq d_i$:

$$a_i^T x^k = a_i^T (x^0 + \alpha\delta^j) \geq d_i \Rightarrow \alpha a_i^T \delta^j \geq d_i - a_i^T x^0.$$

A Figura (23) ilustra a situação de percorrer na direção δ^j no máximo até atingir o ponto x^{max} , que é o encontro com a restrição $a_i^T x \geq d_i$.


 Figura 22 – Tamanho de passo limite para satisfazer as restrições $Ax \geq d$

O caso $a_i^T \delta^j \geq 0$ não oferece um limitante para α , uma vez que o lado direito da última desigualdade é sempre positivo. Mais precisamente, $d_i - a_i^T x^0$ é a folga da i -ésima restrição de demanda no ótimo relaxado, e por isso é sempre positiva. No entanto, se $a_i^T \delta^j < 0$, temos que:

$$\alpha \leq \frac{d_i - a_i^T x^0}{a_i^T \delta^j},$$

e então tomando o tamanho de passo α_3 sendo:

$$0 \leq \alpha_3 = \min_{1 \leq i \leq p} \left\{ \frac{d_i - a_i^T x^0}{a_i^T \delta^j} : a_i^T \delta^j < 0 \right\}.$$

Baseado nas deduções anteriores, um tamanho de passo α^{max} que irá respeitar os três conjuntos de restrições considerados é dado por:

$$\alpha^{max} = \min \{ \alpha_1, \alpha_2, \alpha_3 \}, \quad (4.6)$$

para cada direção extrema δ^j , $j \in \mathcal{N}$, tomada sobre o ótimo relaxado.

4.2.3 Busca na direção δ^j e atualização de α^{max}

Tomada uma das direções de busca δ^j , temos o valor máximo do passo a ser tomado, α^{max} e o hiperplano inicial H passando por $[x^0]$. O procedimento parte do ponto do x^0 e em seguida calculamos o primeiro ponto x^1 que possui alguma componente inteira, utilizando para isso o tamanho de passo α^{min} dado por (4.5). Se $\alpha^{min} \leq \alpha^{max}$, o movimento nesta direção é autorizado ao ponto x^1 , que passa a ser a nova solução atual. Caso contrário, abandonamos esta direção e iniciamos a busca em outra.

Com x^1 , obtemos um outro ponto inteiro $\bar{x}^1 = [x^1]$. Se

$$\mathbf{1}^T \bar{x}^1 \geq \mathbf{1}^T [x^0],$$

então a restrição H inicial é mantida e um possível passo pode ser dado. Caso contrário, se o novo ponto inteiro satisfizer estritamente a restrição H , esta é atualizada, fazendo-a passar por \bar{x}^1 . Nesse caso, o valor de α^{max} é atualizado da seguinte maneira:

$$\alpha^{max} \leftarrow \alpha^{max} - \alpha^{min}.$$

Se após a atualização, $\alpha^{max} > 0$, prosseguimos a busca nesta direção. Caso contrário a busca é abandonada e uma outra direção é tomada.

Este processo de atualização da restrição H pode vir alterar o valor de α^{max} na direção em curso, mesmo quando o passo não for realizado. Entretanto, assim como foi realizado anteriormente na dedução do tamanho de passo para a restrição H , este inconveniente pode ser resolvido calculando-se $\alpha^{H'}$:

$$\alpha^{H'} = \frac{\mathbf{1}^T (\bar{x}^k - x^k)}{\mathbf{1}^T \delta^j},$$

e atualizando α^{max} como

$$\alpha^{max} \leftarrow \min \left\{ \alpha^{max}, \alpha^{H'} \right\}.$$

A Figura (23) ilustra a necessidade desta atualização. Notamos que a restrição H original, o ponto x^{max} era o limite para o avanço na direção atual; após a alteração do hiperplano, o passo nesta direção fica mais reduzido para até $x^{max'}$, eliminando pontos que não seriam promissores na busca por uma solução inteira.

Assim, este procedimento é realizado até que $\alpha^{max} \geq 0$ em todas as direções δ^j . A solução que pelo qual o último hiperplano H passa é a melhor solução inteira encontrada pela HADS.

Exibimos um exemplo numérico para este método a seguir.

Exemplo 4.3. *Seja o seguinte PCUI com apenas dois padrões de corte, cujos custos são 6 e 1. O objetivo é minimizar o custo da produção.*

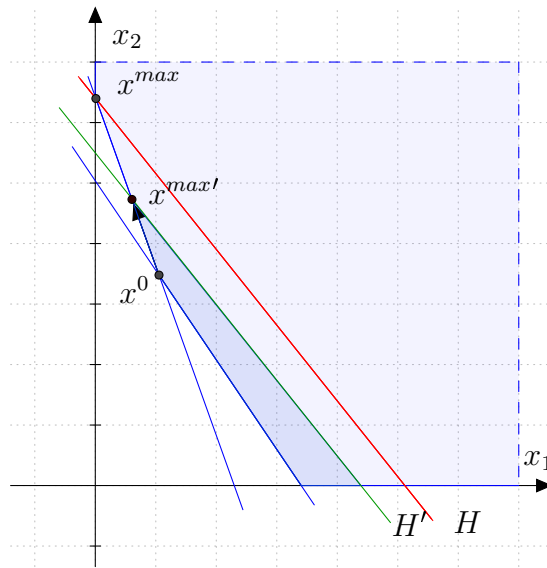


Figura 23 – Tamanho de passo sendo reduzido com a atualização de H

$$\begin{aligned}
 & \text{Minimize } z = 6x_1 + x_2 \\
 & \text{sujeito a } 5x_1 + x_2 \geq 10.1 \\
 & \quad x_1 + x_2 \geq 3 \\
 & \quad x_1 \geq 1.1 \\
 & \quad x_1, x_2 \in \mathbb{N}.
 \end{aligned}$$

Primeiramente, resolvemos a relaxação linear deste problema e encontramos como solução $x^0 = (1.1, 4.6)^T$ e $(6, 1)^T[x^0] = 17$. A região admissível está exibida na Figura (25) bem como o primeiro hiperplano $6x_1 + x_2 = 17$.

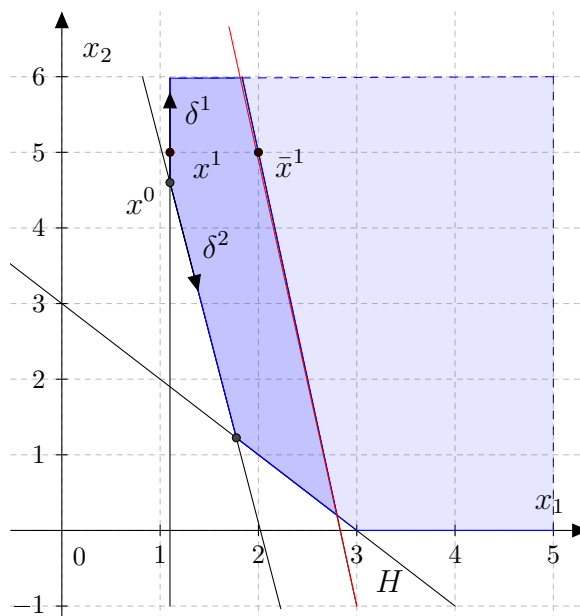


Figura 24 – Região admissível e o primeiro hiperplano para o Exemplo 4.2.1

O método simplex fornece como direções simplex no ponto ótimo relaxado x^0 os vetores $\delta^1 = (0, 1)^T$ e $\delta^2 = (1, -5)^T$.

- **Iteração 1:** Escolhemos primeiramente a direção $\delta^1 = (0, 1)^T$. Determinamos o tamanho de passo α^{max} calculando separadamente:

- Tamanho de passo α_1 para satisfazer a não negatividade das variáveis:

$$\alpha_1 = \min \left\{ \frac{-x_i^0}{\delta_i^1} : \delta_i^1 < 0 \right\} = \infty.$$

- Tamanho de passo α_2 para satisfazer a restrição H :

$$\alpha_2 = \frac{(6, 1) \cdot \left(\begin{bmatrix} 2 \\ 5 \end{bmatrix} - \begin{bmatrix} 1.1 \\ 4.6 \end{bmatrix} \right)}{(6, 1) \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix}} = 5.8.$$

- Tamanho de passo α_3 para satisfazer as outras restrições do problema: note que nenhum produto $a_1^T \delta^1 = 1 > 0$, $a_2^T \delta^1 = 1 > 0$ e $a_3^T \delta^1 = 0$ é negativo, logo $\alpha_3 = \infty$.

Assim,

$$\alpha^{max} = 5.8.$$

Limitado o tamanho de passo a ser dado nesta direção, calculamos o passo mínimo α^{min} a fim de arredondar uma componente do ponto inicial x^0 . Para isso, usamos a expressão (4.5) para obter $\alpha^{min} = 0.4 < 5.8 = \alpha^{max}$ logo o movimento é permitido para obter o ponto $x^1 = (1.1, 5)$ e assim, a primeira solução inteira obtida pela HADS é dada por $\bar{x}^1 = \lfloor x^1 \rfloor = (2, 5)^T$. Atualizamos $\alpha^{max} = 5.8 - 0.4 = 5.4$.

Testando este ponto na função-objetivo do problema, obtemos o valor 17, isto é, a desigualdade estrita para H não foi obtida. Logo, esta restrição é ainda mantida e um próximo passo é dado, se possível.

- **Iteração 2:** Calculamos α^{min} :

$$\alpha^{min} = \frac{\lfloor 5 + 1 \rfloor - 5}{1} = 1$$

e como $\alpha^{min} = 1 < 5.4 = \alpha^{max}$, autorizamos o movimento para o ponto $x^2 = (1.1, 5)^T + 1 \cdot (0, 1) = (1.1, 6)^T$. Arredondando-o, temos $\bar{x}^2 = (2, 6)^T$ cuja função-objetivo é igual à $18 > 17$. Isto significa que este ponto é infactível para o problema com a restrição H . Prosseguindo mais 5 iterações possíveis nessa direção, nenhum ponto inteiro faz a função-objetivo reduzir de valor, logo esta direção é abandonada.

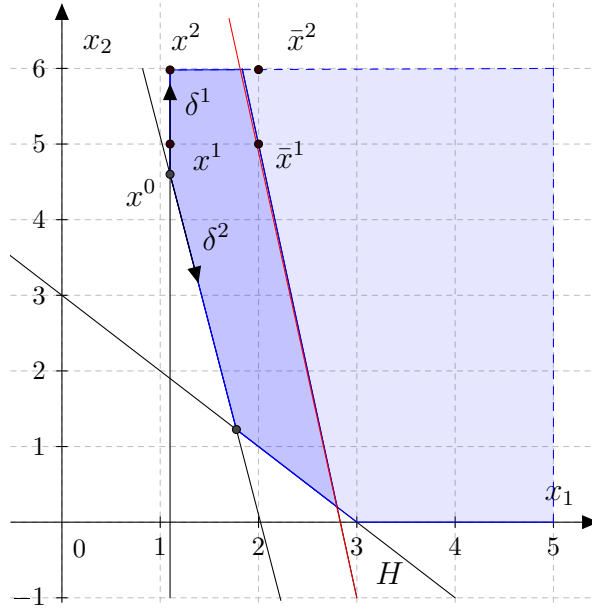


Figura 25 – Direção δ^1 sendo explorada no Exemplo 4.2.1

- **Iteração 3:** escolhemos uma outra direção extrema, $\delta^2 = (1, -5)^T$, e calculamos α^{max} :

– Tamanho de passo α_1 para satisfazer a não negatividade das variáveis:

$$\alpha_1 = \min \left\{ \frac{-x_i^0}{\delta_i^1} : \delta_i^1 < 0 \right\} = \frac{-4,6}{-5} = 0.92.$$

– Tamanho de passo α_2 para satisfazer a restrição H atual:

$$\alpha_2 = \frac{(6, 1) \cdot \left(\begin{bmatrix} 2 \\ 5 \end{bmatrix} - \begin{bmatrix} 1.1 \\ 4.6 \end{bmatrix} \right)}{(6, 1) \cdot \begin{bmatrix} 1 \\ -5 \end{bmatrix}} = 5.8.$$

– Tamanho de passo α_3 para satisfazer as outras restrições do problema. Como apenas $a_2^T \delta^2 = -4 < 0$:

$$\alpha_3 = \min \left\{ \frac{3 - (1.1 + 4.6)}{4} \right\} = \frac{-2.7}{-4} = 0.675.$$

Assim,

$$\alpha^{max} = \min\{0.92, 5.8, 0.675\} = 0.675.$$

Calculamos o tamanho de passo α^{min} pela expressão (4.5) e obtemos $\alpha^{min} = 0.12 < 0.675 = \alpha^{max}$, logo o movimento é permitido. O novo ponto atual é $x^3 = (1.1, 4.6)^T + 0.12 \cdot (1, -5)^T = (1.22, 4)$ e daí, $\bar{x}^3 = (2, 4)^T$. Atualizamos $\alpha^{max} = 0.675 - 0.12 = 0.55$ e testamos este novo ponto inteiro na função-objetivo, que é $16 < 17$, assim, a

restrição H é atualizada. Nesse caso, o valor de α^{max} deverá ser atualizado pelas expressões:

$$\alpha^{H'} = \frac{(6, 1) \cdot \left(\begin{bmatrix} 2 \\ 4 \end{bmatrix} - \begin{bmatrix} 1.22 \\ 4 \end{bmatrix} \right)}{(6, 1) \cdot \begin{bmatrix} 1 \\ -5 \end{bmatrix}} = 4.68,$$

e assim, $\alpha^{max} = \min\{4.68, 0.555\} = 0.555$.

- **Iteração 4:** movimentamos ainda nessa direção a partir do ponto atual x^3 . Para isto, determinamos α^{min} por:

$$\alpha^{min} = \min \left\{ \frac{[1.22 + 1] - 1.22}{1}, \frac{[4 - 1] - 4}{-5} \right\} = \min\{0.78, 0.2\} = 0.2 < 0.55 = \alpha^{max}$$

logo o movimento é realizado, originando o próximo ponto atual $x^4 = (1.42, 3)^T$ e uma nova solução inteira $\bar{x}^4 = (2, 3)^T$. Assim, $\alpha^{max} = 0.555 - 0.2 = 0.355$. Testando este novo ponto na função-objetivo, temos o valor $15 < 16$ e assim, a restrição H é atualizada pela restrição $6x_1 + x_2 \leq 15$.

Isto significa que devemos atualizar α^{max} novamente nesta mesma iteração:

$$\alpha^{H'} = \frac{(6, 1) \cdot \left(\begin{bmatrix} 2 \\ 3 \end{bmatrix} - \begin{bmatrix} 1.42 \\ 3 \end{bmatrix} \right)}{(6, 1) \cdot \begin{bmatrix} 1 \\ -5 \end{bmatrix}} = 3.48,$$

e assim, $\alpha^{max} = \min\{3.48, 0.355\} = 0.355$.

- **Iteração 5:** movimentando ainda nessa direção se $\alpha^{min} \leq \alpha^{max} = 0.355$. Nesse caso:

$$\alpha^{min} = \min \left\{ \frac{[1.42 + 1] - 1.42}{1}, \frac{[3 - 1] - 3}{-5} \right\} = \min\{0.78, 0.2\} = 0.2,$$

logo obtemos um novo ponto atual, a saber, $x^5 = (1.62, 2)$ e $\bar{x}^5 = (2, 2)^T$. Atualizamos $\alpha^{max} = 0.355 - 0.2 = 0.155$. Testamos o novo ponto inteiro \bar{x}^5 na função-objetivo e obtemos o valor $14 < 15$. Isso significa que a restrição do hiperplano deve ser atualizada. Pelo fato de H ter sido trasladada devemos atualizar α^{max} novamente:

$$\alpha^{H'} = \frac{(6, 1) \cdot \left(\begin{bmatrix} 2 \\ 2 \end{bmatrix} - \begin{bmatrix} 1.62 \\ 2 \end{bmatrix} \right)}{(6, 1) \cdot \begin{bmatrix} 1 \\ -5 \end{bmatrix}} = 2.28,$$

e assim, $\alpha^{max} = \min\{2.28, 0.155\} = 0.155$.

- **Iteração 6:** testamos a possibilidade de caminhar novamente nessa direção. Como $\alpha^{min} = 0.2 = \alpha^{min} > 0.155 = \alpha^{max}$, o procedimento é encerrado.

A solução inteira fornecida pela HADS é a que suporta o hiperplano H atual, nesse caso, $\bar{x}^5 = (2, 2)^T$. A solução ótima para este problema no entanto é o ponto $x^* = (2, 1)^T$.

A Figura (26) ilustra os pontos obtidos e as atualizações nos hiperplanos.

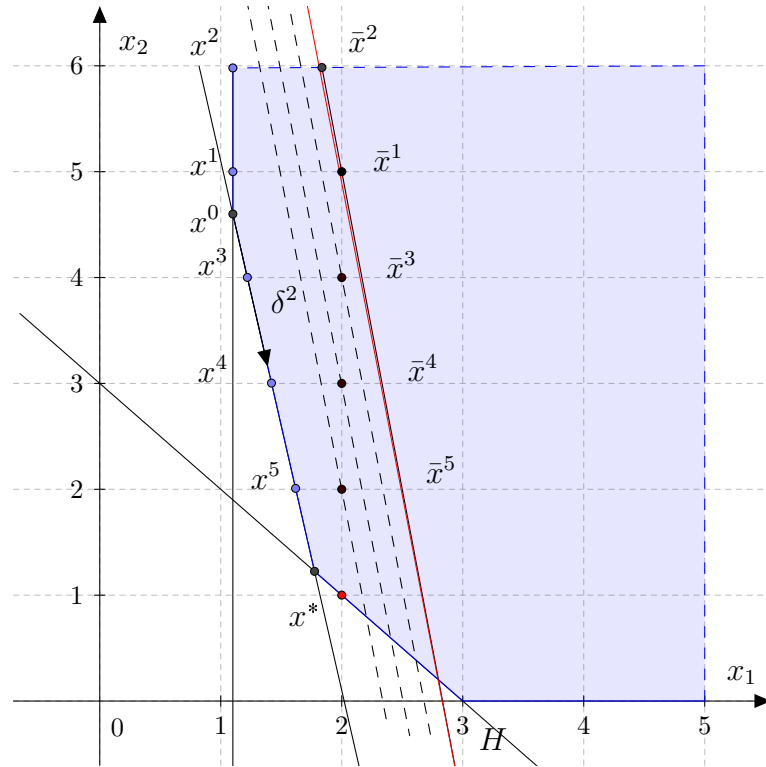


Figura 26 – Direção δ^2 sendo explorada e soluções inteiras no Exemplo 4.2.1

Note que não foi possível obter a solução exata por este procedimento, tendo em vista que, a restrição $x_1 + x_2 \geq 3$ ter impedido o avanço na direção δ^2 para obter o ponto $x^6 = (1.82, 1)^T$.

Organizamos em um pseudocódigo esta heurística.

Algoritmo 3: Heurística de Arredondamento por Direções Simplex

- 1: Entrada: dados do PCUI
 - 2: Obtenha os p padrões de cortes para construir a matriz A e consideramos o problema relaxado (4.3). Seja x^0 a solução do PCUI relaxada e B a matriz básica ótima
 - 3: Calculamos $\delta^j = B^{-1}a^j$ para todo $j \in \mathcal{N}$ não básico, que são as direções simplex em x^0
 - 4: Seja $\bar{x}^0 = \lceil x^0 \rceil$ e construímos a restrição H nesse ponto
 - 5: $l = 0$
 - 6: **for** $j \in \mathcal{N}$ **do**
 - 7: Calculamos α_1, α_2 e α_3 e $\alpha^{max} = \min\{\alpha_1, \alpha_2, \alpha_3\}$
 - 8: Calculamos α^{min} usando como ponto de partida x^l
 - 9: **while** $\alpha^{min} \leq \alpha^{max}$ **do**
 - 10: $\alpha^{max} = \alpha^{max} - \alpha^{min}$
 - 11: $x^{k+1} = x^k + \alpha^{min} \delta^j$
 - 12: $\bar{x}^{k+1} = \lceil x^{k+1} \rceil$
 - 13: **if** $\mathbf{1}^T \bar{x}^{k+1} \leq \mathbf{1}^T \bar{x}^k$ **then**
 - 14: Atualizamos a restrição H , isto é, determinamos H'
 - 15: Calculamos $\alpha^{H'}$ e atualizamos $\alpha^{max} = \min\{\alpha^{H'}, \alpha^{max}\}$
 - 16: Calculamos α^{min} usando como ponto de partida x^{k+1}
 - 17: $k = k + 1$
 - 18: **end if**
 - 19: **end while**
 - 20: **end for**
 - 21: Saída: $\bar{x}^{k+1} \in \mathbb{Z}^p$
-

Esta heurística pode ter vários outros melhoramentos, como por exemplo, tomar algumas combinações convexas de duas ou mais direções em vez de adotar apenas as direções simplex δ^j isoladamente. Um aspecto importante a ser explorado, alternativamente, seria em tomar apenas aquelas direções simplex que fazem um ângulo mais agudo com a função-objetivo. Outra sugestão de melhoramento, seria em adotar como função de arredondamento as heurísticas apresentadas na primeira metade deste capítulo.

4.2.4 Limitantes para a função-objetivo

Quando o PCUI é tratado heurísticamente, como foi apresentado neste capítulo, é necessário estabelecer um critério ou uma métrica que mensure a qualidade da solução encontrada.

O modelo com as restrições de integralidade relaxada para o PCUI (4.1) sem considerar o *setup*, possui uma relação linear forte ((CHEITHAUER; TERNO, 1995)).

Isto significa que o *gap* entre a solução inteira ótima e a relaxação linear é muito pequeno, sendo desconhecido exemplares que tenham um *gap* de integralidade superior à $7/6$. Esta é a principal razão de muitos trabalhos na literatura que abordam métodos heurísticos, considerarem os padrões básicos na geração de soluções inteiras. No entanto, o modelo (4.4) para o PCUI relaxado que considera *setup*, possui uma relaxação linear de péssima qualidade, sendo pouco utilizado na prática.

Essa propriedade é ilustrada na referência em (WASCHER; GAU, 1996) e é muito utilizada para chegar a qualidade de soluções heurísticas em problemas combinatórios.

Definição 4.1 (IRUP - *Integer Round Up Property*). *Preservando a notação anterior, considere \tilde{x} a solução ótima do problema (4.1) relaxado e \bar{x} uma solução inteira para o mesmo problema. Se*

$$[\mathbf{1}^T \tilde{x}] = \mathbf{1}^T \bar{x},$$

então \bar{x} é ótima para o PCUI. Uma classe de problemas onde o valor objetivo de uma solução factível coincide com o valor objetivo arredondado para o menor inteiro superior para a relaxação é chamado de IRUP (Integer Round Up Property).

No entanto, em 1986 no trabalho apresentado em (MARCOTE, 1986), foi demonstrado que a IRUP não é válida para o PCUI. Logo, um limitante inferior \underline{z} para z é $\underline{z} = [\mathbf{1}^T \tilde{x}]$, sendo uma boa estimativa para medir a qualidade da solução heurística encontrada. Por outro lado, um limitante superior \bar{z} para z pode ser dado por $\bar{z} = \mathbf{1}^T \lceil \tilde{x} \rceil$.

Este capítulo apresentou alguns dos métodos heurísticos mais clássicos para tratar o problema de corte com único objetivo e sem levar em conta o *setup*. Entre eles, detalhamos o método de Geração de Colunas de Gilmore-Gomory, algumas heurísticas de arredondamento. Além disso, propusemos um novo método, a HADS, distinto dos demais e que pode também ser aplicado ao mesmo problema. Este novo método será adaptado juntamente com as técnicas multiobjetivos de escalarizações, para gerar um conjunto de soluções eficientes para o PCUIM de maneira mais acelerada. O capítulo seguinte apresenta como fizemos esta adaptação bem como nas técnicas de escalarizações propostas.

Capítulo 5

Os Métodos de Escalarizações Propostos no Problema de Corte Multiobjetivo

Neste capítulo, enfocamos os métodos de escalarizações que foram desenvolvidos e/ou adaptados neste trabalho para resolver o PCUIM. Usamos os resultados e propriedades teóricas das metodologias clássicas de otimização multiobjetivo apresentados no Capítulo 3 para garantir a eficiência das soluções encontradas. Além disso, nossa outra preocupação consiste em encontrar *todas* as soluções do problema que estamos resolvendo.

Em todo este capítulo, $\mathcal{X} \neq \emptyset$ denotará a região admissível e $x \in \mathcal{X}$ o vetor contendo todas as variáveis do PCUIM. Se os padrões de corte não forem fornecidos, o conjunto \mathcal{X} será determinado pelas restrições do modelo (2.20) ou (2.21) e x conterá as variáveis inteiras com as frequências dos padrões de corte, as variáveis binárias que contabilizam o *setup*, as variáveis que definem um padrão de corte bem como as variáveis auxiliares para as devidas linearizações. Por outro lado, para o caso onde os padrões de corte forem fornecidos, \mathcal{X} será obtido pelas restrições do modelo (2.22) ou (2.24) e x será o vetor com as variáveis decisórias contendo as frequências e o *setup*. As funções a serem minimizadas serão z_1 e z_2 determinam a soma das frequências do número de padrões de cortes utilizados e o *setup*, respectivamente.

Na seção a seguir, descrevemos como os métodos multiobjetivos foram implementados, destacando algumas adaptações que foram realizadas.

Em duas escalarizações que serão apresentadas nesse capítulo (Soma Ponderada e Métrica de Tchebycheff), necessitamos normalizar as funções-objetivos envolvidas para que a diferença de dimensão entre elas não interfira no processo de obtenção de soluções eficientes. Para isso, precisamos determinar o pontos lexicográficos do espaço \mathcal{Z} . Isso é feito minimizando-se cada objetivo separadamente:

$$\begin{aligned} \text{Minimize} \quad & z = z_1 + \rho z_2 \\ \text{sujeito a} \quad & x \in \mathcal{X} \end{aligned} \tag{5.1}$$

e

$$\begin{aligned} \text{Minimize} \quad & z = z_2 + \rho z_1 \\ \text{sujeito a} \quad & x \in \mathcal{X}, \end{aligned} \tag{5.2}$$

onde $\rho > 0$ é uma constante. A determinação dos pontos lexicográficos exige um certo cuidado, pois no PCUIM há soluções múltiplas quando os objetivos são minimizados individualmente. A parcela adicional ρz_i na função-objetivo destes subproblemas evitam soluções alternativas, e conseqüentemente, soluções dominadas. É importante que ρ seja um valor pequeno o suficiente para que o problema não dê solução alternativa. Nas nossas implementações, $\rho = 10^{-4}$.

Sejam x^{*1} e x^{*2} as soluções ótimas dos problema (5.1) e (5.2), respectivamente. Retomando a notação vista em capítulos anteriores, denotamos por $f_1(x^{*1}) = z_1^-$, $f_1(x^{*2}) = z_1^+$, $f_2(x^{*2}) = z_2^-$ e por $f_2(x^{*1}) = z_2^+$. Logo, o vetor ideal será obtido por $z^I = (z_1^-, z_2^-)^T$, os pontos lexicográficos da fronteira de Pareto são dados por $z^{*1} = (z_1^-, z_2^+)^T$ e $z^{*2} = (z_1^+, z_2^-)^T$ e o vetor nadir por $z^N = (z_1^+, z_2^+)^T$.

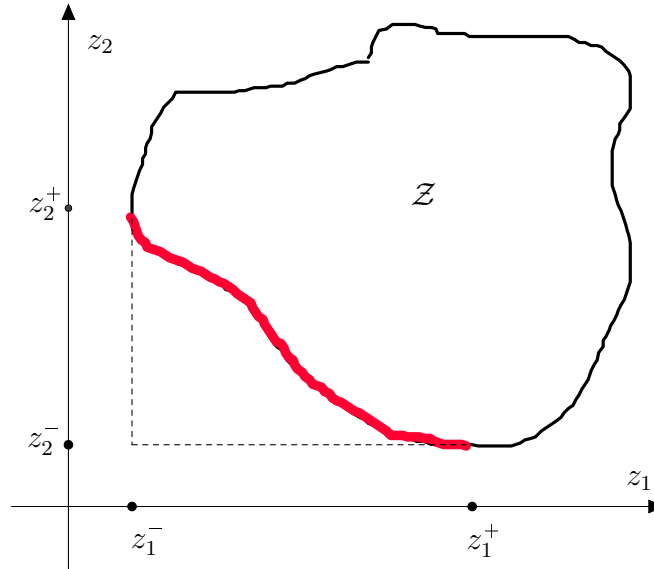
Partindo do pressuposto que o problema tem os dois lexicográficos distintos e usando a expressão (1.3), obtemos os objetivos z_1^{norm} e z_2^{norm} normalizados como:

$$z_1^{\text{norm}} = \frac{z_1 - z_1^-}{z_1^+ - z_1^-} \quad \text{e} \quad z_2^{\text{norm}} = \frac{z_2 - z_2^-}{z_2^+ - z_2^-}.$$

Em virtude do Teorema de Benson ((BEENSON, 1971)), estes pontos delimitam os intervalos mínimo e máximo das funções-objetivos z_1 e z_2 no espaço \mathcal{Z}^* , como pode ser visto na Figura (27).

No espaço de critério normalizado, o lexicográfico z^{*1} será levado no ponto (0, 1) ao passo que outro ponto lexicográfico no ponto (1, 0). Além disso, as normalizações para z^N e z^I estarão associados aos pontos (1, 1) e (0, 0), respectivamente.

Nas próximas subseções detalhamos como cada método multiobjetivo foi aplicado e adaptado no PCUIM.


 Figura 27 – Intervalos de mínimo e máximo no espaço Z^* das soluções eficientes

5.1 O Método Soma Ponderada

O Problema Ponderado no parâmetro w , com $w \in [0, 1]$ é definido como:

$$\begin{aligned} \text{Minimize} \quad & z^w = w \cdot z_1^{\text{norm}} + (1 - w) \cdot z_2^{\text{norm}} \\ \text{sujeito a} \quad & x \in \mathcal{X}. \end{aligned} \quad (5.3)$$

O algoritmo apresentado a seguir, gera a cada iteração, uma solução eficiente para o PCUIM. Não garantimos que a mudança de pesos produz diferentes soluções eficientes, tampouco que haverá uma distribuição regular dos pontos não dominados ao longo da fronteira de Pareto. Como saída, teremos um subconjunto $\hat{\mathcal{X}} \subseteq \mathcal{X}^*$ de soluções eficientes.

Algoritmo 4: O Método Soma Ponderada para o PCUIM

- 1: Entrada: dados do Problema de Corte e Δ
 - 2: $\hat{\mathcal{X}} = \{x^{*1}, x^{*2}\}$
 - 3: $\hat{\mathcal{Z}} = \{(z_1^-, z_2^+)^T, (z_1^+, z_2^-)^T\}$
 - 4: Normalizamos os objetivos z_1 e z_2
 - 5: $w = \Delta$
 - 6: **while** $w \leq 1 - \Delta$ **do**
 - 7: Resolvemos (5.3). Sejam x^{*w} e z^w a sua solução e vetor objetivo ótimo associado
 - 8: $\hat{\mathcal{X}} = \hat{\mathcal{X}} \cup \{x^{*w}\}$
 - 9: $\hat{\mathcal{Z}} = \hat{\mathcal{Z}} \cup \{(z_1^w, z_2^w)^T\}$
 - 10: $w \leftarrow w + \Delta$
 - 11: **end while**
 - 12: Saída: $\hat{\mathcal{X}}$ e $\hat{\mathcal{Z}}$
-

A tolerância nas linhas 5 e 6 do pseudocódigo garantem que os pesos $w = 0$ e $w = 1$ não serão considerados no laço principal, uma vez que as soluções eficientes para estes pesos já foram obtidas na normalização dos objetivos. A questão mais importante neste método é definir o tamanho de passo Δ . Existe uma certa dificuldade em defini-lo. Se Δ for muito pequeno, o algoritmo fica lento, porém, evidentemente esperamos obter mais soluções distintas. Por outro, se Δ for grande, o algoritmo fica mais rápido mas podemos perder muitas soluções eficientes. Com efeito, é preciso ter um cuidado na definição deste parâmetro. Este cuidado está relacionado aos

$$\left\lfloor \frac{1 - 2 \cdot \Delta}{\Delta} \right\rfloor$$

Problema de Programação Linear Inteiros que este algoritmo pode determinar para serem otimizados. Uma maneira de definir Δ para este método no PCUIM padrão (com todos os coeficientes na função-objetivo iguais à 1), consiste em adotar

$$\Delta = \frac{1}{z_2^+ - z_2^-}, \quad (5.4)$$

onde $z_2^+ - z_2^-$ é um limitante superior para o número de soluções eficientes que o PCUIM pode ter¹. O tamanho de passo definido desta maneira fará com que a Soma Ponderada gere $z_2^+ - z_2^-$ subproblemas para serem otimizados.

Um outro ponto desfavorável desta metodologia é o fato de gerar apenas soluções eficientes cuja imagem são pontos suportados no espaço de critério. No PCUIM, a fronteira de Pareto não é convexa e conseqüentemente esse procedimento pode gerar um número pequeno de soluções eficientes distintas.

A Figura (28) exhibe o espaço de critério de um PCUIM com seis soluções eficientes. Neste exemplo, assumimos que a projeção da fronteira de Pareto no eixo z_2 sejam os pontos inteiros no intervalo $\mathcal{I} = [z_2^-, z_2^+]$. Suponhamos que o ponto suportado C seja obtido com o peso $w = \frac{1}{2}$. Então para todo $0 \leq w < \frac{1}{2}$ um dois pontos A ou C deverão ser originados e a solução eficiente cuja imagem é o ponto não suportado B não pode ser solução de nenhum problema ponderado. Da mesma forma se $\frac{1}{2} < w \leq 1$, nenhum ponto no espaço de critério diferente dos pontos C ou F será obtido. Note que as soluções cuja imagem são D e E não podem ser determinadas.

Da forma como o algoritmo (4) foi definido e considerando o exemplo desta figura, devemos gerar seis subproblemas (5.3) a serem otimizados. Nesse caso, o tamanho de passo seria $\Delta = \frac{1}{6} = 0.1\bar{6}$.

Um outro inconveniente desta técnica é que um conjunto de pesos uniformemente distribuídos no intervalo $[\Delta, 1 - \Delta]$ podem perder algumas soluções eficientes cuja

¹ Se os coeficientes na função-objetivo não forem iguais à 1 este limitante se altera. Para mais detalhes, ver Capítulo 6, Seção 4.

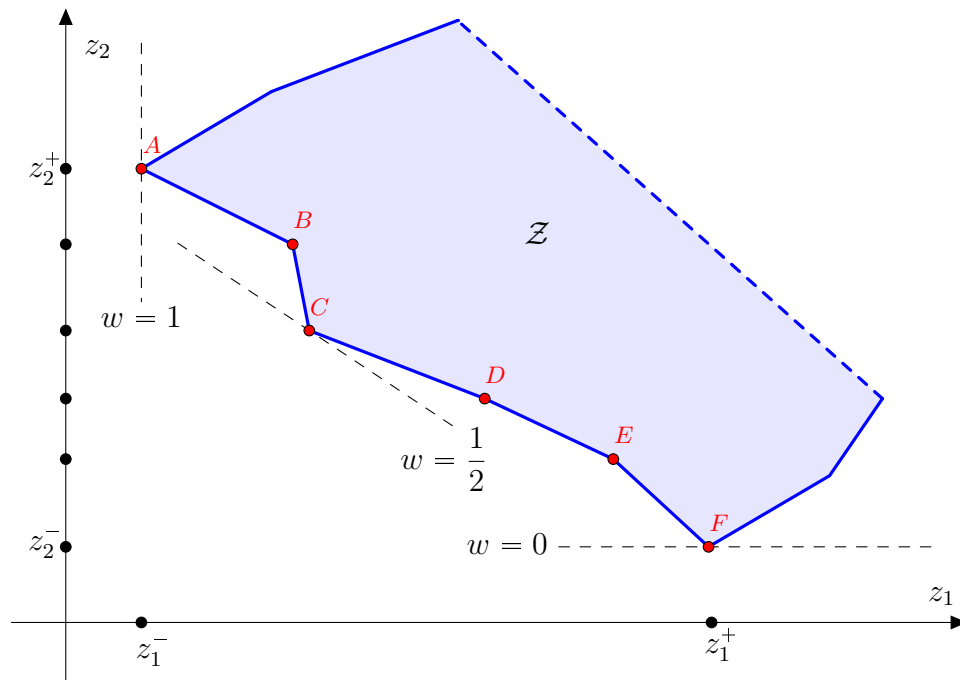


Figura 28 – Exemplo de aplicação da Soma Ponderada

imagem são suportadas. Isto pode acontecer quando dois pontos suportados consecutivos estiverem relativamente próximos um do outro, de modo que a mudança para um novo peso “saltar” pontos próximos.

O procedimento a seguir consegue contornar algumas das dificuldades oriundas deste método.

5.2 O Método da Métrica de Tchebycheff

Como visto no Capítulo 3, esta é uma outra técnica de otimização multiobjetivo que pode ser utilizada para gerar todas as soluções eficientes para o PCUIM. Embora precise de um parâmetro w que pondera os desvios dos objetivos em relação à z^I , este método detecta todos os pontos da fronteira de Pareto, mesmo para problemas não convexos.

Como na Soma Ponderada, é necessário normalizar os objetivos e neste caso o vetor ideal normalizado será $(0, 0)^T$. O Problema de Tchebycheff no peso $w \in [0, 1]$, é definido como aquele que minimiza o maior desvio ponderado entre as funções-objeto e o vetor ideal normalizados, ou seja:

$$\begin{aligned} \text{Minimize} \quad & z^w = \max \{w \cdot z_1^{\text{norm}}, (1 - w) \cdot z_2^{\text{norm}}\} \\ \text{sujeito a} \quad & x \in \mathcal{X}. \end{aligned} \tag{5.5}$$

É muito frequente, na prática, considerar o problema de Tchebycheff ampliado,

onde uma parcela adicional na função-objetivo é considerada de modo a impedir soluções alternativas de (5.5), que serão dominadas. Além disso, linearizamos a função max inserindo-se a variável auxiliar u :

$$\begin{aligned} \text{Minimize} \quad & z^w = u + \rho \cdot (z_1^{\text{norm}} + z_2^{\text{norm}}) \\ \text{sujeito a} \quad & x \in \mathcal{X} \\ & w \cdot z_1^{\text{norm}} \leq u \\ & (1 - w) \cdot z_2^{\text{norm}} \leq u. \end{aligned} \tag{5.6}$$

Conforme w varia no intervalo $[0, 1]$, o problema (5.6) fornece uma solução eficiente. Se $w = 0$ e 1 temos os pontos lexicográficos da fronteira. O parâmetro w , toma valores uniformemente distribuídos no intervalo $[\Delta, 1 - \Delta]$, variando de Δ em Δ , de acordo com a Equação (5.4). À medida que w vai aumentando, os desvios entre z_1^{norm} e 0 vão se tornando cada vez mais importantes na função-objetivo e soluções mais compromissadas em minimizá-la são obtidas.

A Figura (29) ilustra no espaço \mathcal{Z} normalizado, quatro pontos não dominados resultantes de quatro ponderações distintas ($\rho = 0$). As linhas pontilhadas são os contornos das métricas de Tchebycheff ponderadas utilizadas. Com $w = 0$ significa que o desvio de z_1^{norm} em relação à origem não é significativo, originando o ponto não dominado A . O outro extremo é conseguido com $w = 1$. Outras soluções eficientes cuja imagem são pontos suportados (por exemplo, o ponto C) ou não (por exemplo, o ponto E) podem ser geradas, à luz ao Teorema (3.8).

Apresentamos a seguir um pseudocódigo para este método podemos obter o conjunto \mathcal{X}^* se w for convenientemente escolhido. O presente trabalho não encontrou uma maneira de escolher o peso w de modo que todas as soluções eficientes sejam obtidas. Contudo, existem algumas pesquisas na literatura que já trabalharam em como definir w de modo a captar todas as soluções de um problema de otimização biobjetivo combinatório. Citamos as referências apresentadas em (RALPHS; SALTZMAN; WIECEK, 2006) e

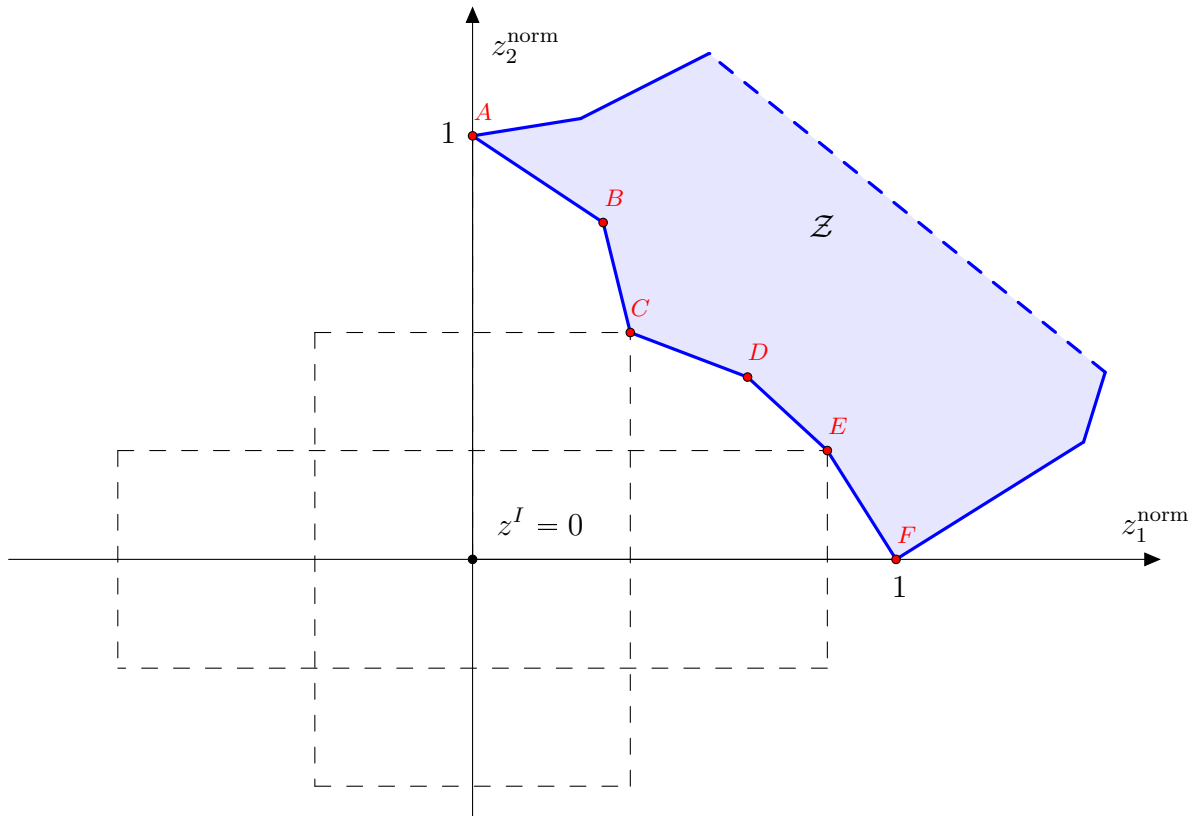


Figura 29 – Método Métrica de Tchebycheff no Problema de Corte multiobjetivo

(SAYIN; KOUVELIS, 2005) deste estudo.

Algoritmo 5: O Método da Métrica de Tchebycheff para o PCUIM

- 1: Entrada: dados do Problema de Corte, ρ e Δ
 - 2: $\mathcal{X}^* = \{x^{*1}, x^{*2}\}$
 - 3: $\mathcal{Z}^* = \{(z_1^-, z_2^+)^T, (z_1^+, z_2^-)^T\}$
 - 4: Normalizamos os objetivos z_1 e z_2
 - 5: $w = \Delta$
 - 6: **while** $w \leq 1 - \Delta$ **do**
 - 7: Resolvemos (5.6). Sejam x^{*w} e z^w a sua solução e vetor objetivo ótimo associado
 - 8: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{*w}\}$
 - 9: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{(z_1^w, z_2^w)^T\}$
 - 10: $w \leftarrow w + \Delta$
 - 11: **end while**
 - 12: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

Existe uma similaridade entre os algoritmos (4) e (5). No entanto, há uma diferença entre os subproblemas escalarizados envolvidos. Além disso, o Método da Métrica de Tchebycheff garante *teoricamente* a geração de \mathcal{X}^* .

Como os Subproblemas (5.6) são combinatórios, nossa meta consiste em gerar o menor número possível destes subproblemas para obter todas as soluções eficientes

do PCUIM. Uma maneira de definir o tamanho de passo Δ no pseudocódigo anterior, é defini-lo como na expressão (5.4). No entanto isto não garante que todas as soluções eficientes serão obtidas. Dito de outra forma, para uma distribuição uniforme de w no intervalo $[\Delta, 1 - \Delta]$ podemos ter subproblemas gerando soluções eficientes já obtidas e ao mesmo tempo não determinar outras ao fim do processo.

Os cinco procedimentos apresentados a seguir dispensam a normalização dos objetivos e garantem, em teoria, a geração de todas as soluções eficientes para o PCUIM. Dois métodos são inéditos, um foi adaptado da literatura e o último foi revisado para ser comparado com os demais.

5.3 O Método Tchebycheff Sem Ponderações

O procedimento apresentado nesta seção, denotado por Método da Métrica de Tchebycheff Sem Ponderações, utiliza uma mistura das ideias do algoritmo NISE ((COHON, 2003)) e contidas nos trabalhos (RALPHS; SALTZMAN; WIECEK, 2006) e (SOLANKI, 1991). O novo algoritmo desenvolvido é especializado para enumerar todas as soluções eficientes para problemas biobjetivos discretos. O algoritmo difere dos métodos originais em três aspectos: (i) na definição dos pontos de referência, (ii) no critério de parada e (iii) na ausência de pesos nos desvios das funções-objetivos com o ponto de referência.

Uma das grandes diferenças entre este e o procedimento clássico de Tchebycheff originalmente proposto em (BOWMAN, 1976), é que neste não existem as ponderações w nos desvios para a obtenção de diferentes soluções eficientes. Usando o fato de que o Método da Métrica de Tchebycheff é capaz de obter qualquer solução eficiente, o método presente toma dois pontos não dominados consecutivos e tentar obter um terceiro entre eles (se existir). Para tanto, o ponto de referência é constantemente atualizado. Com esta ideia, soluções suportadas e não suportadas podem ser geradas.

Considere inicialmente um problema de otimização biobjetivo, onde o espaço \mathcal{Z} é a malha inteira na região hachurada e a fronteira de Pareto são os pontos marcados em vermelho com A , B , C , D , E e F . Primeiramente, os lexicográficos A e F são obtidos e o retângulo $ARFN$ passa a ser conhecido, onde R é o vetor ideal e N o vetor nadir.

Um problema de Tchebycheff (5.6) com $w = 1$ para ambos os desvios poderia ser aplicado utilizando este ponto de referência. No entanto, a solução eficiente resultante poderia ser um ponto no espaço de critério coincidente com A ou F . Para evitar que isso gere um problema ocioso, então deslocamos o ponto R ligeiramente no interior deste retângulo, isto é, consideramos o ponto de referência modificado $R' = R + \Delta$, onde $\Delta > 0$ é uma tolerância. Do mesmo modo modificamos o ponto nadir para $N' = N - \Delta$ e assim, obtemos um novo retângulo $A'R'F'N'$ como na Figura (30). É dentro desta nova região é que pesquisaremos um novo ponto não dominado através do Problema de Tchebycheff

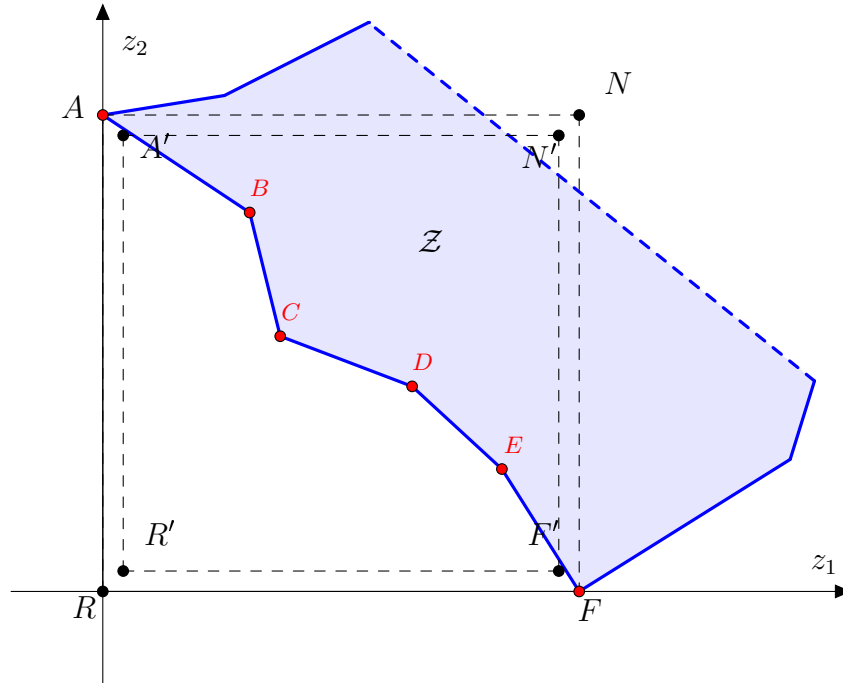


Figura 30 – Pontos de referência e nadir modificados no Problema de Tchebycheff Sem Ponderação

com peso $w = 1$ nos desvios.

Aplicando o Problema (5.6) com $w = 1$ e impondo restrições de modo que a imagem de qualquer ponto admissível permaneça dentro do retângulo $A'R'F'N'$, podemos determinar um ponto não dominado entre A e F se existir. Caso ele não exista, o subproblema detectará uma infactibilidade. A Figura (31) ilustra os contornos da Métrica de Tchebycheff usando como referência R' e o ponto não dominado D sendo detectado.

Após o ponto D ser obtido, as $t = 3$ soluções eficientes até então encontradas são ordenadas pelo valor de z_1 em ordem crescente, isto é, $A = S^1$, $D = S^2$ e $F = S^3$ e novos problemas de Tchebycheff na ponderação $w = 1$ são formulados. Cada um destes novos subproblemas tomam dois pontos consecutivos S^i e S^{i+1} e obtemos novos retângulos com diagonais R' e N' , como é mostrado Figura (32). Note que, se existir uma outra solução eficiente (suportada ou não) entre S^i e S^{i+1} , ela será obtida diferente de ambas, deve estar no interior do retângulo cujo extremo da diagonal são os pontos R' e N' .

Então um problema de Tchebycheff vai sendo aplicado para pares de pontos consecutivos no espaço de critério obtidos de iterações anteriores. O resultado vai sendo a construção da fronteira de Pareto por regiões de buscas limitadas por estes retângulos. No PCUIM esta abordagem foi perfeitamente adaptada, uma vez que a projeção de cada ponto não dominado nos eixos coordenados ser inteiro; nesse caso, a distância entre cada ponto não dominado é “grande” o suficiente, de modo que, se o valor de $\Delta < \frac{1}{2}$, certamente nenhuma solução eficiente será perdida. No entanto, a técnica pode ser adaptável mesmo

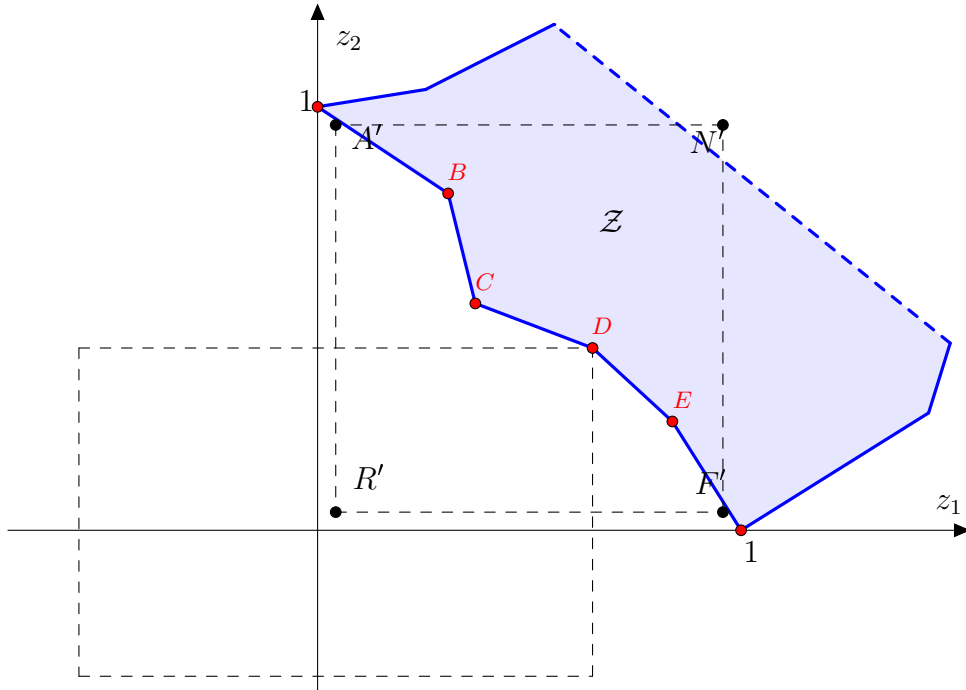


Figura 31 – Novo ponto não dominado obtido com o Problema de Tchebycheff Sem Ponderação

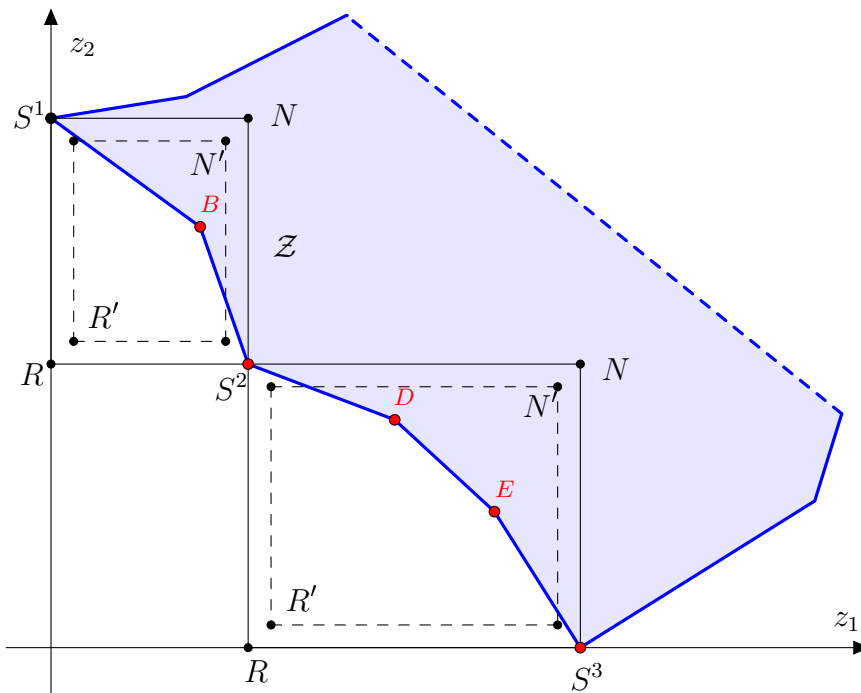


Figura 32 – Soluções eficientes sendo obtidas pelo método de Tchebycheff Sem Ponderações

quando essas projeções não formarem intersecções inteiras e nesse caso, o parâmetro Δ deve ser escolhido de forma mais cuidadosa para não perder soluções eficientes. Na Subseção 6.4 dos resultados computacionais, são propostas algumas variações deste parâmetro, cuja intenção é obter diferentes precisões para a fronteira de Pareto.

O algoritmo é repetido considerando todos os pares consecutivos de pontos não dominados, até que todos os problemas de Tchebycheff na ponderação $w = 1$ fiquem infactíveis. Neste caso, temos todas as soluções eficientes do problema determinadas. Outra grande vantagem desta técnica é seu desempenho não ser afetado pelo distanciamento dentre os pontos lexicográficos, e sim, apenas pelo número de pontos não dominados compreendidos entre eles. O mesmo não ocorre, por exemplo, nos métodos ε -Restrito e Benson, vistos mais adiante.

Suponhamos que já se tenham os pontos não dominados S^1, S^2, \dots, S^t , ordenados em ordem crescente pelo valor de z^1 . Considere o retângulo $S^i R S^{i+1} N$, com $i = 1, \dots, t - 1$, e seus vértices escritos em forma de coordenadas, isto é, $S^i = (z_1^i, z_2^i)^T$, $S^{i+1} = (z_1^{i+1}, z_2^{i+1})^T$, $R = (z_1^i, z_2^{i+1})^T$ e $N = (z_1^{i+1}, z_2^i)^T$. As coordenadas dos vértices modificados serão dadas por $R' = (z_1^i + \Delta, z_2^{i+1} + \Delta)^T$ e $N' = (z_1^{i+1} - \Delta, z_2^i - \Delta)^T$. O Problema de Tchebycheff Sem Ponderação, considerando o par de pontos consecutivos S^i e S^{i+1} é formulado da seguinte maneira:

$$\begin{aligned}
 \text{Minimize} \quad & z^i = u + \Delta \cdot [z_1 - (z_1^i + \Delta) + z_2 - (z_2^{i+1} + \Delta)] \\
 \text{sujeito a} \quad & x \in \mathcal{X} \\
 & z_1 \leq z_1^{i+1} - \Delta \\
 & z_2 \leq z_2^i - \Delta \\
 & z_1 - (z_1^i + \Delta) \leq u \\
 & z_2 - (z_2^{i+1} + \Delta) \leq u \\
 & u \geq 0,
 \end{aligned} \tag{5.7}$$

para $i = 1, \dots, t - 1$ e procura por uma solução eficiente cuja imagem esteja no interior do retângulo $S^i R' S^{i+1} N'$.

As bases teóricas para o funcionamento desta técnica, baseiam-se nos teoremas a seguir.

Teorema 5.1. *Suponha que um Problema de Otimização biobjetivo discreto tenha as soluções eficientes x^1, x^2, \dots, x^t , com $t \geq 2$. Suponha ainda que*

$$z_1^1 > z_1^2 > \dots > z_1^t.$$

Seja $1 < j - 1 < i < j < t$, R e N os pontos de referência e nadir do i -ésimo Problema (5.7). Então este subproblema

- *ou tem uma única solução x^i , e $z_1^{j-1} < z_1^i < z_1^j$;*
- *ou é infactível, caso z^i , a imagem de x^i , não esteja estritamente contida no retângulo de diagonais $R + \Delta$ e $N - \Delta$.*

Demonstração. (i) A unicidade e eficiência de x^i é consequência imediata dos Teoremas (3.6) e (3.7). Basta utilizar R' no lugar de z_i^U naquelas demonstrações.

(ii) Suponha que entre os dois pontos não dominados z_1^{j-1} e z_1^j não exista uma solução eficiente x^i , relacionada ao ponto $z^i = (z_1^i, z_2^i)^T$. Precisamos provar que o Problema (5.7) será infactível. Suponha, por contradição, que ele seja factível, isto é, exista uma solução eficiente x^i , cuja primeira componente de sua imagem satisfaça a relação $z_1^j < z_1^i < z_1^{j+1}$. Assim, este ponto deve obrigatoriamente satisfazer a factibilidade do problema, ou seja, as cinco condições a seguir concomitantemente:

- $x^i \in \mathcal{X}$,
- $z_1^i \leq z_1^{j+1} \Leftrightarrow z_1^i < z_1^{j+1} - \Delta$
- $z_1^i \geq z_1^j \Leftrightarrow z_1^i > z_1^{j+1} + \Delta$
- $z_2^i \leq z_2^j \Leftrightarrow z_2^i < z_2^j - \Delta$
- $z_2^i \geq z_2^{j+1} \Leftrightarrow z_2^i > z_2^{j+1} + \Delta$

Mas essas condições negam a o fato de z^i não pertencer ao retângulo cujas diagonais são dadas pelos extremos R' e N' . Logo, o Problema (5.7) deve ser infactível neste caso. \square

Outros resultados teóricos, de importância prática, são apresentados a seguir.

Teorema 5.2. *O Algoritmo (6) determina todas as soluções eficientes para um Problema de Otimização biobjetivo Combinatório.*

Demonstração. Como $\mathcal{X}^* \subset \mathbb{N}^n$, então ele é finito. Logo, \mathcal{Z}^* também o é. Suponhamos que $t = |\mathcal{Z}^*| < \infty$, então todos os subproblemas $PTSP_i$ serão infactíveis pelo Teorema anterior, para todo $1 \leq i \leq t - 1$. Logo todas as soluções não dominadas serão detectadas por este algoritmo. \square

Teorema 5.3. *O número máximo de Subproblemas (6) a serem resolvidos para encontrar todas as soluções eficientes é igual à $2 \cdot t - 1$, onde $t = |\mathcal{Z}^*|$.*

Demonstração. Como há t soluções não dominadas, então pelo menos t subproblemas devem ser resolvidos. Agora, a cada par de pontos não dominados z^i e z^{i+1} , com $1 \leq i \leq t-1$, há outros $t - 1$ subproblemas infactíveis. Logo, tem-se $t + t - 1 = 2t - 1$ subproblemas a serem resolvidos. \square

O último teorema atesta que a complexidade do Algoritmo (6) é polinomial. Além disso, apenas t subproblemas deverão ser otimizados efetivamente, de modo que os

$t - 1$ remanescentes deverão apenas ser testados quanto à factibilidade. Sem dúvida, esta foi a principal vantagem deste método com relação aos demais algoritmos de escalarização desenvolvidos neste trabalho.

A seguir, organizamos um pseudocódigo para este inédito método de programação multiobjetivo.

Algoritmo 6: O Método Tchebycheff Sem Ponderações para o PCUIM

- 1: Entrada: dados do Problema de Corte e Δ
 - 2: $\mathcal{X}^* = \{x^{*1}, x^{*2}\}$
 - 3: $\mathcal{Z}^* = \{(z_1^1, z_2^1)^T, (z_1^2, z_2^2)^T\} = \{(z_1^-, z_2^+)^T, (z_1^+, z_2^-)^T\}$
 - 4: $t = 2$
 - 5: **for** $i = 1, \dots, t - 1$ **do**
 - 6: $k = 0$ (Contador de problemas infactíveis)
 - 7: Calculamos $R^i = (z_1^i, z_2^{i+1})^T$ e $N^i = (z_1^{i+1}, z_2^i)^T$
 - 8: Resolvemos o Problema (5.7)
 - 9: **if** Se ele é infactível **then**
 - 10: $k = k + 1$
 - 11: **if** $k = t - 1$ **then**
 - 12: Pare. O problema não mais possui soluções eficientes
 - 13: **end if**
 - 14: **else**
 - 15: Sejam x^{*i} a solução ótima de (5.7) e z^i o vetor objetivo associado
 - 16: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{*i}\}$
 - 17: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{(z_1^i, z_2^i)^T\}$
 - 18: Organizamos, segundo os valores de z_1 , as soluções eficientes já encontradas
 - 19: **end if**
 - 20: **end for**
 - 21: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

Ilustramos a aplicação deste procedimento com um exemplo.

Exemplo 5.1. Considere o seguinte problema biobjetivo inteiro:

$$\begin{aligned}
 & \text{Minimize} && z_1 = -5x_1 + 2x_2 \\
 & \text{Minimize} && z_2 = x_1 - 4x_2 \\
 & \text{sujeito a} && -x_1 + x_2 \leq 3 \\
 & && x_1 \leq 6 \\
 & && x_1 + x_2 \leq 8 \\
 & && x_2 \leq 4 \\
 & && x_1 \in \mathbb{N}, \quad x_2 \in \mathbb{N},
 \end{aligned}$$

e o conjunto de todas as soluções eficientes ilustrado na Figura (33), com a presença de soluções suportadas e não suportadas.

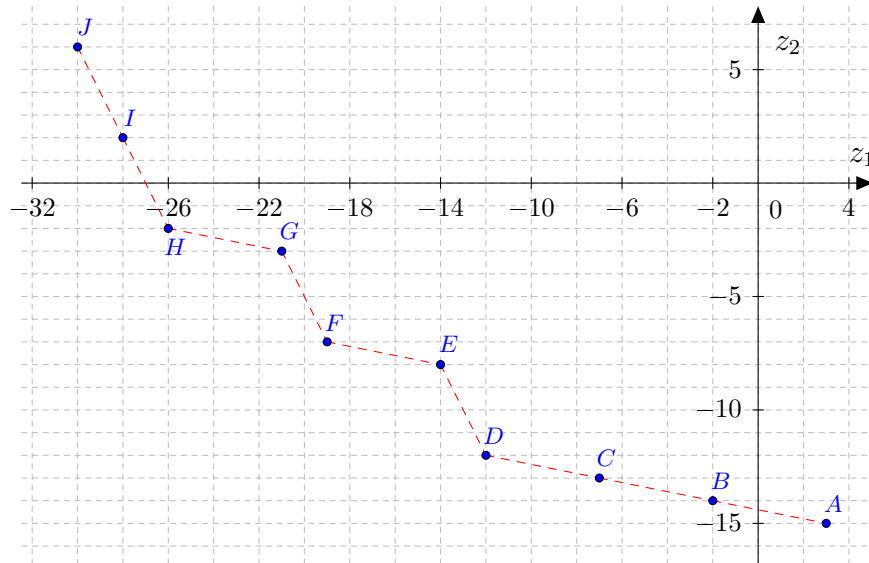


Figura 33 – Fronteira de Pareto para o problema do Exemplo 5.3.4

Seja $\Delta = 0.1$.

- **Cálculo dos lexicográficos:** otimizamos os objetivos separadamente para obter os pontos $A = (3, -15)^T$ e $J = (-30, 6)^T$. Renomeamos como $S^1 = J$, $S^2 = A$ e $t = 2$.
- **Iteração 1:** seja $i = 1$, nesse caso $R = (-30, -15)^T$, $N = (3, 6)^T$ e resolvemos o problema de Tchebycheff:

$$\begin{aligned}
 &\text{Minimize} && z^1 = u + 0.1 \cdot [-5x_1 + 2x_2 - (-29.9) + x_1 - 4x_2 - (-14.9)] \\
 &\text{sujeito a} && -x_1 + x_2 \leq 3 \\
 &&& x_1 \leq 6 \\
 &&& x_1 + x_2 \leq 8 \\
 &&& x_2 \leq 4 \\
 &&& -5x_1 + 2x_2 \leq 2.9 \\
 &&& x_1 - 4x_2 \leq 5.9 \\
 &&& -5x_1 + 2x_2 - (-29.9) \leq u \\
 &&& x_1 - 4x_2 - (-14.9) \leq u \\
 &&& x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

cuja solução é o vetor objetivo $F = (-19, -7)^T$ e $t = 3$.

- **Iteração 2:** fazamos $i = 1$. Organizamos os pontos não dominados encontrados, isto é, $S^1 = (-30, 6)^T$, $S^2 = (-19, -7)^T$, $S^3 = (3, -15)^T$. Consideramos os pontos consecutivos S^1 e S^2 , assim $R = (-30, -7)^T$ e $N = (-19, 6)^T$. Resolvemos o

problema:

$$\begin{aligned}
 \text{Minimize } z^1 &= u + 0.1 \cdot [-5x_1 + 2x_2 - (-29.9) + x_1 - 4x_2 - (-6.9)] \\
 \text{sujeito a } &-x_1 + x_2 \leq 3 \\
 &x_1 \leq 6 \\
 &x_1 + x_2 \leq 8 \\
 &x_2 \leq 4 \\
 &-5x_1 + 2x_2 \leq -19.1 \\
 &x_1 - 4x_2 \leq 5.9 \\
 &-5x_1 + 2x_2 - (-29.9) \leq u \\
 &x_1 - 4x_2 - (-6.9) \leq u \\
 &x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

cuja solução é o vetor objetivo $H = (-26, -2)^T$. Agora $i = 2$ e considerando as soluções eficientes consecutivas S^2 e S^3 , temos outro ponto de referência, a saber, $R = (-19, -15)^T$ e $N = (3, -7)^T$ e o problema:

$$\begin{aligned}
 \text{Minimize } z^2 &= u + 0.1 \cdot [-5x_1 + 2x_2 - (-18.9) + x_1 - 4x_2 - (-14.9)] \\
 \text{sujeito a } &-x_1 + x_2 \leq 3 \\
 &x_1 \leq 6 \\
 &x_1 + x_2 \leq 8 \\
 &x_2 \leq 4 \\
 &-5x_1 + 2x_2 \leq 2.9 \\
 &x_1 - 4x_2 \leq -7.1 \\
 &-5x_1 + 2x_2 - (-18.9) \leq u \\
 &x_1 - 4x_2 - (-14.9) \leq u \\
 &x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

a ser resolvido. Sua solução fornece o vetor objetivo $C = (-7, -13)^T$ e $t = 5$.

- **Iteração 3:** organizamos as soluções eficientes encontradas, isto é, $S^1 = (-30, 6)^T$, $S^2 = (-26, -2)^T$, $S^3 = (-19, -7)^T$, $S^4 = (-7, -13)^T$ e $S^5 = (3, -15)^T$. Agora seja $i = 1$, logo $R = (-30, -2)^T$, $N = (-26, 6)^T$, e o novo subproblema:

$$\begin{aligned}
 \text{Minimize } z^1 &= u + 0.1 \cdot [-5x_1 + 2x_2 - (-29.9) + x_1 - 4x_2 - (-1.9)] \\
 \text{sujeito a } &-x_1 + x_2 \leq 3 \\
 &x_1 \leq 6 \\
 &x_1 + x_2 \leq 8 \\
 &x_2 \leq 4 \\
 &-5x_1 + 2x_2 \leq -26.1 \\
 &x_1 - 4x_2 \leq 5.9 \\
 &-5x_1 + 2x_2 - (-29.9) \leq u \\
 &x_1 - 4x_2 - (-1, 9) \leq u \\
 &x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

fornece uma solução cujo vetor objetivo igual à $I = (-28, 2)^T$. Agora $i = 2$, tomando como referência os pontos $R = (-26, -7)^T$ e $N = (-19, 2)^T$ o subproblema:

$$\begin{aligned} \text{Minimize} \quad & z^2 = u + 0.1 \cdot [-5x_1 + 2x_2 - (-25.9) + x_1 - 4x_2 - (-6.9)] \\ \text{sujeito a} \quad & -x_1 + x_2 \leq 3 \\ & x_1 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_2 \leq 4 \\ & -5x_1 + 2x_2 \leq -19.1 \\ & x_1 - 4x_2 \leq 1.9 \\ & -5x_1 + 2x_2 - (-25.9) \leq u \\ & x_1 - 4x_2 - (-6, 9) \leq u \\ & x_1 \in \mathbb{N}, \quad x_2 \in \mathbb{N}, \quad u \geq 0, \end{aligned}$$

dá como valor objetivo o vetor $G = (-21, -3)^T$. Prosseguindo, $i = 3$, novos pontos de referência são considerados, a saber, $R = (-19, -13)^T$ e $N = (-7, -7)^T$, recaindo no novo subproblema:

$$\begin{aligned} \text{Minimize} \quad & z^3 = u + 0.1 \cdot [-5x_1 + 2x_2 - (-18.9) + x_1 - 4x_2 - (-12.9)] \\ \text{sujeito a} \quad & -x_1 + x_2 \leq 3 \\ & x_1 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_2 \leq 4 \\ & -5x_1 + 2x_2 \leq -7.1 \\ & x_1 - 4x_2 \leq -7.1 \\ & -5x_1 + 2x_2 - (-18, 9) \leq u \\ & x_1 - 4x_2 - (-12, 9) \leq u \\ & x_1 \in \mathbb{N}, \quad x_2 \in \mathbb{N}, \quad u \geq 0, \end{aligned}$$

e terá como solução um vetor cujo objetivo associado é o ponto $E = (-14, -8)^T$. Assim, $i = 4$ e os novos pontos de referências são $R = (-7, -15)^T$ e $N = (3, -13)^T$ gerando o último subproblema desta iteração:

$$\begin{aligned} \text{Minimize} \quad & z^4 = u + 0.1 \cdot [-5x_1 + 2x_2 - (-6.9) + x_1 - 4x_2 - (-14.9)] \\ \text{sujeito a} \quad & -x_1 + x_2 \leq 3 \\ & x_1 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_2 \leq 4 \\ & -5x_1 + 2x_2 \leq 2.9 \\ & x_1 - 4x_2 \leq -13.1 \\ & -5x_1 + 2x_2 - (-6, 9) \leq u \\ & x_1 - 4x_2 - (-14, 9) \leq u \\ & x_1 \in \mathbb{N}, \quad x_2 \in \mathbb{N}, \quad u \geq 0, \end{aligned}$$

que dá como vetor objetivo o ponto $B = (-2, -14)^T$. Até aqui, $t = 9$.

- **Iteração 3:** faça $t = 1$ e organize todas as soluções encontradas até esta iteração, isto é, $S^1 = (-30, 6)^T$, $S^2 = (-28, 2)^T$, $S^3 = (-26, -2)^T$, $S^4 = (-21, -3)^T$, $S^5 = (-19, -7)^T$, $S^6 = (-14, -8)^T$, $S^7 = (-7, -13)^T$, $S^8 = (-2, -14)^T$ e $S^9 = (3, -15)^T$. Para $i = 1$, temos como referência os pontos $R = (-30, 2)^T$ e $N = (-28, 6)^T$ e resulta num problema:

$$\begin{aligned}
 \text{Minimize } z^1 &= u + 0.1 \cdot [-5x_1 + 2x_2 - (-29.9) + x_1 - 4x_2 - (-1.9)] \\
 \text{sujeito a } & -x_1 + x_2 \leq 3 \\
 & x_1 \leq 6 \\
 & x_1 + x_2 \leq 8 \\
 & x_2 \leq 4 \\
 & -5x_1 + 2x_2 \leq 5.9 \\
 & x_1 - 4x_2 \leq -28.1 \\
 & -5x_1 + 2x_2 - (-29, 9) \leq u \\
 & x_1 - 4x_2 - (-1, 9) \leq u \\
 & x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

infectível. Logo $k = 1$. O mesmo acontece para $i = \{2, 3, 4, 5\}$ e assim $k = 5$ problemas infectíveis. Para $i = 6$, o subproblema tem como referência os pontos $R = (-14, -13)^T$ e $N = (-7, -8)^T$ cujo subproblema associado é:

$$\begin{aligned}
 \text{Minimize } z^6 &= u + 0.1 \cdot [-5x_1 + 2x_2 - (-13.9) + x_1 - 4x_2 - (-12.9)] \\
 \text{sujeito a } & -x_1 + x_2 \leq 3 \\
 & x_1 \leq 6 \\
 & x_1 + x_2 \leq 8 \\
 & x_2 \leq 4 \\
 & -5x_1 + 2x_2 \leq -7.1 \\
 & x_1 - 4x_2 \leq -8.1 \\
 & -5x_1 + 2x_2 - (-13, 9) \leq u \\
 & x_1 - 4x_2 - (-12, 9) \leq u \\
 & x_1 \in \mathbb{N}, x_2 \in \mathbb{N}, u \geq 0,
 \end{aligned}$$

fornece o vetor $D = (-12, -12)^T$ como solução. Prosseguindo nesta terceira iteração para $i = \{7, 8, 9\}$ temos problemas infectíveis, logo $k = 8 < 9 = t - 1$, e assim temos mais uma iteração a realizar. Atualizamos $t = 10$.

- Ordenamos todas as soluções eficientes encontradas, a saber, $S^1 = (-30, 6)^T$, $S^2 = (-28, 2)^T$, $S^3 = (-26, -2)^T$, $S^4 = (-21, -3)^T$, $S^5 = (-19, -7)^T$, $S^6 = (-14, -8)^T$, $S^7 = (-12, -12)^T$, $S^8 = (-7, -13)^T$, $S^9 = (-2, -14)^T$ e $S^{10} = (3, -15)^T$. Os subproblemas para $i = \{1, 2, 3, 4, 5\}$ associados nesta iteração já foram checados na

iteração anterior e foi constatado que eram *infectíveis* ($\Rightarrow k = 5$). Resta analisar os subproblemas $i = 6$, cuja referências são $R = (-14, -12)^T$, $N = (-12, -8)^T$ e $i = 7$ de referência $R = (-12, -13)^T$ e $N = (-13, -12)^T$. Estes, por sua vez, são *infectíveis*, e portanto $k \leftarrow k + 2$. Na iteração anterior foi averiguado que os subproblemas i , com $i = \{8, 9\}$ também não admitem solução, logo $k = 9 = t - 1$ e isto fornece o critério de parada para este procedimento, com todas as soluções eficientes encontradas.

A eficiência e robustez deste método é ilustrada mediante os extensos experimentos computacionais ilustrados com detalhes no Capítulo 6.

5.4 O Método ε —Restrito

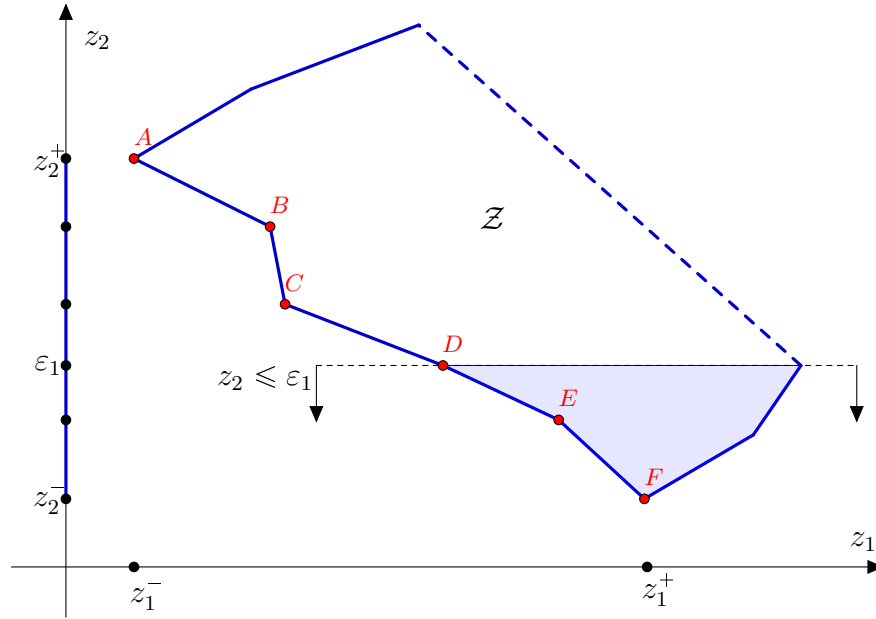
Nesta abordagem, escolhemos como função-objetivo para os subproblemas uma das funções que definem o problema multiobjetivo. No PCUIM, foi mais conveniente considerar como função-objetivo para o subproblema restrito a função z_1 , e como restrição a função z_2 , pois sua amplitude é muito menor se comparada com z_1 . O Problema Restrito com a imposição de um *setup* de até ε é definido a seguir, onde $\rho > 0$:

$$\begin{aligned} \text{Minimize} \quad & z^\varepsilon = z_1 + \rho \cdot z_2 \\ \text{sujeito a} \quad & x \in \mathcal{X} \\ & z_2 \leq \varepsilon. \end{aligned} \tag{5.8}$$

Vários subproblemas escalares (5.8) são resolvidos sequencialmente para cada valor de ε , originando diferentes soluções eficientes. A cada subproblema, uma nova solução eficiente x^ε com *setup* de até ε é obtida. A vantagem desta técnica é a simplicidade de implementação e a geração de pontos suportados e não suportados na fronteira de Pareto, isto é, o procedimento consegue determinar *todas* as soluções eficientes para o PCUIM.

Como a segunda função-objetivo para o PCUIM toma apenas valores inteiros num intervalo fechado, é razoável considerar os valores inteiros de ε variando em $\mathcal{I} = [z_2^-, z_2^+]$, onde z_2^- e z_2^+ é o menor e o maior *setup* que o PCUIM possui, respectivamente. A cada valor de ε inteiro no intervalo \mathcal{I} , uma nova solução eficiente é obtida ao se resolver o Problema de Programação Linear Inteiro (5.8).

A Figura (34) ilustra a ideia deste método para um valor específico para $\varepsilon = \varepsilon_1$ e uma solução eficiente cuja imagem no espaço de critério é o ponto não dominado D . Conforme o valor do lado direito da restrição é alterado para outro valor inteiro, o hiperplano que define esta restrição é deslocado verticalmente e novas soluções eficientes serão obtidas quando a função z_1 é minimizada.


 Figura 34 – ε –Restrito no Problema de Corte multiobjetivo para $\varepsilon = \varepsilon_1$

O método inicia computando os pontos lexicográficos do problema em questão, ao resolver (5.1) e (5.2), e assim, duas soluções eficientes são obtidas. Em seguida, resolvemos uma sequência de subproblemas restritos, iniciando-se com $\varepsilon = z_2^- + \Delta$ e incrementando-o de Δ unidades até ε atinja seu valor máximo permitido, $z_2^+ - \Delta$. Com ε fora do intervalo \mathcal{I} , o subproblema ou fornecerá uma solução eficiente já encontrada (no caso se $\varepsilon > z_2^+$) ou será infactível (caso $\varepsilon < z_2^-$).

Algoritmo 7: O Método ε –Restrito para o PCUIM

- 1: Entrada: dados do Problema de Corte e Δ
 - 2: $\mathcal{X}^* = \{x^{*1}, x^{*2}\}$
 - 3: $\mathcal{Z}^* = \{(z_1^-, z_2^+)^T, (z_1^+, z_2^-)^T\}$
 - 4: $\varepsilon = z_2^- + \Delta$
 - 5: **while** $\varepsilon \leq z_2^+ - \Delta$ **do**
 - 6: Resolvemos o Problema (5.8). Sejam $x^{\varepsilon*}$ a solução ótima e z^ε o vetor objetivo associado
 - 7: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{\varepsilon*}\}$
 - 8: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{(z_1^\varepsilon, z_2^\varepsilon)^T\}$
 - 9: $\varepsilon \leftarrow \varepsilon + \Delta$
 - 10: **end while**
 - 11: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

Um pseudocódigo deste algoritmo foi organizado anteriormente, onde o parâmetro Δ é fornecido pelo usuário. Esse parâmetro controla a precisão da fronteira a ser encontrada. No entanto, quando não for especificado, $\Delta = 1$ para o PCUIM original que possui os coeficientes na função-objetivo todos iguais à 1. Apenas na Seção 6.4

modificaremos esse valor.

5.5 O Método NISE Híbrido com ε –Restrito

O método NISE híbrido com ε –Restrito é uma técnica alternativa para gerar todas as soluções eficientes para o problema considerado.

Como descrito no capítulo anterior, o método NISE é um tipo de Soma Ponderada especificamente desenvolvida para detectar todas as bases eficientes para um Problema de Programação Linear biobjetivo. No presente estudo ele é utilizado e adaptado em um problema de variáveis inteiras e então apenas soluções eficientes suportadas poderão ser detectadas. Em um problema combinatório, é bem sabido que a fronteira de Pareto pode conter muitos pontos não suportados, logo a aplicação pura do procedimento NISE pode não ser muito conveniente em problemas desta natureza.

A fim de contornar este inconveniente, pensamos em obter as soluções não suportadas utilizando o método ε –Restrito, onde o lado direito da restrição $z_2 \leq \varepsilon$ do Problema Restrito toma valores inteiros que ainda não foram atingidos pelo NISE. Como a função z_2 apenas toma valores inteiros no intervalo \mathcal{I} , é relativamente fácil descobrir os “saltos” na projeção do espaço de critério no eixo z_2 espaço de critério após a aplicação do NISE. Sendo assim, o ε –Restrito será aplicado nesses saltos, e ao final do procedimento, o conjunto de todas as soluções eficientes serão obtidas.

Uma outra motivação para o uso de um procedimento híbrido é a dificuldade que pode ser ocasionada com a resolução dos Problemas Restritos, como retrata o trabalho apresentado em (EHRGOTT; RUZIKA, 2008). Segundo estes autores, a restrição $z_2 \leq \varepsilon$ (uma restrição do tipo “mochila”) destrói a estrutura poliedral de \mathcal{X} e torna a técnica de particionamento de conjuntos no *Branch* ineficiente. Além disso, segundo estes autores, os limitantes para a função-objetivo “enganam” os subproblemas geração de colunas, na obtenção de colunas benéficas, fazendo com que se tenha um grande número de iterações com pouco progresso na função-objetivo. No PCUIM, estes inconvenientes podem ser reduzidos, fazendo com que Problema Restrito seja aplicado em um menor número de vezes, isto é, apenas na geração das soluções não suportadas, enquanto os pontos suportados serão obtidos pelo NISE, que utiliza uma função ponderada e que não afeta a estrutura poliédrica do conjunto \mathcal{X} .

Apresentamos um pseudocódigo para este método, onde ε toma apenas valores

inteiros do conjunto \mathcal{J} , como definido a seguir.

Algoritmo 8: O Método NISE híbrido com ε -Restrito para o PCUIM

- 1: Entrada: dados do Problema de Corte e ρ
 - 2: Aplique o método NISE no PCUIM
 - 3: Sejam os pontos lexicográficos $A = (z_1^-, z_2^+)^T$, $B = (z_1^+, z_2^-)^T$ e o conjunto dos pontos inteiros no intervalo $\mathcal{II} = [z_2^-, z_2^+]$
 - 4: Sejam o conjunto \mathcal{N} das t soluções não suportadas obtidas: $\mathcal{N} = \{x^{*1}, \dots, x^{*t}\}$ e os vetores objetivos associados $f(\mathcal{N}) = \{(z_1^1, z_2^1)^T, \dots, (z_1^t, z_2^t)^T\}$ e colocados em ordem crescente segundo os valores de z_1
 - 5: $\mathcal{X}^* = \mathcal{N}$
 - 6: $\mathcal{Z}^* = f(\mathcal{N})$
 - 7: Façamos $\mathcal{J} = \mathcal{I} - \{z_2^1, \dots, z_2^t\}$
 - 8: **if** $\mathcal{J} \neq \emptyset$ **then**
 - 9: **for** $\varepsilon \in \mathcal{J}$ **do**
 - 10: Resolvemos o problema (5.8). Sejam $x^{*\varepsilon}$ sua solução ótima e z^ε seu vetor objetivo associado
 - 11: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{*\varepsilon}\}$
 - 12: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{(z_1^\varepsilon, z_2^\varepsilon)^T\}$
 - 13: **end for**
 - 14: **end if**
 - 15: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

5.6 O Método Benson

Este método foi descrito no capítulo anterior e foi demonstrado que, dada uma solução factível x^0 , o Problema de Benson associado:

$$\begin{aligned}
 &\text{Maximize} && z_0 = l_1 + l_2 \\
 &\text{sujeito a} && x \in \mathcal{X} \\
 & && z_1^0 - z_1 = l_1 \\
 & && z_2^0 - z_2 = l_2 \\
 & && l_1, l_2 \geq 0,
 \end{aligned} \tag{5.9}$$

fornece uma solução x^* eficiente.

Este subproblema é ligeiramente modificado em relação ao original para se adequar às características do PCUIM e ser capaz de determinar todas as soluções eficientes para o problema considerado. Da forma como foi definido o subproblema de Benson (5.9), soluções factíveis distintas podem originar nas mesmas soluções eficientes. Além disso, não há nenhuma garantia de se obter um conjunto eficiente bem distribuído se variarmos o ponto inicial z^0 , a imagem de x^0 no espaço objetivo.

Para tornar esta técnica de escalarização aplicável no PCUIM, modificamos o subproblema de Benson. Em vez de maximizar a soma $l_1 + l_2$, maximizamos $l_1 + \rho \cdot l_2$. A parcela $\rho \cdot l_2$ é para evitar ótimos alternativos, e conseqüentemente soluções fracamente eficientes. Sob certas condições do ponto inicial z^0 , o subproblema modificado é capaz de enumerar todas as soluções eficientes para o PCUIM. A fundamentação teórica desta mudança é garantida pelo teorema a seguir, de nossa autoria.

Teorema 5.4. *Considere um ponto factível x^0 , cujo valor objetivo seja dado pelo vetor $z^0 = (z_1^0, z_2^0)^T$. Suponhamos que este ponto factível satisfaça $z_2^- \leq z_2^0 \leq z_2^+$. Nesse caso, o Problema de Benson Modificado, dado por:*

$$\begin{aligned} \text{Maximize} \quad & z_0 = l_1 + \rho \cdot l_2 \\ \text{sujeito a} \quad & x \in \mathcal{X} \\ & z_1^0 - z_1 = l_1 \\ & z_2^0 - z_2 = l_2 \\ & l_1, l_2 \geq 0, \end{aligned} \tag{5.10}$$

e fornece uma solução (x^*, l^*) tal que x^* é eficiente.

Demonstração. Seja $\rho = 0$ e assumamos que (5.10) tenha solução única. Suponhamos que x^* , de valor objetivo $z^* = (z_1^*, z_2^*)^T$, não seja eficiente e que $z_2^- \leq z_2^0 \leq z_2^+$. Então existe uma outra solução $x' \in \mathcal{X}$, de valor objetivo $z' = (z_1', z_2')^T$, tal que uma das duas condições é sempre verdadeira:

1. $z_1' < z_1^*$ e $z_2' \leq z_2^*$ ou
2. $z_1' \leq z_1^*$ e $z_2' < z_2^*$.

Definimos $l' = z^0 - z'$. Em ambas condições, temos que (x', l') é factível, pois

$$l' = z^0 - z' \geq z^0 - z^* = l^* \geq 0.$$

Agora, usando o caso 1, temos que:

$$l'_1 = z_1^0 - z_1' \geq z_1^0 - z_1^* = l_1^* \geq 0$$

e

$$l'_2 = z_2^0 - z_2' > z_2^0 - z_2^* = l_2^* \geq 0,$$

isto é, $l'_1 \geq l_1^*$ e nesse caso $l'_1 \neq l_1^*$, que é uma contradição, pois o problema (5.10) possui solução única devido à parcela $\rho \cdot l_2$. Agora, se o caso 2 é verdadeiro, temos que:

$$l'_1 = z_1^0 - z_1' > z_1^0 - z_1^* = l_1^* \geq 0$$

e

$$l'_2 = z_2^0 - z_2' \geq z_2^0 - z_2^* = l_2^* \geq 0,$$

Esse procedimento é repetido para os demais pontos não dominados à medida que vão sendo encontrados, sempre utilizando como guia o ponto encontrado na iteração anterior para construir o próximo factível em referência. O valor de Δ quando seu valor não for especificado é 1.

Esta ideia é organizada em um algoritmo, capaz de obter todas as soluções eficientes para o PCUIM e apresentado a seguir.

Algoritmo 9: O Método de Benson para o PCUIM

- 1: Entrada: dados do Problema de Corte e ρ
 - 2: $\mathcal{X}^* = \{x^{*1}, x^{*2}\}$
 - 3: $\mathcal{Z}^* = \{(z_1^-, z_2^+)^T, (z_1^+, z_2^-)^T\}$
 - 4: $k = 1$
 - 5: $\varepsilon = z_2^- + \Delta$
 - 6: **while** $\varepsilon \leq z_2^+ - \Delta$ **do**
 - 7: Sejam a solução inicial x^k cuja imagem é o vetor $z^k = (z_1^k, z_2^k)^T = (z_1^+ - k, \varepsilon)^T$
 - 8: Resolvemos o Problema (5.10). Sejam x^{*k} sua solução e z^{k*} o vetor objetivo associado
 - 9: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{*k}\}$
 - 10: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{z^{k*}\}$
 - 11: $k \leftarrow k + 1$
 - 12: $\varepsilon \leftarrow \varepsilon + \Delta$
 - 13: **end while**
 - 14: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

Numa primeira etapa, resolvemos os Problemas (5.1) e (5.2) e diretamente temos duas soluções x^{*1} e x^{*2} eficientes. Esta fase inicial é importante, no sentido que delimita o número de iterações que o procedimento utilizará. Na etapa k , o Algoritmo (9) toma um ponto z^k , como descrito no parágrafo anterior e resolve o Subproblema (5.10) para obter a solução eficiente x^{*k} . Ao final das $(z_2^+ - z_2^- - 2) \cdot \Delta$ iterações, encontramos todas as soluções eficientes para o PCUIM.

Exemplo 5.2. *Considere o novamente o problema:*

$$\begin{aligned}
 & \text{Minimize} && z = (-5x_1 + 2x_2, x_1 - 4x_2) \\
 & \text{sujeito a} && -x_1 + x_2 \leq 3 \\
 & && x_1 \leq 6 \\
 & && x_1 + x_2 \leq 8 \\
 & && x_2 \leq 4 \\
 & && x_1 \in \mathbb{N}, x_2 \in \mathbb{N},
 \end{aligned}$$

Considerando o Subproblema de Benson usual (3.12), salientamos algumas dificuldades que se podem ter com este método. Se por exemplo, $z^0 = (0, 0)^T$, subproblema

de Benson teria a seguinte solução: $(x^*, l^*) = (6, 2, 26, 2)^T$. Isto implica que $x^* = (6, 2)^T$ é eficiente para o problema biobjetivo. Tomando uma outra solução inicial, a saber, $z^0 = (1, 2)^T$, o mesmo ponto correspondente no espaço de critério seria obtido. Por outro lado, se $x^0 = (5, 3)^T$ o subproblema forneceria a solução ótima igual a $(x^*, l^*) = (5, 3, 0, 0)^T$, isto é, os desvios seriam nulos. Isto implica que esta solução inicial é eficiente pelo Teorema (3.15).

Considerando a formulação modificada, dado $x^0 = (x_1^0, x_2^0)^T$ respeitando as restrições do problema inicial, o problema de Benson Modificado (com $\rho = 0.1$) seria dada por:

$$\begin{aligned} \text{Maximize} \quad & z_0 = l_1 + 0.1 \cdot l_2 \\ \text{sujeito a} \quad & -x_1 + x_2 \leq 3 \\ & x_1 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_2 \leq 4 \\ & l_1 - 5x_1 + 2x_2 = z_1^0 \\ & l_2 + x_1 - 4x_2 = z_2^0 \\ & x_1 \in \mathbb{N}, x_2 \in \mathbb{N} \\ & l_1 \geq 0, l_2 \geq 0. \end{aligned}$$

Como feito anteriormente, os pontos lexicográficos deste problema são dados por $A = (3, -15)^T$ e $J = (-30, 6)^T$. Para determinar as demais soluções eficientes, seja $z^0 = (3 - \Delta, -15 + \Delta)^T = (2, -14)^T$, onde $\Delta = 1$, pois o problema tem coeficientes inteiros, logo todos os pontos não dominados terão componentes inteiras. O subproblema resultante:

$$\begin{aligned} \text{Maximize} \quad & z_0 = l_1 + 0,1 \cdot l_2 \\ \text{sujeito a} \quad & -x_1 + x_2 \leq 3 \\ & x_1 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_2 \leq 4 \\ & l_1 - 5x_1 + 2x_2 = 2 \\ & l_2 + x_1 - 4x_2 = -14 \\ & x_1 \in \mathbb{N}, x_2 \in \mathbb{N} \\ & l_1 \geq 0, l_2 \geq 0, \end{aligned}$$

fornecerá como solução eficiente $x^{*1} = (2, 4, 4, 0)^T$ cujo vetor objetivo é $z^1 = (-2, 14)$. O próximo subproblema utilizará o desvio em relação ao ponto $z^0 = (-2 - \Delta, 14 + \Delta)^T =$

$(-3, 15)^T$, isto é, otimizaremos o novo problema

$$\begin{aligned}
 & \text{Maximize} && z_0 = l_1 + 0,1 \cdot l_2 \\
 & \text{sujeito a} && -x_1 + x_2 \leq 3 \\
 & && x_1 \leq 6 \\
 & && x_1 + x_2 \leq 8 \\
 & && x_2 \leq 4 \\
 & && l_1 - 5x_1 + 2x_2 = -3 \\
 & && l_2 + x_1 - 4x_2 = 15 \\
 & && x_1 \in \mathbb{N}, x_2 \in \mathbb{N} \\
 & && l_1 \geq 0, l_2 \geq 0,
 \end{aligned}$$

e assim por diante, até que o valor da segunda coordenada de z^0 fique igual à $z_2^+ - \Delta$. Ao final de todas as iterações, temos todas as soluções eficientes.

5.7 O Método de Silva & Crema

Este método foi empregado para resolver o PCUIM. É um método que difere dos demais vistos até aqui e servirá como meio de comparação com os métodos desenvolvidos/adaptados até então.

Neste procedimento, a desvantagem é que a cada solução eficiente encontrada o subproblema subsequente terá duas variáveis binárias e duas restrições adicionais, visando eliminar a região do espaço de critério que é dominada por todos os pontos já encontrados. Essa característica pode fazer com que a formulação matemática para estes subproblemas fique fraca, gerando dificuldades no *Branch-and-Bound*.

Para um melhor entendimento deste procedimento, ilustramos com uma aplicação num PCUIM.

Exemplo 5.3. Considere o PCUIM biobjetivo com 6 itens:

$$\begin{aligned}
 & \text{Minimize} && z = (\|x\|_1, \|y\|_1) \\
 & \text{sujeito a} && Ax \geq d, \\
 & && x \leq N \cdot y, \\
 & && x \geq y, \\
 & && y \in \mathbb{B}^6, x \in \mathbb{N}^6,
 \end{aligned}$$

os padrões de corte já fornecidos pela matriz A com a demanda d , onde:

$$A = \begin{bmatrix} 0 & 4 & 0 & 7 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 6 \\ 0 & 4 & 4 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7 & 0 \\ 4 & 0 & 0 & 0 & 0 & 4 \end{bmatrix} \quad e \quad d = \begin{bmatrix} 5 \\ 8 \\ 6 \\ 7 \\ 7 \\ 7 \end{bmatrix}.$$

Seja $w = (1/4, 3/4)^T$. Resolva o problema:

$$\begin{aligned} \text{Minimize} \quad & z = \frac{1}{4} \cdot \|x\|_1 + \frac{3}{4} \cdot \|y\|_1 \\ \text{sujeito a} \quad & Ax \geq d, \\ & x \leq N \cdot y, \\ & x \geq y, \\ & y \in \mathbb{B}^6, x \in \mathbb{N}^6, \end{aligned}$$

onde $N = (2, 2, 2, 1, 1, 2)^T$ e cuja solução é o vetor não dominado $z^w = (5, 5)^T$.

- **Iteração 1:** Retire do conjunto \mathcal{X} todos os pontos que são dominados pelo vetor $z^w = (5, 5)^T$. Consideremos os dois limitantes para a função-objetivo do problema considerado: $M_1 = \sum_{i=1}^6 d_i = 40$ e $M_2 = 6$. Como $l = 1$ (descobrimos apenas uma solução eficiente) o problema resultante a ser resolvido será dado por:

$$\begin{aligned} \text{Minimize} \quad & z = \frac{1}{4} \cdot \|x\|_1 + \frac{3}{4} \cdot \|y\|_1 \\ \text{sujeito a} \quad & x \in \mathcal{X}, \\ & \|x\|_1 \leq (5 - 1) \cdot t_1^1 + 40 \cdot (1 - t_1^1), \\ & \|y\|_1 \leq (5 - 1) \cdot t_2^1 + 6 \cdot (1 - t_2^1), \\ & t_1^1 + t_2^1 \geq 1, \\ & t_1^1, t_2^1 \in \mathbb{B}, \end{aligned}$$

cuja solução ótima fornece $t_2^1 = 0$ (notamos que se $t_2^1 = 1$, o problema seria infactível), então $t_1^1 = 1$ e $z^1 = (6, 4)^T$. A Figura (36) ilustra a região hachurada no espaço objetivo que foi excluída, que é o conjunto dos pontos do espaço de critério dominados pelo vetor $z^w = (5, 5)^T$; também é ilustrado a curva de nível da função ponderada tangenciando o espaço critério no ponto z^w .

- **Iteração 2:** O próximo problema possui a mesma região admissível do problema anterior acrescida das restrições que eliminam os pontos que são dominados pelo

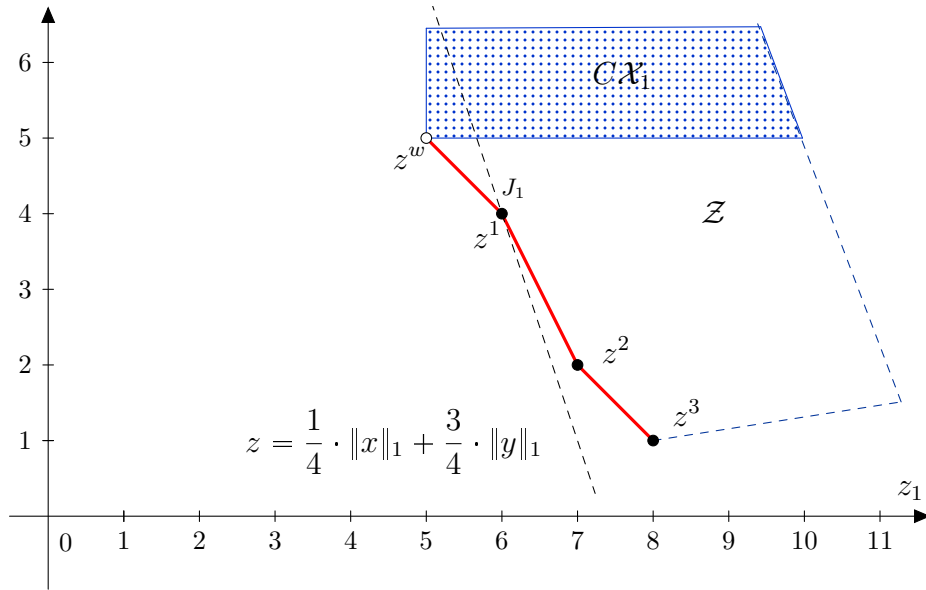


Figura 36 – Ilustração de pontos dominados no primeiro subproblema

vetor $z^1 = (6, 4)^T$. Isto resulta no seguinte subproblema:

$$\begin{aligned}
 & \text{Minimize} && z = \frac{1}{4} \cdot \|x\|_1 + \frac{3}{4} \cdot \|y\|_1 \\
 & \text{sujeito a} && x \in \mathcal{X}, \\
 & && \|x\|_1 \leq (5 - 1) \cdot t_1^1 + 40 \cdot (1 - t_1^1), \\
 & && \|y\|_1 \leq (5 - 1) \cdot t_2^1 + 6 \cdot (1 - t_2^1), \\
 & && \|x\|_1 \leq (6 - 1) \cdot t_1^2 + 40 \cdot (1 - t_1^2), \\
 & && \|y\|_1 \leq (4 - 1) \cdot t_2^2 + 6 \cdot (1 - t_2^2), \\
 & && t_1^1 + t_2^1 \geq 1, \\
 & && t_1^2 + t_2^2 \geq 1, \\
 & && t_1^1, t_2^1, t_1^2, t_2^2 \in \mathbb{B},
 \end{aligned}$$

e a solução é o vetor não dominado $z^2 = (7, 2)^T$ e $l = 3$. Note a exclusão da região admissível do problema original, os pontos que são dominados por z^w e z^1 e o ponto não dominado z^2 sendo obtido.

- **Iteração 3:** O próximo problema conserva as mesmas restrições do problema anterior e acrescenta desigualdades que eliminam os pontos que são dominados pelo vetor z^2 .

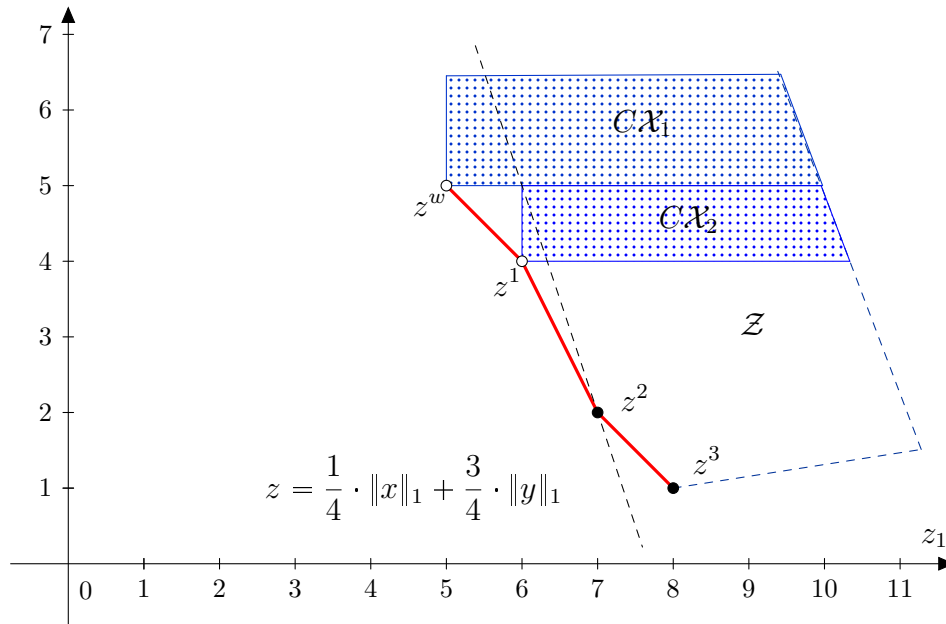


Figura 37 – Ilustração das soluções excluídas com o segundo subproblema

A formulação deste subproblema será dada por:

$$\begin{aligned}
 & \text{Minimize} && z = \frac{1}{4} \cdot \|x\|_1 + \frac{3}{4} \cdot \|y\|_1 \\
 & \text{sujeito a} && x \in \mathcal{X}, \\
 & && \|x\|_1 \leq (5 - 1) \cdot t_1^1 + 40 \cdot (1 - t_1^1), \\
 & && \|y\|_1 \leq (5 - 1) \cdot t_2^1 + 6 \cdot (1 - t_2^1), \\
 & && \|x\|_1 \leq (6 - 1) \cdot t_1^2 + 40 \cdot (1 - t_1^2), \\
 & && \|y\|_1 \leq (4 - 1) \cdot t_2^2 + 6 \cdot (1 - t_2^2), \\
 & && \|x\|_1 \leq (7 - 1) \cdot t_1^3 + 40 \cdot (1 - t_1^3), \\
 & && \|y\|_1 \leq (2 - 1) \cdot t_2^3 + 6 \cdot (1 - t_2^3), \\
 & && t_1^1 + t_2^1 \geq 1, \\
 & && t_1^2 + t_2^2 \geq 1, \\
 & && t_1^3 + t_2^3 \geq 1, \\
 & && t_1^1, t_2^1, t_1^2, t_2^2, t_1^3, t_2^3 \in \mathbb{B},
 \end{aligned}$$

e fornece como solução no espaço de critério o vetor $z^3 = (8, 1)^T$ e $l = 4$. A a Figura (5.3) ilustra a exclusão do conjunto $C\mathcal{X}_1 \cup C\mathcal{X}_2 \cup C\mathcal{X}_3$, onde C é uma matriz 2×6 de uns, restando como único ponto não dominado z^3 .

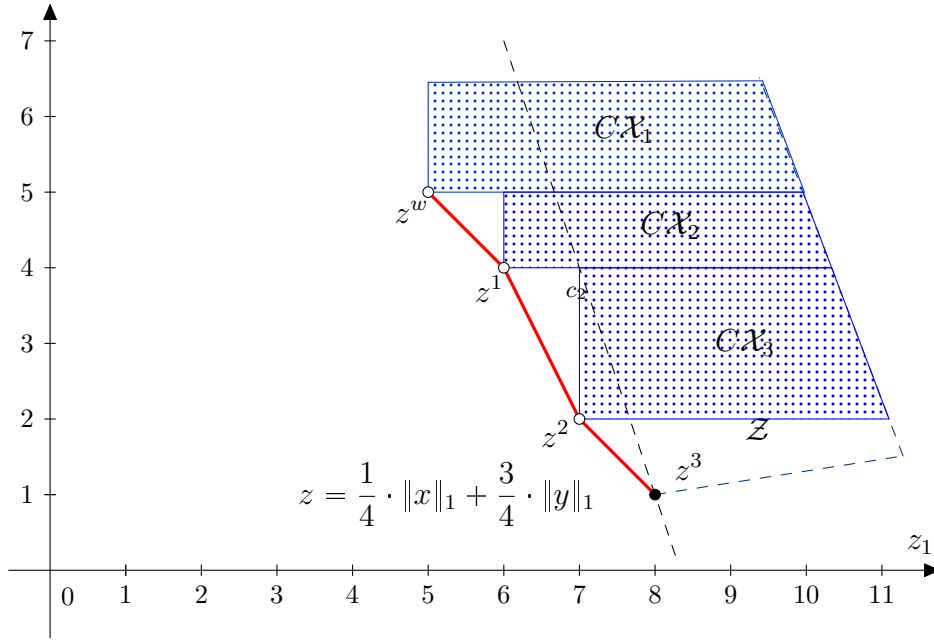


Figura 38 – Ilustração das soluções excluídas do terceiro subproblema

• **Iteração 5:** O próximo problema, será dado por:

$$\text{Minimize } z = \frac{1}{4} \cdot \|x\|_1 + \frac{3}{4} \cdot \|y\|_1$$

$$\text{sujeito a } x \in \mathcal{X},$$

$$\|x\|_1 \leq (5 - 1) \cdot t_1^1 + 40 \cdot (1 - t_1^1),$$

$$\|y\|_1 \leq (5 - 1) \cdot t_2^1 + 6 \cdot (1 - t_2^1),$$

$$\|x\|_1 \leq (6 - 1) \cdot t_1^2 + 40 \cdot (1 - t_1^2),$$

$$\|y\|_1 \leq (4 - 1) \cdot t_2^2 + 6 \cdot (1 - t_2^2),$$

$$\|x\|_1 \leq (7 - 1) \cdot t_1^3 + 40 \cdot (1 - t_1^3),$$

$$\|y\|_1 \leq (2 - 1) \cdot t_2^3 + 6 \cdot (1 - t_2^3),$$

$$\|x\|_1 \leq (8 - 1) \cdot t_1^4 + 40 \cdot (1 - t_1^4),$$

$$\|y\|_1 \leq (1 - 1) \cdot t_2^4 + 6 \cdot (1 - t_2^4),$$

$$t_1^1 + t_2^1 \geq 1,$$

$$t_1^2 + t_2^2 \geq 1,$$

$$t_1^3 + t_2^3 \geq 1,$$

$$t_1^4 + t_2^4 \geq 1,$$

$$t_k^s \in \mathbb{B},$$

$$k = 1, 2, \quad s = 1, \dots, 4,$$

que é infactível. Logo todas as soluções eficientes para este problema foram encontradas, cujo vetores objetivos são dados pelos seguintes pontos:

$$\mathcal{Z}^* = \{(8, 1)^T, (7, 2)^T, (6, 4)^T, (5, 5)^T\}.$$

Observe que este método pode determinar pontos não suportados.

A seguir ilustramos o pseudocódigo deste método para o PCUIM.

Algoritmo 10: O Método de Silva & Crema para o PCUIM

- 1: Entrada: dados do Problema de Corte e $w = (w_1, w_2)^T > 0$
- 2: Resolvemos o problema ponderado a seguir:

$$\begin{aligned} \text{Minimize} \quad & z^w = w_1 \cdot \|x\|_1 + w_2 \cdot \|y\|_1 \\ \text{sujeito a} \quad & x \in \mathcal{X}, \end{aligned}$$

e sejam x^{*w} e z^w a solução eficiente e o vetor objetivo associado

- 3: $\mathcal{X}^* = \{x^{*w}\}$
- 4: $\mathcal{Z}^* = \{z^w\}$
- 5: $l = 1$
- 6: **while** Enquanto o subproblema a seguir for factível **do**
- 7:

$$\begin{aligned} \text{Minimize} \quad & z^l = w_1 \cdot \|x\|_1 + w_2 \cdot \|y\|_1 \\ \text{sujeito a} \quad & x \in \mathcal{X}, \\ & z_1 \leq (z_1^s - 1) \cdot t_1^s + M_1 \cdot (1 - t_1^s), & s = 1, \dots, l, \\ & z_2 \leq (z_2^s - 1) \cdot t_2^s + M_2 \cdot (1 - t_2^s), & s = 1, \dots, l, \\ & t_1^s + t_2^s \geq 1, & s = 1, \dots, l, \\ & t_k^s \in \mathbb{B}, & k = 1, 2, \quad s = 1, \dots, l. \end{aligned}$$

- 8: Denotamos por x^{*l} e z^l a solução eficiente e o vetor objetivo associado
 - 9: $\mathcal{X}^* = \mathcal{X}^* \cup \{x^{*l}\}$
 - 10: $\mathcal{Z}^* = \mathcal{Z}^* \cup \{z^l\}$
 - 11: $l = l + 1$
 - 12: **end while**
 - 13: Saída: \mathcal{X}^* e \mathcal{Z}^*
-

A seguir, propomos um procedimento que utiliza os padrões de corte dados a priori (obtidos pelo Método de Gilmore e Gomory, por exemplo) e pode ser visualizado como uma técnica que, embora não garanta a fronteira ótima de Pareto, fornece uma razoável aproximação em um tempo computacional aceitável.

5.8 Relaxação com os padrões básicos

Este procedimento utiliza como base os p padrões de corte dados a priori para gerar uma aproximação para a fronteira de Pareto para o PCUIM. Consideramos a versão

matricial do problema (2.22):

$$\begin{aligned}
 &\text{Minimize} && z = (\|x\|_1, \|y\|_1) \\
 &\text{sujeito a} && A^*x \geq d, \\
 &&& x \leq N \cdot y, \\
 &&& x \geq y, \\
 &&& y \in \mathbb{B}^p, x \in \mathbb{N}^p,
 \end{aligned} \tag{5.11}$$

onde A^* é a matriz definida pelos padrões de corte, $N = (N_1, \dots, N_p)^T$ cuja componente j é o limite superior para x_{jk} , calculado conforme a expressão (2.23) e o operador “ \cdot ” é o produto elemento a elemento de dois vetores. A fim de obter diferentes soluções eficientes para este problema, uma sequência de Problema de Programação Linear Inteiros necessitam ser resolvidos, cada um resultando numa solução eficiente. Se o PCUIM apresentar muitas destas soluções, este procedimento pode se tornar caro à medida que m e p aumentarem. A ideia utilizada é uma estratégia para acelerar a obtenção de um conjunto diverso de soluções potencialmente eficientes e descrito a seguir. Considere uma ligeira modificação da Formulação (2.24), onde a integralidade de x é relaxada:

$$\begin{aligned}
 &\text{Minimize} && z = (\|x\|_1, \|y\|_1) \\
 &\text{sujeito a} && A^*x \geq d, \\
 &&& x \leq N \cdot y, \\
 &&& x \geq y, \\
 &&& y \in \mathbb{B}^p, x \in \mathbb{R}_+^p.
 \end{aligned} \tag{5.12}$$

A ideia básica consiste em escalarizar a Formulação (5.12) em vez da Formulação (5.11). Potencialmente, as componentes em x desta formulação não serão inteiras. Nesse sentido, é necessário empregar algum procedimento heurístico, visto no Capítulo 4 para integralizar cada solução eficiente \tilde{x}^* encontrada. A vantagem desta abordagem é que o problema “parcialmente relaxado” (5.12) contém apenas p variáveis binárias, a metade do número de variáveis em relação a Formulação (2.22) e pode ser resolvido pelo CPLEX forma mais ágil. Esse ganho em termos do tempo computacional é ainda mais ressaltado no PCUIM, pois vários subproblemas são otimizados para a obtenção da fronteira de Pareto.

Por outro lado, a integralidade de y é mantida. Relaxar esta condição significa perder o controle do *setup* para o problema. Mantendo-a binária, temos o pleno domínio dos padrões que estão sendo efetivamente usados ou não na solução.

A próxima etapa consiste em definir uma maneira de integralizar \tilde{x} após a escalarização e geração de soluções eficientes inteiras. O simples arredondamento para o menor inteiro superior é fácil de ser implementado e barato computacionalmente, porém, pode produzir soluções eficientes de má qualidade. Em vez disso, propomos empregar a HADS, definida no capítulo anterior e adaptada neste problema. Esse método

é um procedimento simples de ser implementado e se baseia em alguma fundamentação matemática, podendo determinar um vetor \bar{x} inteiro, cuja norma-1 é muito próxima com a norma-1 do vetor ótimo x^* . Além disso, uso dessa heurística tornou viável porque m e p nos nossos exemplares foi da ordem de dezenas e conseqüentemente o número de direções simplex não foi tão elevado, sendo possível explorar todas. Somado a esse fato, quanto menor é *setup* de uma dada solução eficiente relaxada, menos direções temos que explorar para arredondar as componentes de x , tornando esse método rápido em nosso problema. Na seção dos resultados computacionais, esta ideia será comprovada nos exemplares de problemas que geramos.

Esta ideia é ilustrado com um breve exemplo a seguir.

Exemplo 5.4. *Considere um exemplar para o PCUIM com $m = 5$ itens, $b = 1$ bobina-mestre e um vetor demanda igual à $d = (161, 193, 135, 17, 172)^T$. Suponha que o procedimento de Gilmore-Gomore forneça os cinco padrões de corte alocados na matriz A^* em forma de coluna, como a seguir:*

$$A^* = \begin{bmatrix} 8 & 2 & 2 & 1 & 1 \\ 0 & 3 & 4 & 1 & 1 \\ 0 & 1 & 4 & 5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 4 \end{bmatrix}.$$

Utilizando a formulação (5.12), podemos construir o seguinte problema biobjetivo parcialmente relaxado:

$$\begin{aligned} & \text{Minimize } z = (\|x\|_1, \|y\|_1) \\ & \text{sujeito a } \begin{bmatrix} 8 & 2 & 2 & 1 & 1 \\ 0 & 3 & 4 & 1 & 1 \\ 0 & 1 & 4 & 5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \geq \begin{bmatrix} 161 \\ 193 \\ 135 \\ 17 \\ 172 \end{bmatrix}, \\ & x_1 \leq 172 \cdot y_1, \\ & x_2 \leq 172 \cdot y_2, \\ & x_3 \leq 81 \cdot y_3, \\ & x_4 \leq 193 \cdot y_4, \\ & x_5 \leq 193 \cdot y_5, \\ & x_j \geq y_j, \quad j = 1, \dots, 5, \\ & y_j \in \mathbb{B}, \quad x_j \in \mathbb{R}_+, \quad j = 1, \dots, 5. \end{aligned}$$

Nesta formulação, podemos aplicar as técnicas de esalarizações vistas no Capítulo 3 para obter as soluções eficientes para o PCUIM. Neste exemplo, em particular,

iremos aplicar o ε -Restrito. No entanto, após encontrar todas as soluções eficiente relaxadas deveremos arredondá-las pela HADS.

Aplicando o Algoritmo (7) para obter uma fronteira de Pareto heurística, \tilde{Z} tendo como base uma fronteira de Pareto \tilde{Z} relaxada resultante da formulação (5.12).

Primeiramente, calculamos os pontos lexicográficos, a saber $\tilde{z}^1 = (f_1(\tilde{x}^1), f_2(\tilde{x}^1))^T = (172; 1)^T$ e $\tilde{z}^5 = (f_1(\tilde{x}^5), f_2(\tilde{x}^5))^T = (85.84, 5)^T$ onde

$$\tilde{x}^5 = (4.61, 17.00, 25.92, 2.86, 35.45, 1, 1, 1, 1, 1)^T \text{ e } \tilde{x}^1 = (0, 172, 0, 0, 0, 0, 1, 0, 0, 0)^T.$$

Logo $\mathcal{I} = [1, 5]$.

- **Iteração 1:** resolva o Problema Restrito colocando a restrição $z_2 \leq 2$, isto é:

$$\begin{array}{l} \text{Minimize } z_1 = x_1 + x_2 + x_3 + x_4 + x_5 \\ \text{sujeito a } \begin{bmatrix} 8 & 2 & 2 & 1 & 1 \\ 0 & 3 & 4 & 1 & 1 \\ 0 & 1 & 4 & 5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \geq \begin{bmatrix} 161 \\ 193 \\ 135 \\ 17 \\ 172 \end{bmatrix}, \end{array}$$

$$x_1 \leq 172 \cdot y_1,$$

$$x_2 \leq 172 \cdot y_2,$$

$$x_3 \leq 81 \cdot y_3,$$

$$x_4 \leq 193 \cdot y_4,$$

$$x_5 \leq 193 \cdot y_5,$$

$$x_j \geq y_j, \quad j = 1, \dots, 5,$$

$$y_1 + y_2 + y_3 + y_4 + y_5 \leq 2,$$

$$y_j \in \mathbb{B}, \quad x_j \in \mathbb{R}_+, \quad j = 1, \dots, 5,$$

e a solução é o vetor $\tilde{x}^2 = (0, 62.20, 0, 36.60, 0, 0, 1, 0, 1, 0)^T$ cujo objetivo é $\tilde{z}^2 = (98.8, 2)^T$. Incrementamos ε em $\Delta = 1$ e iteramos novamente.

- **Iteração 2:** resolvemos novamente o Problema Restrito com a nova restrição $z_2 \leq 3$:

$$\begin{array}{l} \text{Minimize } z_1 = x_1 + x_2 + x_3 + x_4 + x_5 \\ \text{sujeito a } \begin{bmatrix} 8 & 2 & 2 & 1 & 1 \\ 0 & 3 & 4 & 1 & 1 \\ 0 & 1 & 4 & 5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \geq \begin{bmatrix} 161 \\ 193 \\ 135 \\ 17 \\ 172 \end{bmatrix}, \end{array}$$

$$x_1 \leq 172 \cdot y_1,$$

$$x_2 \leq 172 \cdot y_2,$$

$$x_3 \leq 81 \cdot y_3,$$

$$x_4 \leq 193 \cdot y_4,$$

$$x_5 \leq 193 \cdot y_5,$$

$$\begin{aligned} x_j &\geq y_j, \quad j = 1, \dots, 5, \\ y_1 + y_2 + y_3 + y_4 + y_5 &\leq 3, \\ y_j &\in \mathbb{B}, \quad x_j \in \mathbb{R}_+, \quad j = 1, \dots, 5, \end{aligned}$$

e a solução é $\tilde{x}^3 = (2.39, 51.17, 0, 39.47, 0, 1, 1, 0, 1, 0)^T$ o vetor $\tilde{z}^3 = (93.04, 3)^T$ no espaço de critério. Incrementamos ε por 1 e iteramos pela última vez.

- **Iteração 3:** resolvemos o problema com a nova restrição $z_2 \leq 4$:

$$\begin{aligned} \text{Minimize} \quad & z_1 = x_1 + x_2 + x_3 + x_4 + x_5 \\ \text{sujeito a} \quad & \begin{bmatrix} 8 & 2 & 2 & 1 & 1 \\ 0 & 3 & 4 & 1 & 1 \\ 0 & 1 & 4 & 5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \geq \begin{bmatrix} 161 \\ 193 \\ 135 \\ 17 \\ 172 \end{bmatrix}, \\ & x_1 \leq 172 \cdot y_1, \\ & x_2 \leq 172 \cdot y_2, \\ & x_3 \leq 81 \cdot y_3, \\ & x_4 \leq 193 \cdot y_4, \\ & x_5 \leq 193 \cdot y_5, \\ & x_j \geq y_j, \quad j = 1, \dots, 5, \\ & y_1 + y_2 + y_3 + y_4 + y_5 \leq 4, \\ & y_j \in \mathbb{B}, \quad x_j \in \mathbb{R}_+, \quad j = 1, \dots, 5, \end{aligned}$$

a solução é $\tilde{x}^4 = (3.77, 17.00, 29.50, 0, 37.80, 1, 1, 1, 0, 1)^T$ é o vetor $\tilde{z}^4 = (88.08, 4)^T$ e o algoritmo é finalizado.

Este método retornou cinco soluções eficientes \tilde{x}^j , $j = 1, \dots, 5$ com respeito a formulação relaxada, que potencialmente não são inteiras.

Após o término do Método ε -Restrito, é necessário arredondar tais soluções nas primeiras cinco componentes e depois de feito este arredondamento fazer os devidos ajustes nas componentes em y . Fazendo a aplicação desta HADS em cada solução, temos o seguinte:

- **solução \tilde{x}^1 :** já é inteira por sua vez, logo não precisa ser arredondada, e assim, $\bar{x}^1 = (0, 172, 0, 0, 0, 0, 1, 0, 0, 0)^T$, com $\bar{z}^1 = (172, 1)^T$.
- **solução \tilde{x}^2 :** necessita dos padrões 2 e 4 de A^* . Aplicando HADS, tentamos obter

uma solução inteira para o problema:

$$\begin{aligned} & \text{Minimize} && z_1 = x_2 + x_4 \\ & \text{sujeito a} && \begin{bmatrix} 2 & 1 \\ 3 & 1 \\ 1 & 5 \\ 1 & 0 \\ 1 & 3 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ x_4 \end{bmatrix} \geq \begin{bmatrix} 193 \\ 17 \end{bmatrix}, \\ & && x_j \in \mathbb{R}_+, \quad j = 1, 2, \end{aligned}$$

como componentes inteiras $\bar{x}_2^2 = 62$ e $\bar{x}_4^2 = 37$, logo a solução com todas as componentes serão dadas por $\bar{x}^2 = (0, 62, 0, 37, 0, 0, 1, 0, 1, 0)^T$, com $\bar{z}^2 = (99, 2)^T$. Note que apenas os padrões efetivamente utilizados na relaxação (o segundo e o quarto) são considerados na integralização da solução.

- **solução \tilde{x}^3** : necessita dos padrões 1, 2 e 4 de A^* . O arredondamento desta solução produz $\bar{x}^4 = (3, 51, 0, 40, 0, 1, 1, 0, 1, 0)^T$, com $\bar{z}^3 = (94, 3)^T$.
- **solução \tilde{x}^4** : esta solução precisa dos padrões 1, 2, 3 e 5. O arredondamento desta gera a solução $\bar{x}^4 = (4, 17, 30, 0, 38, 1, 1, 1, 0, 1)^T$, com $\bar{z}^4 = (89, 4)^T$.
- **solução \tilde{x}^5** : utiliza todos os cinco padrões de corte contidos na matriz A^* . O arredondamento produz uma solução inteira $\bar{x}^5 = (5, 17, 26, 3, 36, 1, 1, 1, 1, 1)^T$, com $\bar{z}^1 = (87, 5)^T$.

Assim, ao final do procedimento geral e conjunto, temos uma fronteira de Pareto $\bar{\mathcal{Z}}$ aproximada para o PCUIM.

Uma fronteira de Pareto exata \mathcal{Z}^* poderia ser obtida utilizando-se a formulação (5.11) que considera $x_j \in \mathbb{Z}_+$. Aplicando o Método ε -Restrito em tal formulação, teríamos as seguintes soluções eficientes:

- **solução x^{*1}** = $(0, 172, 0, 0, 0, 0, 1, 0, 0, 0)^T$ com $z^{*1} = (172, 1)^T$;
- **solução x^{*2}** = $(0, 62, 0, 37, 0, 0, 1, 0, 1, 0)^T$ com $z^{*2} = (99, 2)^T$;
- **solução x^{*3}** = $(3, 51, 0, 40, 0, 1, 1, 0, 1, 0)^T$ com $z^{*3} = (94, 3)^T$;
- **solução x^{*4}** = $(4, 17, 30, 0, 38, 1, 1, 1, 0, 1)^T$ com $z^{*4} = (89, 4)^T$;
- **solução x^{*5}** = $(5, 17, 26, 3, 36, 1, 1, 1, 1, 1)^T$ com $z^{*5} = (87, 5)^T$.

Observamos que as fronteiras $\bar{\mathcal{Z}} = \tilde{\mathcal{Z}}$. O capítulo dos resultados computacionais atesta a ideia de que essas fronteiras são muito próximas uma da outra.

A Tabela (6) a seguir resume as três fronteiras obtidas.

Tabela 6 – Três soluções eficientes obtidas para o Exemplo 5.8.1 e seus respectivos valores objetivos

\mathcal{X}					\mathcal{X}					\mathcal{X}^*				
\tilde{x}^1	\tilde{x}^2	\tilde{x}^3	\tilde{x}^4	\tilde{x}^5	\bar{x}^1	\bar{x}^2	\bar{x}^3	\bar{x}^4	\bar{x}^5	x^{*1}	x^{*2}	x^{*3}	x^{*4}	x^{*5}
0	0	2.39	3.77	4.61	0	0	3	4	5	0	0	3	4	5
172.00	62.20	51.17	17.00	17.00	172	62	51	17	17	172	62	51	17	26
0	0	0	29.50	25.92	0	0	0	30	26	0	0	0	30	3
0	36.60	39.47	0	2.86	0	37	40	0	3	0	37	40	0	3
0	0	0	37.80	35.45	0	0	0	38	36	0	0	0	38	36
0	0	1	1	1	0	0	1	1	1	0	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	0	0	1	1	0	0	0	1	1
0	1	1	0	1	0	1	1	0	1	0	1	1	0	1
0	0	0	1	1	0	0	0	1	1	0	0	0	1	1
\tilde{z}^1	\tilde{z}^2	\tilde{z}^3	\tilde{z}^4	\tilde{z}^5	\bar{z}^1	\bar{z}^2	\bar{z}^3	\bar{z}^4	\bar{z}^5	z^{*1}	z^{*2}	z^{*3}	z^{*4}	z^{*5}
172.00	98.8	93.04	88.08	85.84	172	99	94	89	87	172	99	94	89	87
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5

Este capítulo forneceu como cada método de esclarização multiobjetivo foi adaptado para resolver o PCUIM. Apresentou vantagens e desvantagens de cada procedimento e suas limitações. Além disso, propusemos um novo método para resolver problemas combinatórios biobjetivos, baseado numa combinação do Método NISE e o Método da Métrica de Tchebycheff. Apresentamos uma heurística de relaxação para acelerar a obtenção de uma aproximação para o conjunto de soluções eficientes. O próximo capítulo visa validar e comparar as ideias desenvolvidas, ao realizar uma série de experimentos computacionais.

Capítulo 6

Testes Computacionais

Neste capítulo, apresentamos os experimentos computacionais realizados com o PCUIM. As simulações foram divididas em quatro fases distintas, subdivididas nas subseções que seguem. Para avaliar e comparar os métodos multiobjetivos e heurísticas adaptadas e propostas, todos os problemas nas Fases 2 e 3 foram resolvidos pelos sete métodos apresentados no capítulo anterior, a saber¹:

- Método da Soma Ponderada (SP);
- Método da Métrica de Tchebycheff (Tch);
- Método da Métrica de Tchebycheff Sem Ponderação (Tch-SP);
- Método do ε -Restrito (ε -R);
- Método NISE híbrido com ε -Restrito (NISE+ ε -R);
- Método de Benson (Ben) e
- Método de Silva & Crema (S-C).

Nas Fases 1 e 4, aplicamos apenas as técnicas ε -Restrito, Tchebycheff Sem Ponderações e Benson.

A primeira fase contempla os resultados computacionais relacionados com instâncias com poucos itens (≤ 10) e onde os padrões de corte não são fornecidos a priori. A segunda e a terceira fases têm em comum o fato de utilizarem os padrões de corte a priori. Finalmente a quarta fase das simulações englobam experimentos que consideram custos c_{jk}^1 e c_{jk}^2 não unitários para x_{jk} e y_{jk} , respectivamente.

Os testes computacionais e os algoritmos desenvolvidos para este problema foram codificados usando o *software* Matlab, versão 7.10.0 R2010a (([MATLAB, 2010](#))).

¹ Nas tabelas, iremos utilizar o nome dos métodos de maneira abreviada, como entre parêntesis.

Os subproblemas foram resolvidos através da API do CPLEX 12.5 ((IBM... , 2010)) com este *software*. Todos os algoritmos foram simulados em um computador Core i7 com 8GB de memória, do Instituto de Matemática, Estatística e Computação Científica - IMECC/UNICAMP, Campinas, SP, Brasil.

Utilizamos os parâmetros padrões do CPLEX para resolver os subproblemas de Problema de Programação Linear Inteiros mistos. No entanto, na Fase 1, limitamos à 60.000 nós da árvore do *Branch-and-Bound* em cada Problema de Programação Linear Inteiro ou um *gap* de 15% entre os limitantes superior e inferior, isto é, o CPLEX foi interrompido quando uma dessas condições for satisfeita. Se estes parâmetros não fossem adotados, os subproblemas da Fase 1 demandariam um tempo considerável para serem otimizados completamente. Alguns testes isolados com $m = 8$ itens demoraram cerca de 1.5 hora para serem completamente otimizados. Noutros exemplares, o CPLEX com 1 hora de execução tinha atingido 100.000 nós e uma solução com *gap* de 11%. Nas Fases 2, 3 e 4, o número de nós foi fixado em 5.000 e um *gap* de integralidade de 1%.

A principal maneira de avaliar a qualidade dos métodos apresentados é comparando as fronteiras de Pareto obtidas e o tempo computacional. A comparação das fronteiras é feita mediante a aplicação de várias métricas comparativas apresentadas no Capítulo 3, Seção 7. Porém, antes de rever toda a notação empregada para estas métricas, adotamos a seguinte terminologia para as fronteiras de Pareto que nós obtemos:

- \mathcal{Z}^* : representa a fronteira de Pareto ótima para o PCUIM, mas com respeito aos $p < \mu$ padrões de corte que estão sendo fornecidos a priori, considerando a Formulação (5.11). A aplicação de técnicas exatas de otimização (como o *Branch-and-Bound*) para resolver cada subproblema, produz soluções eficientes para o PCUIM. A palavra *ótima* é um abuso da linguagem neste texto e não está relacionada à questão de otimalidade global do problema de corte, que só seria conseguida se todos os μ padrões de corte possíveis estivessem fornecidos e fossem resolvidos os subproblemas inteiros resultantes. Dessa forma, esta aplicação é impossível de ser feita neste contexto e uma simplificação utilizando apenas os padrões “bons” é considerada. Naturalmente, \mathcal{Z}^* está associada aos padrões que são dados para a formulação.

Esta fronteira é obtida nas Fases 1, 2 e 4.

- $\tilde{\mathcal{Z}}$: é a fronteira de Pareto relacionada à formulação do Problema (5.12), onde a integralidade de x_{jk} é relaxada. A obtenção desta fronteira é realizada pela aplicação de técnicas exatas de otimização nos subproblemas, como o *Branch-and-Bound*, implementado pelo CPLEX.

Esta fronteira é obtida nas Fases 2, 3 e 4, mas apenas na Fase 3 é utilizada para efeitos de comparação.

- \bar{Z} : denota uma fronteira de Pareto heurística, obtida ao fazer o arredondamento de cada solução eficiente parcialmente relaxada de \tilde{Z} . É recomendável que ela se aproxime de Z^* .

É obtida nas Fases 2, 3 e 4 para fins comparativos.

Pela definição destas fronteiras, temos a seguinte relação: $\tilde{Z} \leq Z^* \leq \bar{Z}$.

A maneira de comparar fronteiras de Pareto é feito pelas métricas apresentadas no Capítulo 3. A seguinte terminologia será utilizada neste capítulo para representar estas medidas:

- σ^1 : denota a Métrica 1 e fornece a distância Euclidiana média dos pontos de
 - \bar{Z} em relação aos pontos de Z^* , nas Fases 2 e 4;
 - \tilde{Z} em relação aos pontos de \bar{Z} , na Fase 3.
- σ^2 : denota a Métrica 2 e fornece a porção dos pontos de \bar{Z} que não coincidem com os pontos de Z^* (pois $\delta = 0$). É calculada nas Fases 2 e 4.
- σ_{11}^3 e σ_{12}^3 : denotam a Métrica 3, mensurando as amplitudes entre os pontos lexicográficos da fronteira Z^* com respeito primeira e segunda função-objetivo, respectivamente. Da mesma forma σ_{21}^3 e σ_{22}^3 denotam as amplitudes entre os lexicográficos de \bar{Z} com respeito primeira e segunda função-objetivo, respectivamente. Essa métrica foi calculada nas Fases 2 e 3 apenas.
- σ_1^4 e σ_2^4 : denotam a Métrica 4 e significam, respectivamente, as áreas delimitadas pelas das fronteiras Z^* e \bar{Z} . Foi calculada em todas as fases.
- σ_1^5 e σ_2^5 : denotam a Métrica 5 e significam, respectivamente, a cardinalidade dos conjuntos Z^* e \bar{Z} . Após a resolução de todos os subproblemas, esses conjuntos são obtidos tomando apenas os pontos não dominados, utilizando para isso o Algoritmo (11) no Anexo 1.1. Esta métrica também foi calculada em todas as fases.

Além destas cinco métricas, uma sexta medida foi empregada neste trabalho nas Fases 1 e 4. A Métrica σ^6 denota o número de subproblemas que cada técnica de escalarização empregou para obter a fronteira resultante.

Os resultados para cada fase estão organizados nas subseções seguintes.

6.1 Resultados computacionais sem os padrões de corte a priori - Fase 1

Dentre os modelos linearizados (2.20) ou (2.21) vistos no Capítulo 2, aquele que produzir um menor número de variáveis inteiras foi utilizado para a aplicação dos três métodos de escalarização utilizados nessa fase: ε -Restrito, Tchebycheff Sem Ponderações e Benson.

As três Métricas σ^4 , σ^5 e σ^6 foram aplicadas para servir de base para as devidas comparações.

É importante frisar que a cardinalidade do conjunto $P = \{1, \dots, p\}$ dos padrões foi fixada em $2 \cdot m$ nos experimentos. Isso é uma opção heurística, baseada na experiência de que o “*Em um PCUIM com m itens, o setup terá um valor máximo próximo de m* ”. Sucessivos testes comprovam que esse número nunca foi maior que $2 \cdot m$. A cardinalidade deste conjunto deve ser definida de tal maneira a não perder nenhuma solução eficiente para o problema, mas ao mesmo tempo, envolver o menor número de variáveis possível para a formulação.

A fim de testar e comparar os métodos desenvolvidos, geramos aleatoriamente instâncias para o PCUIM através de uma adaptação do gerador CUTGEN desenvolvido em (GAU; WASCHER, 1995), amplamente utilizado na literatura. Esta adaptação foi baseada no trabalho em (POLDI; ARENALES, 2009). Os exemplares foram divididos em 10 classes e cada uma com 20 problemas teste.

As classes têm as seguintes características:

- O número de itens m demandados foi igual à 5, 8 e 10.
- As larguras dos itens ℓ_i foram geradas entre $[v_1 \cdot \bar{L}, v_2 \cdot \bar{L}]$, onde L_k foi escolhido no intervalo $[300, 1000]$ e \bar{L} é a média dos valores L_k . Para itens pequenos (P) utilizamos $v_1 = 0.01$ e $v_2 = 0.2$ e $v_1 = 0.01$ e $v_2 = 0.8$ e para itens grandes (G), $v_1 = 0.2$ e $v_2 = 0.8$.
- O número de facas foi fixo igual a $q = \left\lceil \sum_{i=1}^m \frac{L}{m \cdot \ell_i} \right\rceil$.
- A demanda d_i para cada item foi gerada aleatoriamente no intervalo $[10, 200]$.
- O número de bobinas-mestre b foi 1 e 3.

O PCUIM com itens grandes e $m = 5$ frequentemente possuiu apenas uma solução eficiente. Por isso estas classes não foram geradas. A numeração das classes está disposta na Tabela (7) a seguir.

Tabela 7 – Classes para o PCUIM - Fase 1

Classe	m	Tipo do item	b
1	5	P	1
2	5	P	3
3	8	P	1
4	8	P	3
5	8	G	1
6	8	G	3
7	10	P	1
8	10	P	3
9	10	G	1
10	10	G	3

Em cada tabela apresentada, calculamos as médias e coeficiente de variação (cv) dentre os 20 problemas dentro de cada classe. Os resultados para σ^4 encontram-se na Tabela (8).

Devido à limitação de 60000 do número de nós em cada Problema de Programação Linear Inteiro, as fronteiras eficientes sofreram pequenas variações conforme uma ou outra técnica de escalarização foi adotada. Essa diferença foi também influenciada porque após a otimização dos subproblemas, apenas as soluções eficientes são consideradas e as demais descartadas. Ficou claro que para esta quantidade máxima de nós, o Método Tchebycheff Sem Ponderações foi capaz de encontrar uma fronteira de Pareto “mais inferior” em relação aos métodos concorrentes. Noutras palavras, a área média delimitada pela fronteira Tchebycheff Sem Ponderações foi 8.8% e 6.0% maior em relação ao ε -Restrito e Benson, respectivamente.

Os resultados numéricos para a Métrica σ^5 encontram-se disponibilizados na Tabela (9). Notavelmente, esta e as outras métricas são influenciadas pelo critério de parada dos Problema de Programação Linear Inteiros (60000 nós ou 15% de *gap*).

Como se pôde notar, em média, o Método Tchebycheff Sem Ponderações forneceu 3.24 soluções eficientes, contra 2.74 do ε -Restrito e 2.86 do Benson (com o mesmo cv , aproximadamente), comprovando que o Método Tchebycheff Sem Ponderações determina mais soluções que os demais sob o mesmo critério de parada. Uma outra análise que pode ser feita consiste no baixo número de soluções obtidas, em média, 2.95. Como a formulação empregada nesta fase também fornece os padrões de corte implicitamente, computados à medida que o valor do *setup* se modifica, gerando um conjunto de padrões de corte diferente e fazendo com que o problema tenha um número de soluções eficientes relativamente pequeno. Além disso, como era de se esperar, os PCUIMs com itens grandes possuem menos soluções eficientes do que os PCUIMs com itens pequenos em relação a largura da bobina-mestre.

Tabela 8 – Valores médios para a Métrica σ^4 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1

Classe	ε -R		Tch-SP		Ben		Média
	σ^4	cv	σ^4	cv	σ^4	cv	
1	62.32	0.66	61.96	0.67	62.18	0.66	62.15
2	46.13	1.11	41.34	0.95	45.34	1.10	44.27
Média	54.22	0.88	51.65	0.81	53.76	0.88	53.21
3	296.85	0.68	317.55	0.68	289.75	0.65	301.38
4	147.20	1.91	159.10	1.78	149.70	1.91	152.00
5	179.55	1.81	215.50	1.72	180.54	2.03	191.86
6	112.95	0.74	166.10	0.95	161.54	1.06	146.86
Média	184.14	1.28	214.56	1.28	195.38	1.41	198.03
7	177.51	1.28	194.76	1.19	170.73	1.33	181.00
8	91.70	1.18	106.80	1.14	82.55	1.21	93.68
9	126.05	1.10	124.84	1.24	89.90	1.35	113.60
10	406.60	1.44	407.08	1.40	401.12	1.48	404.93
Média	200.46	1.25	208.37	1.24	186.07	1.35	198.30
Média Geral	160.44	1.18	174.59	1.16	159.12	1.26	164.72

Tabela 9 – Valores médios para a Métrica σ^5 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1

Classe	ε -R		Tch-SP		Ben		Média
	σ^5	cv	σ^5	cv	σ^5	cv	
1	2.87	0.17	3.20	0.17	2.94	0.18	3.00
2	2.67	0.25	3.24	0.21	2.83	0.23	2.92
Média	2.77	0.21	3.22	0.19	2.89	0.21	2.96
3	3.60	0.26	4.15	0.23	3.75	0.26	3.83
4	2.20	0.45	2.85	0.29	2.50	0.27	2.52
5	2.35	0.37	3.00	0.40	2.40	0.24	2.58
6	2.40	0.40	3.00	0.36	2.60	0.31	2.67
Média	2.64	0.37	3.25	0.32	2.81	0.27	2.90
7	3.38	0.28	3.90	0.26	3.60	0.39	3.63
8	2.55	0.48	3.35	0.38	2.90	0.36	2.93
9	2.65	0.27	2.95	0.35	2.55	0.25	2.72
10	2.68	0.32	2.80	0.37	2.55	0.29	2.68
Média	2.81	0.34	3.25	0.34	2.90	0.32	2.99
Média Geral	2.74	0.32	3.24	0.30	2.86	0.28	2.95

A Métrica σ^6 também foi analisada. Da forma como os algoritmos ε -Restrito e Benson foram desenvolvidos no PCUIM, é óbvio que ambos determinam o mesmo número de subproblemas, isto é, $\sigma^6 = z_2^+ - z_2^- - 2$. Em relação ao Tchebycheff Sem Ponderações, um número diferente de subproblemas é esperado, haja vista que alguns deles foram infactíveis, especialmente quando entre os pontos de referência não houver uma solução eficiente. Mesmo sob essa consideração, σ^6 ficou muito próxima entre as três técnicas em média como pode ser analisado na Tabela (10).

Tabela 10 – Valores médios para a Métrica σ^6 nos métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1

Classe	ε -R		Tch-SP		Ben		Média
	σ^6	cv	σ^6	cv	σ^6	cv	
1	4.13	0.21	5.40	0.21	4.13	0.21	4.55
2	4.82	0.24	5.41	0.25	4.82	0.24	5.02
Média	4.48	0.23	5.40	0.23	4.48	0.23	4.79
3	7.55	0.40	7.30	0.27	7.55	0.40	7.47
4	4.40	0.50	4.35	0.47	4.40	0.50	4.38
5	5.65	0.59	4.80	0.52	5.65	0.59	5.37
6	5.25	0.57	4.85	0.48	5.25	0.57	5.12
Média	5.71	0.52	5.33	0.43	5.71	0.52	5.58
7	6.59	0.31	6.80	0.30	6.59	0.31	6.66
8	3.55	0.33	5.30	0.54	3.55	0.33	4.13
9	5.25	0.34	4.90	0.42	5.25	0.34	5.13
10	6.70	0.38	4.50	0.44	6.70	0.38	5.97
Média	5.52	0.34	5.38	0.43	5.52	0.34	5.47
Média Geral	5.35	0.38	5.36	0.38	5.35	0.38	5.35

Tabela 11 – Tempos computacionais médios (em seg.) para resolver cada subproblema para os métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1

Classe	ε -R		Tch-SP		Ben		Média
	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	
1	35.56	0.21	17.85	0.22	36.79	0.19	30.07
2	44.64	0.24	24.63	0.31	45.00	0.23	38.09
Média	40.10	0.23	19.24	0.27	40.90	0.21	34.08
3	60.26	0.40	39.56	0.58	60.61	0.41	53.48
4	71.72	0.50	50.55	0.52	70.74	0.50	64.34
5	65.67	0.59	45.96	0.53	66.79	0.59	59.47
6	66.40	0.57	51.61	0.58	66.54	0.57	61.52
Média	66.01	0.52	46.92	0.55	66.17	0.51	59.70
7	94.01	0.31	62.30	0.41	94.56	0.31	83.62
8	88.69	0.33	63.74	0.39	84.85	0.33	79.09
9	95.14	0.34	71.24	0.42	95.73	0.33	87.37
10	96.19	0.38	91.30	0.47	96.00	0.38	94.50
Média	93.50	0.34	72.14	0.42	92.79	0.34	86.14
Média Geral	70.61	0.38	50.70	0.44	70.57	0.38	63.96

A Tabela (11) ilustra os tempos computacionais médios (t) transcorridos para a resolução de cada subproblema gerado por estes métodos. Observamos praticamente o mesmo tempo utilizado pelo ε -Restrito e Benson. No entanto, é notável que o Método Tchebycheff Sem Ponderações foi notavelmente mais rápido que os demais. Em termos quantitativos, ele consumiu em média 71% do tempo dos concorrentes para cada subproblema.

Essa diferença no tempo de resolução desses problemas se espelha na Tabela

(12), onde colocamos os tempo computacionais totais médios (em seg.) para obtenção das respectivas fronteiras de Pareto. O Método proposto neste trabalho consumiu, em média, apenas 36% do tempo dos demais. Como os três produziram praticamente o mesmo número de subproblemas, os resultados atestam que a formulação dos subproblemas do Tchebycheff Sem Ponderações, embora tenham as quatro restrições limitando as funções-objetivo, fizeram o CPLEX ter mais facilidade em atingir os 60000 nós. Isso atesta e comprova a ideia de que esses subproblemas fornecem melhores limitantes para a função-objetivo, fazendo a ramificação e as podas no *Branch-and-Bound* ficarem mais rápidas. Sem dúvida, essa foi a característica mais importante no método que propomos.

Outra análise importante está na diferença entre as classes com itens pequenos e grandes. Considerando, por exemplo, as classes 3 e 5 ($m = 8$) ou 7 e 9 ($m = 10$) e comparando-as entre si, vemos que as classes com itens pequenos demoram mais do dobro do tempo para serem resolvidas.

Tabela 12 – Tempos computacionais médios (em seg.) para os métodos ε -Restrito, Tchebycheff Sem Ponderações e Benson - fase 1

Classe	ε -R		Tch-SP		Ben		Média
	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	
1	66.00	0.98	12.35	1.50	71.60	1.00	49.98
2	113.28	1.10	25.71	0.92	115.66	1.10	84.88
Média	89.64	1.04	21.03	1.21	93.63	1.05	67.43
3	96.29	0.68	38.85	0.67	93.28	0.68	76.14
4	278.21	1.27	100.65	1.30	281.05	1.32	219.95
5	246.59	1.52	63.78	1.21	258.86	1.51	189.74
6	165.50	1.37	97.20	1.10	166.89	1.37	143.20
Média	196.65	1.21	75.12	1.07	200.02	1.22	157.26
7	100.47	0.53	79.80	0.76	84.28	0.53	88.18
8	343.98	0.99	125.53	1.18	347.94	1.03	272.47
9	224.06	0.70	74.06	0.73	226.70	0.68	174.94
10	327.94	1.05	111.08	1.13	326.47	1.06	255.14
Média	249.11	0.82	97.62	0.95	246.35	0.82	197.69
Média Geral	192.13	1.02	70.83	1.05	193.29	1.03	152.08

A Figura (39) ilustra como tempo computacional é influenciado pela instância do problema bem como a técnica de escalarização que foi utilizada.

Em suma, o Método Tchebycheff Sem Ponderações consumiu menos tempo e obteve um número ligeiramente maior de soluções eficientes.

A Tabela (13) apenas ilustra a quantidade média de variáveis e restrições geradas pelas linearizações, que depende do número de bobinas-mestre e do tamanho dos itens. Embora garanta a otimalidade matemática da fronteira de Pareto, o elevado aumento da dimensão do problema e as dificuldades computacionais para sua completa otimização atestam que a formulação não linear e posteriormente linearizada pode ter

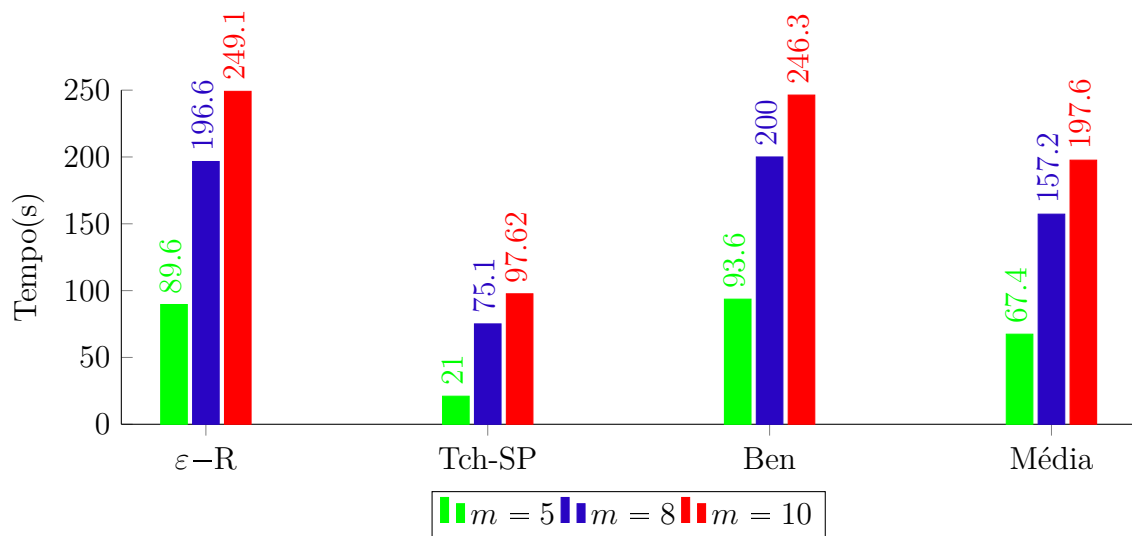


Figura 39 – Comparação dos tempos computacionais (em seg.) utilizados para obter a fronteira de Pareto para o PCUIM - Fase 1

aplicação bastante restrita na prática.

Tabela 13 – Número médio de variáveis e restrições do PCUIM - Fase 1

Classe	#Restrições	<i>cv</i>	#Variáveis	<i>cv</i>
1	1380.16	0.04	590.06	0.04
2	4129.77	0.04	1769.91	0.04
Média	2754.96	0.04	1179.98	0.04
3	3524.80	0.03	1432.80	0.02
4	9012.00	0.20	4496.00	0.10
5	1150.00	0.57	526.00	0.48
6	6879.18	0.67	3038.41	0.50
Média	5141.20	0.37	2373.30	0.28
7	5510.00	0.00	2200.00	0.00
8	16510.00	0.00	6600.00	0.00
9	1130.00	0.00	720.00	0.00
10	2830.30	0.43	1800.22	0.36
Média	6495.30	0.31	2830.05	0.24
Média Geral	4797.15	0.27	2127.43	0.21

A Tabela (14) ilustra os *gaps* médios de cada subproblema resolvido no momento em que a execução do CPLEX foi interrompida. Também mostramos o número médio de nós utilizados por cada subproblema.

Na resolução dos subproblemas escalarizados, ficou nítido a maior rapidez em resolver os subproblemas cujo objetivo é minimizar o *setup*. Em todas as classes, houve uma grande dificuldade em determinar o ponto lexicográfico com valor mínimo em z_1 . A Tabela (14) apenas confirma os resultados ilustrados ao longo desta seção, onde podemos concluir que os subproblemas de Tchebycheff Sem Ponderações são mais facilmente resolvidos

Tabela 14 – Valores médios para o *gap* e número de nós para os subproblemas no ε -Restrito, Tchebycheff Sem Ponderações e Benson - Fase 1

Classe	ε -R		Tch-SP		Ben	
	<i>gap</i>	#nós	<i>gap</i>	#nós	<i>gap</i>	#nós
1	0.00	25280	0.00	28822	0.00	15488
2	0.00	3178	0.00	2332	0.00	7941
Média	0.00	14229	0.00	15577	0.00	11714
3	3.53	48981	2.91	44962	5.88	56876
4	7.89	38983	6.78	42821	11.89	49892
5	9.89	25954	10.23	23542	13.97	30532
6	12.78	28456	11.89	29835	14.32	35612
Média	8.52	35593	7.95	35290	11.51	43228
7	13.83	58982	13.78	53776	14.82	59732
8	14.87	59772	12.62	58921	14.22	58998
9	12.22	47971	12.25	45832	13.72	42551
10	12.79	53620	12.87	48443	13.20	55821
Média	13.42	55063	12.88	51743	13.99	54275
Média Geral	7.31	34961	6.94	34203	12.75	36425

pelo CPLEX, pois o número de nós pesquisados na árvore de busca e o *gap* da solução determinada, em média, foi mais reduzido.

Exemplo 6.1. *A seguir ilustramos um exemplo de um conjunto eficiente e de uma fronteira de Pareto para um exemplar da classe 8. A largura dos itens é dada pelo vetor $\ell = (48, 59, 118, 24, 104, 80, 49, 28, 55, 61)^T$, a largura da bobina-mestre dada pelo vetor $L = (768, 433, 532)^T$ e o vetor demanda dado por $d = (33, 123, 53, 84, 121, 58, 66, 128, 61, 167)^T$. A fronteira é exibida na Figura (40), onde quatro soluções eficientes foram obtidas, cujos vetores objetivos são os pontos A, B, C e D.*

As quatro soluções eficientes são completamente distintas entre si, como as frequências e os padrões de corte utilizados. Essas soluções são ilustradas na Tabela (15) com detalhes.

Para cada solução anterior, calculamos a soma das perdas (perda em um padrão \times frequência) e o excesso de produção relativos (em %). Notamos que a medida que o setup diminui, há um considerável aumento no excesso de produção. No entanto, as perdas aumentam e diminuem com o decréscimo do setup porque os padrões variam a cada solução eficiente.

Exemplo 6.2. *Considere o mesmo Exemplo 4.1.3. Ilustramos que há $\mu = 378$ combinações de corte possíveis. Gerando-as e aplicando o Método ε -Restrito (ou qualquer outro método de escalarização) na formulação biobjetivo (5.11) com os estes μ padrões de corte, temos duas soluções eficientes:*

- **solução 1:** usa dois padrões de corte, a saber $a_1^* = (3, 3, 4, 3)^T$ e $a_2^* = (1, 2, 0, 7)^T$,

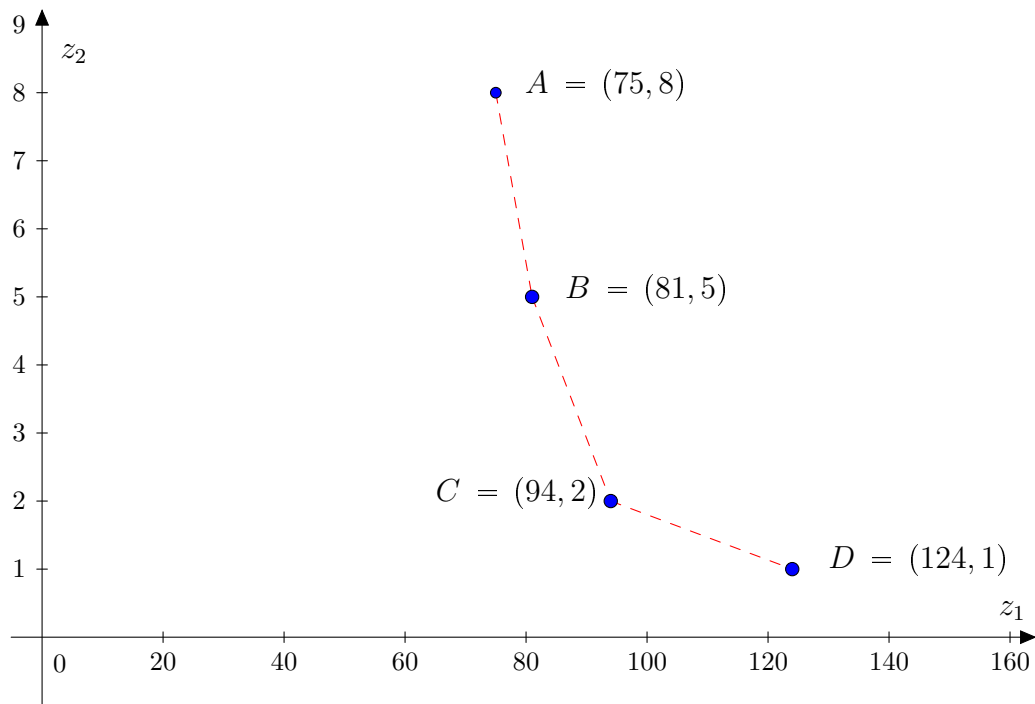


Figura 40 – Exemplo de uma fronteira de Pareto para o PCUIM para um problema da classe 8

com frequências de $x_1^* = 43$ e $x_2^* = 2$.

- **solução 2:** usa $x_1^* = 50$ vezes o padrão de corte $a_1^* = (2, 3, 3, 4)^T$.

O procedimento de linearização também foi aplicado neste exemplar, gerando duas soluções eficientes seguintes:

- **solução 1:** usa dois padrões de corte, a saber $a_1^* = (2, 3, 3, 4)^T$ e $a_2^* = (9, 1, 6, 1)^T$, com frequências de $x_1^* = 39$ e $x_2^* = 4$.
- **solução 2:** usa $x_1^* = 50$ vezes o padrão de corte $a_1^* = (2, 3, 3, 4)^T$ (idêntica à solução 2 do procedimento anterior).

Embora não sejam exatamente as mesmas soluções eficientes, os vetores objetivos nos dois métodos são iguais. No entanto, a primeira abordagem consumiu 96 segundos ao passo que a segunda demandou apenas 14.3 segundos.

6.2 Resultados computacionais com os padrões de corte a priori - Fase 2

Nesta segunda fase de simulações computacionais, analisamos duas maneiras distintas de resolver o PCUIM. Em ambas, utilizamos os padrões de corte a priori.

Tabela 15 – Quatro diferentes soluções para o PCUIM com os padrões não fornecidos a priori

	padrões de corte	Freq.	Perda	Excesso Produção
Solução A	$(0 \ 0 \ 0 \ 0 \ 2 \ 5 \ 2 \ 0 \ 0 \ 1)^T$	2	2×1	
	$(0 \ 0 \ 0 \ 7 \ 1 \ 1 \ 0 \ 14 \ 0 \ 0)^T$	5	5×24	
	$(0 \ 9 \ 2 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)^T$	1	1×1	
	$(0 \ 0 \ 0 \ 0 \ 4 \ 1 \ 3 \ 0 \ 2 \ 0)^T$	16	16×15	
	$(2 \ 1 \ 0 \ 0 \ 2 \ 1 \ 0 \ 1 \ 2 \ 3)^T$	17	17×4	
	$(0 \ 0 \ 3 \ 3 \ 1 \ 1 \ 0 \ 1 \ 0 \ 2)^T$	17	7×8	
	$(0 \ 6 \ 0 \ 0 \ 0 \ 0 \ 1 \ 2 \ 0 \ 5)^T$	16	16×4	
	$(0 \ 7 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)^T$	1	1×20	
	$\Sigma = 75$	$\Sigma = 0.09\%$	$\Sigma = 1.90\%$	
Solução B	$(5 \ 0 \ 3 \ 2 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1)^T$	5	5×16	
	$(2 \ 0 \ 1 \ 9 \ 0 \ 0 \ 6 \ 1 \ 0 \ 0)^T$	4	4×16	
	$(0 \ 0 \ 5 \ 6 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0)^T$	4	4×6	
	$(0 \ 2 \ 0 \ 0 \ 2 \ 1 \ 1 \ 2 \ 1 \ 3)^T$	64	64×19	
	$(0 \ 0 \ 5 \ 4 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0)^T$	4	4×5	
	$\Sigma = 81$	$\Sigma = 2.25\%$	$\Sigma = 8.22\%$	
Solução C	$(1 \ 2 \ 1 \ 1 \ 1 \ 1 \ 2 \ 2 \ 0 \ 2)^T$	80	80×0	
	$(0 \ 0 \ 0 \ 1 \ 3 \ 1 \ 0 \ 0 \ 5 \ 1)^T$	14	14×16	
		$\Sigma = 94$	$\Sigma = 0.03\%$	
Solução D	$(2 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 2 \ 1 \ 2)^T$	121	121×5	
		$\Sigma = 121$	$\Sigma = 0.65\%$	

A primeira delas resolve os subproblemas escalarizados considerando a frequência x_{jk} dos padrões de corte como inteiro. A segunda abordagem relaxa esta integralidade e posteriormente, realizamos um arredondamento destas soluções usando a HADS.

Geramos 27 classes onde contemplam as combinações possíveis entre itens pequenos, médios e grandes, quantidade de itens e número de bobinas-mestres, da seguinte forma:

- O número de itens m demandados variou entre 10, 20 e 40.
- As larguras dos itens ℓ_i foram geradas no intervalo $[v_1 \cdot \bar{L}, v_2 \cdot \bar{L}]$, onde L_k foi escolhido entre $[300, 1000]$ e \bar{L} é a média dos valores L_k . Para itens pequenos (P) utilizamos $v_1 = 0.01$ e $v_2 = 0.2$, para itens de diversos tamanhos (M) $v_1 = 0.01$ e $v_2 = 0.8$ e para itens grandes (G) utilizamos $v_1 = 0.2$ e $v_2 = 0.8$.

- O número de facas for fixo igual a $q = \left\lceil \sum_{i=1}^m \frac{L}{m \cdot \ell_i} \right\rceil$
- A demanda d_i para cada item foi gerada aleatoriamente no intervalo $[10, 200]$.
- O número de bobinas-mestres b foi igual a 1, 3 e 5.

A numeração de cada classe é mostrada nas Tabela (16).

Tabela 16 – Classes para o PCUIM - Fase 2

Classe	m	Tipo do item	b
1	10	P	1
2	10	P	3
3	10	P	5
4	10	M	1
5	10	M	3
6	10	M	5
7	10	G	1
8	10	G	3
9	10	G	5
10	20	P	1
11	20	P	3
12	20	P	5
13	20	M	1
14	20	M	3
15	20	M	5
16	20	G	1
17	20	G	3
18	20	G	5
19	40	P	1
20	40	P	3
21	40	P	5
22	40	M	1
23	40	M	3
24	40	M	5
25	40	G	1
26	40	G	3
27	40	G	5

Para cada problema gerado as duas abordagens foram utilizadas pelos sete algoritmos de escalarizações apresentados no Capítulo 5. A primeira delas adota o modelo (5.11) que usa como argumento de entrada os padrões de corte obtidos pelo método de GC e com todas as variáveis inteiras, resultando numa fronteira \mathcal{Z}^* para cada um dos procedimentos de escalarização propostos.

Em seguida, uma segunda abordagem considera o Problema (5.12) parcialmente relaxado, gerando uma fronteira $\tilde{\mathcal{Z}}$ para cada técnica de escalarização adotada. Esta fronteira, por sua vez, é integralizada usando-se a HADS, cujo resultado é uma fronteira

heurística \bar{Z} . Sendo assim, para cada técnica de escalarização, as duas fronteiras Z^* e \bar{Z} são obtidas e comparadas entre si.

Essa comparação é realizada detalhadamente pelas cinco Métricas. Dentro de cada classe do PCUIM, calculamos os valores médios de cada Métrica, σ^i , $i = 1, \dots, 5$ e o coeficiente de variação cv dos 20 problemas correspondentes. Os resultados são disponibilizados nas tabelas que seguem, separados por cada métrica.

Inicialmente, na Tabela (17), apresentamos os resultados da Métrica σ^1 , que é a distância Euclidiana média de cada solução de \bar{Z} em relação à Z^* . Como podemos notar, esta distância média geral foi igual à 0.42, crescente com a instância do problema. No entanto, em termos relativos este valor é baixo, demonstrando que as fronteiras exata e heurística estão próximas. Notamos também que nos PCUIM cujos itens são pequenos, estas distâncias foram maiores que nas instâncias com itens grandes.

Tabela 17 – Resultados computacionais para σ^1 - Fase 2

Classe	SP		Tch		Tch-SP		$\varepsilon-R$		NISE+ $\varepsilon-R$		Ben		S-C		Média
	σ^1	cv	σ^1	cv	σ^1	cv	σ^1	cv	σ^1	cv	σ^1	cv	σ^1	cv	
1	1.66	1.58	1.18	1.54	0.08	1.19	0.07	1.23	0.07	1.23	0.07	1.23	0.06	1.36	0.45
2	0.04	2.97	0.21	1.88	0.02	4.27	0.02	4.27	0.02	4.27	0.02	4.27	0.02	4.27	0.05
3	0.64	2.24	0.74	2.11	0.01	4.36	0.01	4.36	0.01	4.36	0.01	4.36	0.01	4.36	0.20
4	0.00	0.00	0.00	0.00	0.07	2.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
5	0.16	2.93	0.08	3.17	0.06	3.66	0.06	3.66	0.06	3.66	0.06	3.66	0.06	3.66	0.08
6	0.53	4.14	0.60	3.62	0.06	2.09	0.04	2.46	0.03	2.36	0.04	2.46	0.03	2.36	0.19
7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	0.45	4.36	0.28	4.35	0.04	3.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11
9	0.22	4.05	0.18	2.92	0.03	3.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06
Média	0.41	2.62	0.36	2.43	0.04	2.54	0.02	2.62	0.02	2.74	0.02	2.62	0.02	2.83	0.13
10	0.93	1.74	1.02	1.29	0.06	1.60	0.05	1.48	0.05	1.48	0.05	1.48	0.05	1.48	0.32
11	1.46	2.02	0.76	1.27	0.10	1.24	0.09	1.32	0.09	1.32	0.09	1.32	0.08	1.40	0.38
12	1.23	2.05	0.95	1.25	0.08	1.61	0.08	1.61	0.08	1.61	0.08	1.61	0.07	1.92	0.37
13	0.51	2.47	0.89	1.99	0.14	1.02	0.06	2.27	0.06	2.27	0.06	2.27	0.06	2.30	0.26
14	1.05	2.97	2.57	1.65	0.04	1.83	0.03	2.02	0.04	1.95	0.04	1.95	0.04	2.13	0.54
15	0.68	3.60	1.40	1.98	0.10	1.69	0.08	2.27	0.06	1.96	0.06	1.96	0.05	2.18	0.35
16	0.30	3.23	1.23	3.17	0.04	2.02	0.00	0.00	0.01	4.12	0.00	0.00	0.00	0.00	0.23
17	0.13	2.11	1.38	1.89	0.09	2.53	0.07	3.12	0.07	3.12	0.08	2.84	0.07	3.12	0.27
18	0.41	2.44	0.15	2.01	0.15	2.01	0.13	2.32	0.10	3.00	0.11	2.66	0.10	3.00	0.17
Média	0.74	2.41	1.15	1.84	0.09	1.67	0.07	2.07	0.06	2.15	0.06	2.06	0.06	2.25	0.32
19	3.59	0.92	2.09	0.42	0.25	0.79	0.25	0.79	0.25	0.79	0.25	0.79	0.25	0.79	0.99
20	2.98	1.33	1.16	0.81	0.17	0.75	0.17	0.75	0.17	0.75	0.17	0.76	0.67	3.25	0.79
21	2.94	1.26	1.67	0.52	0.22	0.71	0.22	0.71	0.22	0.71	0.22	0.71	2.30	2.17	1.11
22	1.28	1.83	4.56	1.11	0.22	1.14	0.15	1.05	0.14	1.01	0.15	1.00	0.33	2.74	0.98
23	2.37	1.50	5.04	0.84	0.22	0.85	0.20	0.83	0.21	0.78	0.18	0.93	0.18	0.95	1.20
24	2.25	1.90	3.38	0.97	0.26	0.88	0.21	0.87	0.21	0.86	0.21	0.86	0.19	0.93	0.96
25	0.51	2.03	1.08	3.52	0.08	1.15	0.04	2.55	0.04	2.60	0.04	2.55	0.04	2.55	0.26
26	0.63	2.20	1.97	2.55	0.08	1.10	0.03	2.55	0.03	2.55	0.03	2.68	0.03	2.64	0.40
27	0.55	1.05	3.08	1.34	0.20	1.18	0.19	1.00	0.18	1.02	0.19	1.02	0.19	0.98	0.65
Média	1.90	1.41	2.67	1.17	0.19	0.93	0.16	0.92	0.16	0.91	0.16	0.94	0.46	2.15	0.82
Média Geral	1.02	1.82	1.39	1.47	0.11	1.34	0.08	1.38	0.08	1.39	0.08	1.38	0.18	2.19	0.42

Na Tabela (18), disponibilizamos as médias das proporções das soluções da fronteira heurística que não coincidiram com os pontos da fronteira ótima. Em média, considerando-se todos os métodos de escalarização, 9% das soluções de \bar{Z} não coincidiram com as de Z^* , o que é um resultado muito bom. Assim, o erro que se comete ao utilizar a HADS para integralizar as soluções do PCUIM parcialmente relaxado é pequeno, baseando nessa média de 91% de soluções coincidentes e um erro de 0.42 em valores absolutos das 9% não coincidentes. Em relação ao tipo de método utilizado, apenas Soma Ponderada e Métrica

de Tchebycheff ficaram acima da média, onde obtiveram 87% e 81% de soluções coincidentes com as ótimas, respectivamente. Os demais métodos tiveram resultados similares. Vale ressaltar que esse percentual foi sendo modificado à medida que m aumentou, como era de se esperar. Nas instâncias com $m = 40$, há 81% de pontos coincidentes em média, mas considerando a escalarização pelo Tchebycheff Sem Ponderações, esse percentual cai para apenas 68%.

Tabela 18 – Resultados computacionais para σ^2 - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ^2	cv	σ^2	cv	σ^2	cv	σ^2	cv	σ^2	cv	σ^2	cv	σ^2	cv	
1	0.16	0.89	0.22	0.81	0.09	1.09	0.06	1.22	0.06	1.22	0.06	1.22	0.06	1.36	0.07
2	0.03	3.02	0.07	1.64	0.02	4.27	0.02	4.27	0.02	4.27	0.02	4.27	0.02	4.27	0.05
3	0.11	1.45	0.12	1.43	0.01	4.36	0.01	4.36	0.01	4.36	0.01	4.36	0.01	4.36	0.03
4	0.00	0.00	0.00	0.00	0.07	2.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.05	2.58	0.05	2.99	0.05	2.53	0.04	3.28	0.04	3.28	0.04	3.28	0.04	2.74	0.08
6	0.04	2.38	0.09	2.10	0.06	2.09	0.04	2.46	0.03	2.36	0.04	2.46	0.03	2.36	0.07
7	0.00	0.00	0.00	0.00	0.02	4.27	0.00	0.00	0.00	0.00	0.02	4.27	0.00	0.00	0.01
8	0.02	4.27	0.02	4.27	0.04	3.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9	0.02	2.94	0.06	2.55	0.03	3.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Média	0.05	1.77	0.02	2.54	0.07	1.63	0.04	2.38	0.02	2.51	0.02	2.69	0.02	2.51	0.03
10	0.16	0.89	0.21	0.83	0.06	1.45	0.05	1.49	0.05	1.49	0.05	1.49	0.05	1.49	0.06
11	0.17	0.93	0.21	0.93	0.10	1.24	0.09	1.32	0.09	1.32	0.09	1.32	0.08	1.40	0.11
12	0.18	0.79	0.22	0.79	0.07	1.32	0.07	1.32	0.07	1.32	0.07	1.32	0.05	1.57	0.08
13	0.12	1.65	0.14	1.05	0.14	1.02	0.06	2.27	0.06	2.27	0.06	2.27	0.06	2.30	0.11
14	0.13	1.15	0.22	1.03	0.04	1.83	0.03	2.02	0.04	1.95	0.04	1.95	0.04	2.13	0.06
15	0.09	1.88	0.17	1.41	0.09	1.53	0.06	2.15	0.05	1.95	0.05	1.95	0.04	2.26	0.07
16	0.06	2.25	0.08	2.60	0.04	2.02	0.00	0.00	0.01	4.12	0.00	0.00	0.00	0.00	0.01
17	0.09	2.58	0.15	1.62	0.09	2.53	0.07	3.12	0.07	3.12	0.08	2.84	0.07	3.12	0.16
18	0.15	2.03	0.15	2.01	0.15	2.01	0.13	2.32	0.10	3.00	0.11	2.66	0.10	3.00	0.22
Média	0.13	1.41	0.06	2.01	0.17	1.23	0.09	1.61	0.06	2.16	0.06	2.03	0.06	2.23	0.09
19	0.30	0.53	0.40	0.25	0.21	0.81	0.21	0.81	0.21	0.81	0.21	0.81	0.21	0.81	0.19
20	0.21	0.51	0.28	0.37	0.15	0.71	0.15	0.71	0.15	0.71	0.15	0.71	0.16	0.81	0.13
21	0.27	0.50	0.37	0.38	0.20	0.72	0.20	0.72	0.20	0.72	0.20	0.72	0.26	0.84	0.19
22	0.19	0.96	0.36	0.68	0.18	0.96	0.15	1.04	0.14	0.98	0.15	0.98	0.14	1.01	0.14
23	0.25	0.55	0.46	0.44	0.19	0.77	0.17	0.87	0.18	0.84	0.17	0.93	0.17	0.96	0.16
24	0.20	0.72	0.37	0.60	0.21	0.80	0.18	0.79	0.18	0.79	0.18	0.79	0.17	0.85	0.16
25	0.06	1.58	0.09	1.53	0.08	1.14	0.04	2.51	0.04	2.57	0.04	2.51	0.04	2.51	0.07
26	0.12	1.07	0.14	1.44	0.08	1.11	0.03	2.55	0.03	2.55	0.03	2.68	0.03	2.64	0.05
27	0.23	0.78	0.36	0.59	0.17	0.96	0.17	0.95	0.17	0.96	0.17	0.97	0.17	0.92	0.16
Média	0.20	0.69	0.14	0.92	0.32	0.56	0.16	0.85	0.14	0.91	0.14	0.93	0.15	0.95	0.18
Média Geral	0.13	1.07	0.19	0.90	0.10	1.29	0.08	1.36	0.07	1.36	0.08	1.38	0.07	1.40	0.09

No cálculo da Métrica 3, como os métodos Soma Ponderada, Métrica de Tchebycheff, Tchebycheff Sem Ponderações, ε -Restrito e Benson utilizaram exatamente os mesmos pontos lexicográficos, as amplitudes foram disponibilizadas numa só coluna para estes métodos. A única técnica que não os determina para guiar no processo de obtenção das demais soluções é o Método Sylva & Crema e por isso, os resultados referentes à esta metodologia foram postos em uma coluna à parte.

Notamos uma similaridade dos pontos lexicográficos encontrados explicitamente e as soluções lexicográficas encontradas pelo Método Sylva & Crema, onde os determina sem uma ordem pré-determinada e implícita, demonstrando matematicamente uma boa performance desta técnica. Há também uma grande proximidade entre os pontos lexicográficos exatos e heurísticos encontrados, como já era esperado pelos resultados de σ^1 . Além disso, notamos diferentes tipos de fronteiras de Pareto à medida que a classe é modificada, principalmente aquelas que se diferenciam quanto ao tamanho dos itens.

Tabela 19 – Resultados computacionais para σ_{11}^3 , σ_{12}^3 , σ_{21}^3 e σ_{22}^3 - Fase 2

Classe	SP, Tch, Tch-SP, ε -R, NISE+ ε -R, Ben				S-C				Média σ_{*1}^3	Média σ_{*2}^3
	σ_{11}^3	σ_{12}^3	σ_{21}^3	σ_{22}^3	σ_{11}^3	σ_{12}^3	σ_{21}^3	σ_{22}^3		
1	160.59	5.18	160.75	5.12	160.87	5.17	160.40	5.15	160.65	5.15
2	143.55	5.39	143.45	5.35	143.58	5.35	143.10	5.33	143.42	5.37
3	126.92	5.31	126.08	5.25	126.55	5.30	126.13	5.27	126.42	5.28
4	60.29	2.50	60.23	2.48	60.15	2.52	60.94	2.45	60.40	2.49
5	77.76	3.40	77.91	3.30	77.85	3.43	77.96	3.29	77.87	3.35
6	89.75	3.08	89.15	3.06	89.62	3.10	89.58	3.09	89.53	3.07
7	50.38	2.71	50.83	2.63	50.35	2.70	50.06	2.71	50.40	2.66
8	88.57	2.70	88.54	2.66	88.51	2.72	88.23	2.70	88.46	2.68
9	95.08	2.89	96.00	2.85	95.40	2.89	95.35	2.85	95.46	2.87
Média	99.21	3.68	99.22	3.64	99.21	3.67	99.08	3.65	99.18	3.66
10	290.05	9.71	290.08	9.65	290.08	9.70	290.82	9.68	290.26	9.68
11	270.53	9.42	270.44	9.38	270.24	9.40	270.02	9.39	270.31	9.40
12	340.78	9.53	340.11	9.43	340.12	9.51	340.04	9.51	340.26	9.48
13	154.93	5.69	154.96	5.65	154.18	5.65	154.17	5.63	154.56	5.67
14	165.13	6.96	165.00	6.90	165.24	6.94	165.65	6.90	165.26	6.93
15	150.57	6.79	150.77	6.75	150.42	6.70	150.73	6.72	150.62	6.77
16	105.47	4.16	105.82	4.10	105.05	4.16	105.65	4.15	105.50	4.13
17	153.01	6.79	153.87	6.75	153.90	6.79	153.45	6.75	153.56	6.77
18	131.34	5.74	131.08	5.70	131.94	5.74	131.55	5.75	131.48	5.72
Média	195.76	7.19	195.79	7.15	195.34	7.17	195.75	7.15	195.88	7.18
19	550.16	18.92	550.40	17.92	550.49	18.90	550.30	18.89	550.34	18.00
20	590.79	18.04	590.26	17.90	600.49	18.02	590.74	18.05	593.07	17.97
21	530.31	16.94	530.80	16.90	550.34	16.93	530.19	16.91	535.41	16.92
22	360.53	13.96	360.43	13.90	360.90	13.90	360.69	13.89	360.64	13.93
23	390.17	16.03	390.91	16.01	390.37	16.01	390.18	15.98	390.41	16.02
24	350.60	14.86	350.18	14.80	350.11	14.80	350.37	14.85	350.32	14.83
25	290.26	10.89	290.26	10.85	290.78	10.92	290.63	10.90	290.48	10.87
26	290.65	11.06	290.15	10.98	290.39	11.04	290.78	10.98	290.49	11.02
27	320.69	14.54	320.14	14.50	310.24	14.51	310.08	14.50	315.29	14.52
Média	408.24	14.95	408.17	14.85	410.46	14.93	407.11	14.91	408.49	14.90
Média Geral	234.40	8.60	234.39	8.58	235.12	8.59	233.99	8.60	234.48	8.58

Classes com itens pequenos têm uma amplitude maior em ambas funções-objetivo; classes com itens de variados tamanhos possuem amplitude intermediária e finalmente as classes com itens grandes possuem uma amplitude bastante reduzida em ambas dimensões de \mathcal{Z} . Esse aspecto da fronteira tem enorme interferência no nível de dificuldade dos algoritmos para solucionarem o PCUIM. Noutras palavras, classes com pequena amplitude em z_2 são mais rápidas de serem resolvidas do aquelas com maior amplitude, pois um maior número de Problema de Programação Linear Inteiros serão exigidos. Por exemplo, em média 3.66, 7.18 e 14.90 Problemas Restritos foram resolvidos para as instâncias com $m = 10, 20$ e 40 , respectivamente. Além disso, a diferença de amplitude entre z_1 e z_2 foi fundamental para justificar a implementação do ε -Restrito e Benson no PCUIM.

A Tabela (20) ilustra as médias da Métrica σ^4 em relação à fronteira exata \mathcal{Z}^* . A última linha denota a média geral de cada método e notoriamente os métodos Métrica de Tchebycheff, ε -Restrito, NISE híbrido com ε -Restrito e Sylva & Crema delimitaram aproximadamente a mesma área, ao considerar os 540 PCUIM resolvidos, mas com ligeira vantagem do Método Tchebycheff Sem Ponderações. Os métodos que usam ponderações, Soma Ponderada e Métrica de Tchebycheff, foram os que alcançaram novamente os piores resultados, tendo em vista que muitos pontos não dominados não foram obtidos, interferindo de forma direta nesse resultado.

Tabela 20 – Resultados computacionais para $\sigma_1^4 \times 10^{-4}$ - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ_1^4	cv	σ_1^4	cv	σ_1^4	cv	σ_1^4	cv	σ_1^4	cv	σ_1^4	cv	σ_1^4	cv	
1	0.14	0.71	0.23	0.43	0.22	0.77	0.22	0.77	0.22	0.77	0.22	0.77	0.22	0.77	0.21
2	0.14	0.71	0.15	0.67	0.2	0.65	0.2	0.65	0.2	0.65	0.2	0.65	0.2	0.65	0.19
3	0.13	0.77	0.13	0.77	0.17	0.65	0.21	0.19	0.17	0.65	0.17	0.65	0.17	0.65	0.17
4	0.02	1.00	0.02	1.00	0.04	1.00	0.03	1.33	0.03	1.33	0.03	1.33	0.03	1.33	0.03
5	0.13	0.77	0.03	3.32	0.06	0.83	0.06	0.83	0.06	0.83	0.06	0.83	0.06	0.83	0.06
6	0.05	1.20	0.04	1.50	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.06
7	0.02	1.00	0.01	1.98	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02
8	0.05	1.00	0.04	1.25	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.05
9	0.05	1.00	0.04	1.25	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.06
Média	0.08	0.87	0.08	0.87	0.1	0.80	0.1	0.70	0.1	0.80	0.1	0.80	0.1	0.80	0.1
10	0.8	0.64	0.91	0.56	1.23	0.57	1.21	0.58	1.21	0.58	1.21	0.58	1.21	0.58	1.11
11	0.59	0.53	0.8	0.39	1.09	0.66	1.08	0.67	1.08	0.67	1.08	0.67	1.08	0.67	0.97
12	0.88	0.55	1.08	0.44	1.45	0.51	1.45	0.51	1.45	0.51	1.45	0.51	1.45	0.52	1.31
13	0.18	0.89	0.17	0.94	0.36	0.97	0.33	1.06	0.33	1.06	0.33	1.06	0.33	1.06	0.29
14	0.3	0.87	0.24	1.08	0.48	0.90	0.47	0.91	0.47	0.91	0.47	0.91	0.47	0.91	0.41
15	0.24	1.00	0.22	1.09	0.06	1.83	0.42	1.21	0.06	1.83	0.06	1.83	0.06	1.83	0.16
16	0.08	1.25	0.06	1.66	0.15	1.00	0.13	1.08	0.13	1.08	0.13	1.08	0.13	1.08	0.12
17	0.24	1.00	0.16	1.50	0.35	0.86	0.34	0.88	0.34	0.88	0.34	0.88	0.34	0.88	0.3
18	0.15	0.87	0.24	0.54	0.24	0.79	0.24	0.79	0.24	0.79	0.24	0.79	0.24	0.79	0.23
Média	0.39	0.69	0.43	0.63	0.6	0.68	0.63	0.73	0.59	0.69	0.59	0.69	0.59	0.69	0.54
19	4.34	0.43	5.22	0.36	7.15	0.43	7.15	0.43	7.15	0.43	7.15	0.43	7.15	0.43	6.47
20	4.25	0.37	6.09	0.26	7.89	0.42	7.89	0.42	7.89	0.42	7.89	0.42	7.89	0.42	7.14
21	3.68	0.37	4.87	0.28	6.7	0.45	6.7	0.45	6.7	0.45	6.7	0.45	6.61	0.40	6.0
22	1.74	0.70	1.73	0.71	3.73	0.75	3.29	0.81	3.29	0.81	3.29	0.81	3.43	0.84	2.93
23	2.46	0.54	2.18	0.61	4.56	0.55	4.41	0.58	4.49	0.57	4.26	0.58	4.25	0.58	3.8
24	2.14	0.58	1.89	0.66	3.88	0.64	3.68	0.67	3.68	0.67	3.68	0.67	3.66	0.68	3.23
25	0.91	0.78	0.74	0.96	1.67	0.77	1.58	0.81	1.58	0.81	1.58	0.81	1.58	0.81	1.38
26	1.07	0.70	0.77	0.97	1.81	0.68	1.68	0.71	1.68	0.71	1.68	0.71	1.68	0.71	1.48
27	1.68	0.67	1.46	0.77	3.05	0.62	3.02	0.63	3.02	0.63	3.02	0.63	3.02	0.63	2.61
Média	2.47	0.50	2.77	0.45	4.49	0.54	4.38	0.55	4.39	0.54	4.36	0.55	4.38	0.55	3.89
Média Geral	0.98	0.54	1.09	0.49	1.73	0.56	1.7	0.57	1.69	0.57	1.68	0.57	1.69	0.57	1.51

O mesmo cálculo foi realizado para as fronteiras \bar{Z} , cujos resultados estão na Tabela (21). Observamos que as médias das áreas delimitadas por Z^* é de apenas 4.8% maior em relação a área delimitada por \bar{Z} , mostrando e confirmando a afirmação de que estas fronteiras estão próximas.

O número de soluções não dominadas determinadas, $\sigma_1^5 = |Z^*|$, foi computado e estão presentes na Tabela (22). Os resultados mostram que as instâncias com itens pequenos possuem mais soluções eficientes que aquelas com itens grandes. Além disso, em média, o aumento do número de soluções é aproximadamente proporcional à m .

Em relação ao métodos de escalarização propostos, ε -Restrito, Tchebycheff Sem Ponderações, NISE híbrido com ε -Restrito, Benson e Sylva & Crema forneceram em torno de 9.5 soluções eficientes, com ligeira vantagem do Método Tchebycheff Sem Ponderações. Em relação à Soma Ponderada, onde apenas algumas soluções suportadas são captadas, obtivemos de 5.8 soluções, representando 61% do número total de soluções do problema. O Método de Métrica de Tchebycheff, é capaz de captar alguns pontos suportados e não suportados dependendo da escolha de w e com esta habilidade determinou 6.5% pontos a mais em relação à Soma Ponderada, confirmando uma prévia expectativa que havíamos feito nos capítulos anteriores e quantificando a limitação dessas duas técnicas.

A mesma análise pode ser feita utilizando a fronteira \bar{Z} , cuja conclusões são bastantes similares, utilizando com referência os resultados apresentados na Tabela (23).

Tabela 21 – Resultados computacionais para $\sigma_2^4 \times 10^{-4}$ - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ_2^4	cv	σ_2^4	cv	σ_2^4	cv	σ_2^4	cv	σ_2^4	cv	σ_2^4	cv	σ_2^4	cv	
1	0.15	0.67	0.16	0.75	0.23	0.74	0.22	0.77	0.22	0.77	0.22	0.77	0.22	0.77	0.19
2	0.13	0.61	0.15	0.67	0.2	0.65	0.2	0.65	0.2	0.65	0.2	0.65	0.2	0.65	0.16
3	0.13	0.69	0.13	0.61	0.17	0.59	0.17	0.59	0.17	0.59	0.17	0.59	0.02	1.00	0.1
4	0.03	1.33	0.03	1.33	0.03	1.33	0.02	1.00	0.03	1.33	0.03	1.33	0.03	1.33	0.03
5	0.04	0.75	0.03	1.00	0.06	0.83	0.06	0.83	0.06	0.83	0.06	0.83	0.06	0.83	0.05
6	0.05	1.20	0.04	1.25	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.07	1.14	0.07
7	0.02	1.00	0.01	0.99	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02	1.00	0.02
8	0.05	1.00	0.04	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06	1.00	0.06
9	0.05	1.00	0.04	1.25	0.07	1.00	0.07	1.00	0.07	1.00	0.07	1.00	0.07	1.00	0.07
Média	0.07	0.86	0.07	0.86	0.1	0.80	0.1	0.80	0.1	0.80	0.1	0.80	0.08	0.87	0.08
10	0.79	0.59	0.93	0.59	1.22	0.57	1.2	0.57	1.2	0.57	1.2	0.57	1.2	0.57	0.91
11	0.63	0.57	0.81	0.72	1.09	0.65	1.08	0.66	1.08	0.66	1.08	0.66	1.07	0.64	0.86
12	0.89	0.57	1.11	0.54	1.43	0.52	1.43	0.52	1.43	0.52	1.43	0.52	1.4	0.52	1.03
13	0.18	0.89	0.17	1.06	0.36	0.94	0.33	1.06	0.33	1.06	0.33	1.06	0.33	1.06	0.34
14	0.3	0.80	0.25	1.04	0.47	0.91	0.46	0.96	0.46	0.96	0.46	0.96	0.46	0.96	0.45
15	0.23	0.96	0.22	1.47	0.44	1.16	0.41	1.24	0.41	1.24	0.41	1.24	0.41	1.24	0.47
16	0.08	1.25	0.06	1.00	0.14	1.07	0.13	1.08	0.13	1.08	0.13	1.08	0.13	1.08	0.14
17	0.23	0.96	0.16	0.94	0.35	0.86	0.34	0.88	0.34	0.88	0.34	0.88	0.34	0.88	0.32
18	0.14	0.86	0.24	0.79	0.24	0.79	0.23	0.83	0.23	0.83	0.23	0.78	0.23	0.78	0.21
Média	0.39	0.69	0.43	0.98	0.64	0.70	0.62	0.73	0.62	0.73	0.62	0.73	0.62	0.73	0.49
19	4.26	0.45	5.14	0.46	7.03	0.43	7.03	0.43	7.76	0.42	7.76	0.39	7.03	0.43	4.98
20	4.37	0.36	6.1	0.44	7.76	0.42	7.76	0.42	6.57	0.44	6.57	0.49	8.03	0.48	4.91
21	3.75	0.39	4.92	0.42	6.57	0.44	6.57	0.44	3.26	0.81	3.26	0.88	6.92	0.50	3.61
22	1.74	0.69	1.73	0.87	3.69	0.75	3.27	0.81	4.45	0.58	4.23	0.63	3.41	0.85	3.27
23	2.46	0.56	2.23	0.59	4.52	0.56	4.38	0.58	3.61	0.65	3.61	0.68	4.21	0.58	3.04
24	2.12	0.60	1.9	0.72	3.8	0.62	3.61	0.65	1.58	0.81	1.58	1.49	3.59	0.65	2.15
25	0.93	0.75	0.74	0.88	1.66	0.77	1.58	0.81	1.67	0.72	1.67	0.77	1.58	0.81	1.42
26	1.05	0.70	0.77	0.74	1.67	0.72	1.67	0.72	3	0.63	3	0.40	1.67	0.72	1.88
27	1.65	0.67	1.41	0.69	3.04	0.62	3	0.63	4.33	0.54	4.3	0.44	3	0.63	2.81
Média	2.48	0.51	2.75	0.54	4.42	0.53	4.32	0.54	4.03	0.56	4.0	0.58	4.38	0.57	3.76
Média Geral	0.98	0.54	1.08	0.63	1.72	0.56	1.68	0.57	1.58	0.57	1.57	0.53	1.69	0.59	1.44

Tabela 22 – Resultados computacionais para σ_1^5 - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	
1	4.25	0.15	4.6	0.11	6.2	0.13	6.1	0.13	6.1	0.13	6.1	0.13	6.1	0.13	5.64
2	4.6	0.27	4.8	0.20	6.2	0.19	6.2	0.19	6.2	0.19	6.2	0.19	6.2	0.19	5.77
3	4.65	0.24	4.8	0.19	6.35	0.11	6.35	0.11	6.35	0.11	6.35	0.11	6.35	0.11	5.89
4	2.5	0.46	2.35	0.41	3.5	0.43	3.1	0.54	3.3	0.44	3.3	0.44	3.1	0.54	3.02
5	3.35	0.24	3.15	0.23	4.3	0.31	4.3	0.31	4.3	0.31	4.3	0.31	4.3	0.31	4
6	3.1	0.35	2.85	0.30	4.15	0.33	4.05	0.35	4.1	0.33	4.1	0.33	4.05	0.35	3.77
7	2.5	0.45	2.2	0.40	2.9	0.34	2.65	0.48	2.9	0.34	2.9	0.34	2.65	0.48	2.67
8	3.25	0.36	2.9	0.32	3.9	0.34	3.75	0.38	3.8	0.36	3.8	0.36	3.75	0.38	3.59
9	3.25	0.30	2.9	0.27	4.1	0.33	4	0.33	4	0.33	4	0.33	4	0.33	3.75
Média	3.49	0.30	3.39	0.25	4.62	0.25	4.5	0.28	4.56	0.26	4.56	0.26	4.5	0.28	4.23
10	6.8	0.18	7.8	0.17	10.8	0.16	10.75	0.16	10.75	0.16	10.75	0.16	10.75	0.16	9.77
11	6.2	0.14	7.55	0.17	10.35	0.17	10.3	0.17	10.3	0.17	10.3	0.17	10.3	0.17	9.33
12	6.6	0.17	7.9	0.15	10.75	0.11	10.75	0.11	10.75	0.11	10.75	0.11	10.75	0.11	9.75
13	4.45	0.28	4.15	0.33	7.05	0.35	6.7	0.39	6.7	0.39	6.7	0.39	6.7	0.39	6.06
14	5.5	0.25	4.7	0.27	8.05	0.29	8.05	0.29	8.05	0.29	8.05	0.29	8.05	0.29	7.21
15	5.15	0.24	4.6	0.28	7.9	0.29	7.7	0.31	7.7	0.31	7.7	0.31	7.7	0.31	6.92
16	3.35	0.41	3.15	0.30	5.35	0.39	5.1	0.43	5.2	0.39	5.2	0.39	5.1	0.43	4.64
17	5.5	0.25	4.45	0.18	7.85	0.20	7.75	0.21	7.75	0.21	7.75	0.21	7.75	0.21	6.97
18	4.6	0.28	6.85	0.27	6.85	0.27	6.8	0.27	6.8	0.27	6.8	0.27	6.8	0.27	6.5
Média	5.35	0.23	5.68	0.22	8.33	0.23	8.21	0.24	8.22	0.24	8.22	0.24	8.21	0.24	7.46
19	10.15	0.11	12.8	0.11	18.3	0.10	18.3	0.10	18.3	0.10	18.3	0.10	18.3	0.10	16.35
20	9.65	0.14	13.55	0.14	18.8	0.13	18.8	0.13	18.8	0.13	18.8	0.13	18.8	0.13	16.74
21	9.55	0.09	12.6	0.13	17.95	0.13	17.95	0.13	17.95	0.13	17.95	0.13	17.65	0.12	15.94
22	7.75	0.19	8.25	0.28	15.15	0.30	14.8	0.31	14.8	0.31	14.8	0.31	14.8	0.31	12.91
23	8.65	0.16	9.15	0.18	17	0.16	16.85	0.17	16.85	0.17	16.85	0.17	16.85	0.17	14.6
24	8.6	0.18	8.6	0.25	15.9	0.23	15.7	0.23	15.7	0.23	15.7	0.23	15.7	0.23	13.7
25	7.05	0.21	6.35	0.28	12.05	0.24	11.75	0.26	11.75	0.26	11.75	0.26	11.75	0.26	10.35
26	7.6	0.24	6.2	0.31	12.25	0.27	11.95	0.29	11.95	0.29	11.95	0.29	11.95	0.29	10.55
27	8.4	0.19	8	0.23	15.5	0.17	15.4	0.19	15.4	0.19	15.4	0.19	15.4	0.19	13.36
Média	8.6	0.16	9.5	0.19	15.88	0.18	15.72	0.19	15.72	0.19	15.72	0.19	15.69	0.19	13.83
Média Geral	5.81	0.21	6.19	0.21	9.61	0.21	9.48	0.22	9.5	0.22	9.5	0.22	9.47	0.22	8.51

Em média, para $m = 10$ tivemos 4.12 soluções heurísticas contra 4.23 exatas; para $m = 20$ obtivemos 7.34 soluções heurísticas contra 7.46 exatas; e finalmente para $m = 40$, obtivemos 13.57 soluções heurísticas e 13.83 exatas. Isso permite concluir que o arredondamento das soluções eficientes relaxadas e posteriormente a aplicação do filtro de Pareto promove uma perda média de aproximadamente 2% da soluções em relação à quantidade de soluções original.

Tabela 23 – Resultados computacionais para σ_2^5 - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ_2^5	cv	σ_2^5	cv	σ_2^5	cv	σ_2^5	cv	σ_2^5	cv	σ_2^5	cv	σ_2^5	cv	
1	4.5	0.04	4.65	0.03	6.1	0.02	6.1	0.02	6.1	0.02	6.1	0.02	6.05	0.02	5.66
2	4.4	0.05	4.85	0.04	6.1	0.03	6.1	0.03	6.2	0.03	6.1	0.03	6.1	0.03	5.68
3	4.55	0.05	4.7	0.04	6	0.02	6	0.02	6.35	0.02	6	0.02	6	0.02	5.61
4	2.4	0.19	2.3	0.17	3.1	0.17	3.1	0.17	3.3	0.15	3.1	0.17	3.1	0.17	2.89
5	3.4	0.07	3.15	0.07	4.25	0.07	4.25	0.07	4.3	0.07	4.25	0.07	4.25	0.07	3.97
6	3	0.11	2.9	0.10	4	0.09	4	0.09	4.1	0.08	4	0.09	4	0.09	3.7
7	2.45	0.17	2.15	0.17	2.6	0.19	2.65	0.18	2.9	0.15	2.65	0.18	2.65	0.18	2.54
8	3.15	0.10	2.8	0.12	3.65	0.10	3.65	0.10	3.8	0.10	3.65	0.10	3.65	0.10	3.46
9	3.05	0.09	2.85	0.11	3.8	0.09	3.8	0.09	4	0.08	3.8	0.09	3.8	0.09	3.56
Média	3.43	0.08	3.37	0.07	4.4	0.06	4.41	0.06	4.56	0.06	4.41	0.06	4.4	0.06	4.12
10	6.95	0.03	8	0.02	10.6	0.02	10.6	0.02	10.75	0.02	10.6	0.02	10.55	0.02	9.7
11	6.5	0.03	7.75	0.02	10.3	0.02	10.3	0.02	10.3	0.02	10.3	0.02	10.25	0.02	9.39
12	6.55	0.03	7.9	0.02	10.35	0.01	10.35	0.01	10.75	0.01	10.35	0.01	10.25	0.01	9.44
13	4.55	0.06	4.15	0.07	6.6	0.06	6.6	0.06	6.7	0.06	6.6	0.06	6.65	0.06	5.96
14	5.45	0.04	4.7	0.06	7.8	0.04	7.75	0.04	8.05	0.04	7.75	0.04	7.75	0.04	6.99
15	5	0.05	4.8	0.06	7.65	0.04	7.65	0.04	7.7	0.04	7.6	0.04	7.6	0.04	6.84
16	3.45	0.12	3.2	0.10	5.1	0.08	5.05	0.08	5.2	0.08	5.05	0.08	5.05	0.08	4.57
17	5.25	0.05	4.45	0.04	7.7	0.03	7.7	0.03	7.75	0.03	7.7	0.03	7.7	0.03	6.89
18	4.4	0.07	6.6	0.04	6.6	0.04	6.55	0.04	6.8	0.04	6.5	0.04	6.5	0.04	6.24
Média	5.34	0.04	5.73	0.04	8.08	0.03	8.06	0.03	8.22	0.03	8.05	0.03	8.03	0.03	7.34
19	10.05	0.01	13.6	0.01	17.95	0.01	17.95	0.01	18.3	0.01	17.95	0.01	17.95	0.01	16.2
20	9.95	0.01	13.6	0.01	18.25	0.01	18.25	0.01	18.8	0.01	18.2	0.01	18.25	0.01	16.39
21	9.75	0.01	12.7	0.01	17.15	0.01	17.15	0.01	17.95	0.01	17.15	0.01	17.15	0.01	15.46
22	7.65	0.03	8.05	0.04	14.5	0.02	14.5	0.02	14.8	0.02	14.45	0.02	14.45	0.02	12.58
23	8.65	0.02	9.15	0.02	16.45	0.01	16.5	0.01	16.85	0.01	16.35	0.01	16.3	0.01	14.3
24	8.55	0.02	8.6	0.03	15.15	0.01	15.2	0.01	15.7	0.01	15.2	0.01	15.05	0.01	13.28
25	7.2	0.03	6.3	0.04	11.75	0.02	11.75	0.02	11.75	0.02	11.75	0.02	11.75	0.02	10.32
26	7.55	0.03	6.3	0.05	11.95	0.02	11.9	0.02	11.95	0.02	11.9	0.02	11.85	0.02	10.48
27	8.3	0.02	7.8	0.03	15.2	0.01	15.15	0.01	15.4	0.01	15.2	0.01	15.2	0.01	13.15
Média	8.63	0.02	9.57	0.02	15.37	0.01	15.37	0.01	15.72	0.01	15.35	0.01	15.33	0.01	13.57
Média Geral	5.8	0.04	6.22	0.03	9.28	0.02	9.28	0.02	9.5	0.02	9.27	0.02	9.25	0.02	8.34

O mais interessante é o tempo computacional t^* levado pelas duas abordagens, um dos resultados mais evidentes deste trabalho. A Tabela (24) ilustra o tempo computacional total² médio (em seg.) que cada método de escalarização usou em cada classe.

Ficou evidente que a Soma Ponderada foi o método mais rápido e há um motivo para esse fato: seus subproblemas diferem um dos outros por uma simples ponderação na função-objetivo, mantendo a estrutura de restrições inalterada ao longo do procedimento. Embora a quantidade de subproblemas a ser resolvida é exatamente a mesma que nas demais técnicas, concluímos que esses subproblemas são mais facilmente resolvidos pelo CPLEX.

Uma análise do tempo computacional de cada subproblema ponderado é ilustrado na Figura (41), em um experimento particular com um problema com $m = 100$. O

² Nesse tempo computacional, não é incluso o tempo para gerar as colunas de Gilmore e Gomory.

Tabela 24 – Tempos computacionais médios (em seg.) para obtenção de \mathcal{Z}^* - Fase 2

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	t^*	cv	t^*	cv	t^*	cv	t^*	cv	t^*	cv	t^*	cv	t^*	cv	
1	0.23	0.39	0.29	0.28	0.15	1.53	0.22	0.32	0.4	0.35	0.23	0.43	0.53	0.34	0.46
2	0.33	0.30	0.4	0.30	0.2	2.25	0.31	0.45	0.61	0.41	0.34	0.38	0.73	0.55	0.63
3	0.36	0.42	0.45	0.36	0.21	2.38	0.33	0.39	0.61	0.39	0.42	0.36	0.84	0.38	0.69
4	0.11	0.27	0.13	0.38	0.08	2.62	0.12	0.50	0.13	0.54	0.13	0.46	0.16	0.62	0.21
5	0.15	0.40	0.18	0.44	0.1	2.80	0.14	0.43	0.2	0.55	0.18	0.50	0.24	0.54	0.29
6	0.18	0.56	0.19	0.47	0.11	3.09	0.16	0.50	0.21	0.67	0.18	0.56	0.29	0.65	0.31
7	0.1	0.30	0.11	0.36	0.06	2.16	0.09	0.22	0.1	0.30	0.1	0.20	0.11	0.45	0.18
8	0.14	0.29	0.16	0.37	0.08	2.25	0.12	0.42	0.15	0.47	0.12	0.33	0.19	0.47	0.23
9	0.14	0.29	0.15	0.33	0.08	2.12	0.12	0.33	0.16	0.50	0.13	0.38	0.23	0.43	0.25
Média	0.19	0.37	0.23	0.35	0.12	2.33	0.18	0.39	0.28	0.46	0.2	0.40	0.37	0.46	0.36
10	0.82	0.37	1.22	0.35	0.67	4.33	0.92	0.47	1.73	0.36	1.11	0.49	4.86	0.60	2.34
11	0.84	0.39	1.65	1.18	0.75	5.85	1.04	0.69	1.73	0.58	1.3	0.72	6.31	0.97	2.71
12	1	0.37	1.68	0.31	0.83	3.42	1.13	0.57	2.19	0.37	1.43	0.45	7.39	0.66	3.09
13	0.27	0.67	0.44	0.86	0.24	4.75	0.32	0.81	0.56	0.95	0.38	0.89	0.99	1.59	0.69
14	0.5	0.50	0.7	0.54	0.41	3.22	0.55	0.58	0.89	0.60	0.72	0.65	2.09	0.96	1.18
15	0.53	0.58	0.85	0.75	0.43	5.18	0.58	0.76	0.95	0.76	0.75	0.76	2.26	1.15	1.32
16	0.16	0.37	0.22	0.45	0.11	2.91	0.15	0.33	0.22	0.59	0.18	0.56	0.3	0.80	0.32
17	0.32	0.56	0.42	0.43	0.25	3.00	0.34	0.59	0.61	0.67	0.43	0.53	1.16	1.07	0.75
18	0.3	0.53	1.93	0.38	0.28	2.64	0.33	0.58	0.51	0.61	0.45	0.53	0.97	1.03	0.91
Média	0.52	0.46	1.01	0.58	0.43	4.30	0.59	0.61	1.04	0.54	0.75	0.60	2.92	0.86	1.48
19	3.83	0.46	18.85	0.91	39.08	0.96	6.46	0.53	10.65	0.44	9.25	0.53	9.25	0.53	16.42
20	3.36	0.71	20.28	1.39	36.71	1.47	7.25	0.97	10.68	0.85	9.38	0.72	59.34	0.88	23.36
21	4.07	0.55	22.03	1.22	42.98	1.35	9.24	1.22	14.05	1.26	12.35	1.03	83.99	1.05	29.72
22	2.57	1.17	7.61	1.75	26.75	2.30	4.5	1.43	6.82	1.30	6.48	1.29	25.32	0.93	13.15
23	3.28	0.52	16.1	1.13	40.7	1.19	6.28	0.75	8.71	0.62	10.36	0.59	31.56	0.40	19.32
24	3.97	0.79	29.3	1.18	57.56	1.43	7.73	0.90	11.49	0.90	10.6	0.79	34.86	0.66	25.91
25	0.7	0.67	1.23	0.48	4.45	0.73	0.91	0.60	1.59	0.66	1.15	0.59	5.33	0.98	2.48
26	0.63	0.67	1.28	0.58	3.93	0.78	0.76	0.61	1.53	0.73	1.07	0.55	5.99	0.97	2.42
27	2.15	0.78	6.76	1.99	18.71	2.04	3.18	1.16	5.23	0.99	5.22	1.21	20.16	0.78	9.98
Média	2.73	0.68	13.71	1.24	3.4	2.61	5.14	0.96	7.86	0.90	7.32	0.83	30.64	0.84	15.86
Média Geral	1.15	0.63	4.99	1.18	1.3	3.08	1.97	0.91	3.06	0.84	2.76	0.80	11.31	0.84	5.9

tempo para completa otimização de cada subproblema foi medido e o resultado comprova que a minimização de z_2 é mais rápida que minimização de z_1 . Além disso, os subproblemas mais difíceis foram aqueles cujo w estão no intervalo $[\frac{1}{2}, \frac{3}{4}]$, isto é, onde as duas funções-objetivo possuem importâncias similares.

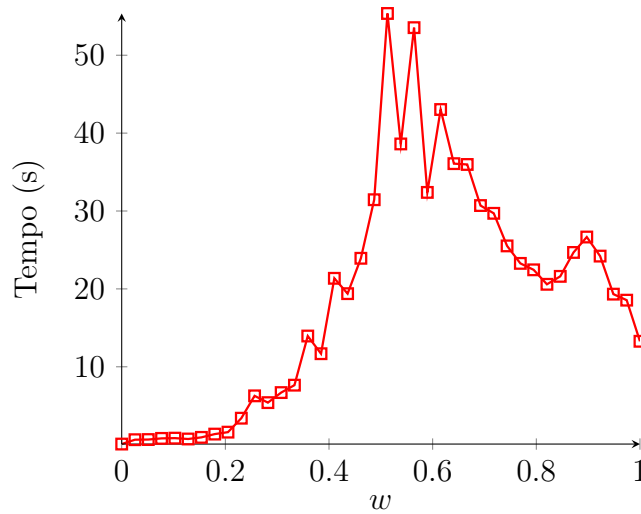


Figura 41 – Tempo computacional demandado por cada subproblema em termos de w

Em relação aos métodos Tchebycheff Sem Ponderações, ε -Restrito, NISE híbrido com ε -Restrito e Benson, conseguiram obter \mathcal{Z}^* com um tempo computacional

bem próximo e tiveram desempenhos similares no PCUIM. O Tchebycheff Sem Ponderações, em média, foi o mais rápido dentre todos, porém, com um coeficiente de variação mais elevado significando que teve sua eficiência computacional bastante variada conforme o problema. Por outro lado, o Método NISE híbrido com ε -Restrito foi o que demorou mais dentre estes.

Em com respeito a ε -Restrito e Benson, este último devido às restrições de igualdade na formulação de seus subproblemas utilizou um maior tempo computacional envolvido. A Figura (42) faz esta comparação destes dois métodos, tomando um exemplar com $m = 100$ e comprovando que há um maior esforço computacional por cada subproblema de Benson. Outra análise é o reduzido tempo consumido nos subproblemas com baixo *setup* e sendo crescente conforme o *setup*.

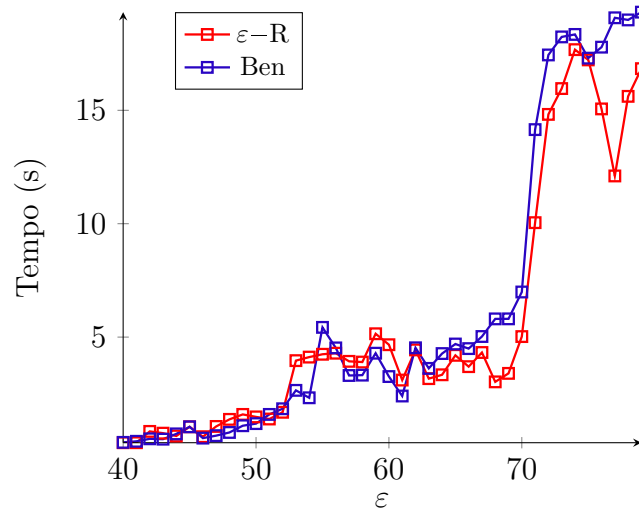


Figura 42 – Comparação dos tempos computacionais demandados por cada subproblema no ε -Restrito e Benson

Em relação ao Método de Sylva & Crema, certamente é o mais caro. Isso se deve ao fato de que seus subproblemas se tornam cada vez mais difíceis de serem resolvidos. A cada solução eficiente encontrada, duas novas variáveis binárias e restrições são acrescentadas à formulação, os subproblemas ficam mais difíceis de serem resolvidos, principalmente se a cardinalidade de $|\mathcal{Z}^*|$ for elevada. A Figura (43) faz uma experimentação do tempo envolvido em cada subproblema para um caso particular com $m = 100$.

A Tabela (25) ilustra os tempos computacionais médios \bar{t} para obter a fronteira $\bar{\mathcal{Z}}$, que inclui o tempo para gerar a fronteira parcialmente relaxada ($\tilde{\mathcal{Z}}$) somado com o tempo para fazer o arredondamento de cada solução eficiente pela HADS. Uma média geral, utilizando todos os sete algoritmos, mostra que esta abordagem utiliza apenas $100 \times \frac{2.55}{5.90} \approx 43\%$ do tempo computacional da abordagem exata.

Observamos que a relaxação das condições de integralidade das variáveis x_{jk} contribui para uma melhora no tempo de resolução dos Problema de Programação Linear

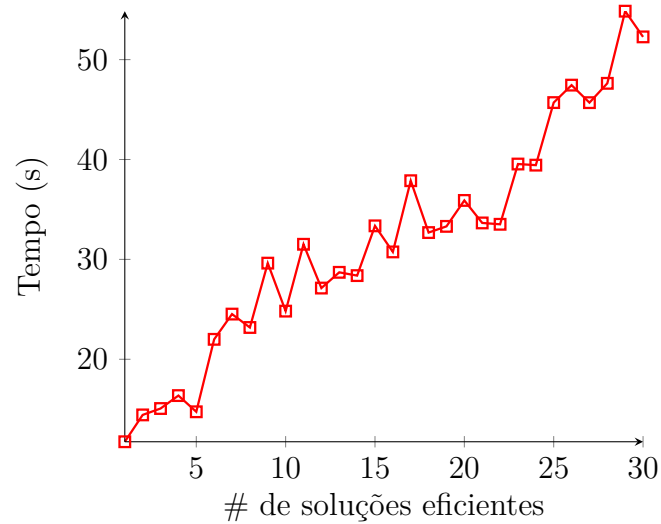


Figura 43 – Tempo computacional demandado por cada subproblema de Sylva & Crema em função do número de soluções eficientes

Tabela 25 – Tempos computacionais médios (em seg.) para obtenção de \bar{Z} - Fase 2

Classe	SP		Tch		Tch-SP		$\epsilon-R$		NISE+ $\epsilon-R$		Ben		S-C		Média
	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	\bar{t}	cv	
1	0.2	0.35	0.24	0.29	0.08	0.50	0.18	0.17	0.3	0.17	0.19	0.26	0.35	0.23	0.24
2	0.23	0.26	0.25	0.24	0.09	0.78	0.21	0.38	0.34	0.38	0.23	0.39	0.45	0.24	0.27
3	0.26	0.54	0.26	0.19	0.11	0.64	0.24	0.42	0.34	0.26	0.28	0.39	0.51	0.31	0.3
4	0.12	0.42	0.12	0.25	0.05	0.60	0.1	0.20	0.15	0.60	0.13	0.77	0.17	0.59	0.13
5	0.15	0.20	0.18	0.33	0.06	0.83	0.12	0.25	0.21	0.62	0.15	0.53	0.23	0.35	0.17
6	0.14	0.29	0.18	0.44	0.06	1.16	0.13	0.31	0.18	0.44	0.16	0.44	0.25	0.44	0.17
7	0.11	0.36	0.12	0.25	0.05	0.40	0.1	0.20	0.12	0.33	0.1	0.20	0.12	0.42	0.11
8	0.13	0.23	0.13	0.23	0.05	0.40	0.11	0.18	0.14	0.29	0.11	0.18	0.18	0.28	0.13
9	0.12	0.17	0.14	0.21	0.05	0.60	0.11	0.18	0.14	0.36	0.13	0.31	0.2	0.35	0.14
Média	0.16	0.31	0.18	0.28	0.06	0.67	0.14	0.29	0.21	0.38	0.16	0.44	0.27	0.33	0.18
10	0.86	0.78	0.93	0.77	0.41	1.61	0.91	0.79	1.64	0.60	0.97	0.75	2.2	0.65	1.2
11	0.6	0.38	0.74	0.32	0.3	0.93	0.65	0.40	1.39	0.44	0.75	0.37	2.45	0.90	1.04
12	0.83	0.23	1	0.33	0.38	0.84	0.85	0.31	1.87	0.30	0.96	0.28	3.13	0.61	1.36
13	0.25	0.44	0.32	0.41	0.11	1.45	0.24	0.46	0.41	0.85	0.3	0.60	0.63	1.00	0.35
14	0.35	0.46	0.39	0.31	0.15	1.00	0.33	0.45	0.68	0.60	0.42	0.43	1.04	0.64	0.51
15	0.41	0.66	0.48	0.50	0.18	1.33	0.4	0.60	0.77	0.78	0.48	0.56	1.15	0.84	0.6
16	0.16	0.31	0.21	0.33	0.06	1.33	0.13	0.31	0.21	0.52	0.17	0.29	0.29	0.52	0.19
17	0.27	0.33	0.32	0.25	0.12	0.83	0.26	0.31	0.46	0.70	0.33	0.36	0.63	0.49	0.37
18	0.23	0.39	0.29	0.41	0.11	1.09	0.24	0.42	0.41	0.56	0.3	0.40	0.57	0.56	0.33
Média	0.44	0.48	0.52	0.44	0.2	1.15	0.45	0.49	0.87	0.54	0.52	0.48	1.34	0.71	0.66
19	3.22	0.24	5.41	0.24	2.51	0.84	5.59	0.26	8.92	0.22	6.41	0.25	6.41	0.25	6.1
20	3.34	0.52	5.25	0.53	2.62	1.30	5.83	0.43	8.72	0.45	6.97	0.42	30.47	0.48	9.62
21	3.4	0.26	6.22	0.48	2.9	1.05	6.43	0.46	9.1	0.41	7.19	0.43	33.14	0.35	10.39
22	2.73	1.30	3.12	1.06	1.35	4.87	3	1.19	6.15	1.35	4.09	1.11	24.94	1.05	7.0
23	3.05	0.49	3.86	0.47	1.71	1.82	3.82	0.47	7.46	0.53	5.25	0.46	36.09	0.53	9.34
24	3.49	0.72	4.27	0.65	1.99	3.72	4.43	0.71	8.47	0.89	6.16	0.77	37.53	0.73	10.29
25	0.92	0.61	1.2	0.46	0.52	1.94	1.16	0.66	1.96	0.58	1.39	0.53	4.52	0.99	1.8
26	0.72	0.49	0.96	0.44	0.4	1.52	0.9	0.54	1.54	0.64	1.17	0.53	4.47	1.26	1.55
27	2.02	0.66	2.27	0.70	1.12	3.04	2.49	0.69	4.82	0.70	3.51	0.81	17.89	0.98	5.25
Média	2.54	0.58	3.62	0.54	1.68	2.03	3.74	0.55	6.35	0.61	4.68	0.56	21.72	0.66	6.81
Média Geral	1.05	0.55	1.44	0.51	0.64	1.92	1.44	0.53	2.48	0.59	1.79	0.55	7.78	0.66	2.55

Inteiros, pois nessa relaxação há metade do número de variáveis inteiras. Embora o ganho no tempo de resolução de apenas um Problema de Programação Linear Inteiro não seja tão considerável, a economia na geração do conjunto eficiente pode ser muito significativa.

Com respeito aos métodos de escalarizações, o método que propomos comprova ser mais rápido. Por outro lado, o Método de Sylva & Crema demora cerca de 12 vezes

mais que o Tchebycheff Sem Ponderações.

Este método, por sua vez, economizou cerca de 22% do tempo computacional considerando as abordagens exatas e heurísticas. O método híbrido economizou-se 19% do tempo ao passo que ε -Restrito e Benson economizaram cerca de 27% e 37%, respectivamente.

As Figuras (44), (45) e (46) ilustram o tempo computacional interferido pela dimensão e as técnicas de escalarizações adotadas³. A primeira figura visa comparar o tempo para obter \mathcal{Z}^* em função técnicas de escalarização e m ; na segunda figura fazemos a mesma comparação, mas para $\tilde{\mathcal{Z}}$; finalmente a terceira compara o tempo entre as abordagens exata e heurística.

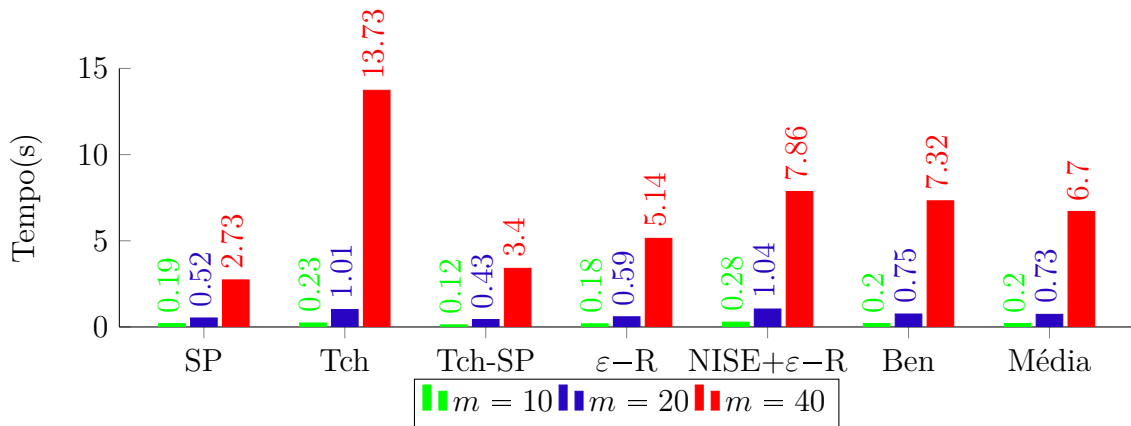


Figura 44 – Comparação dos tempos computacionais (em seg.) utilizados para obter \mathcal{Z}^*

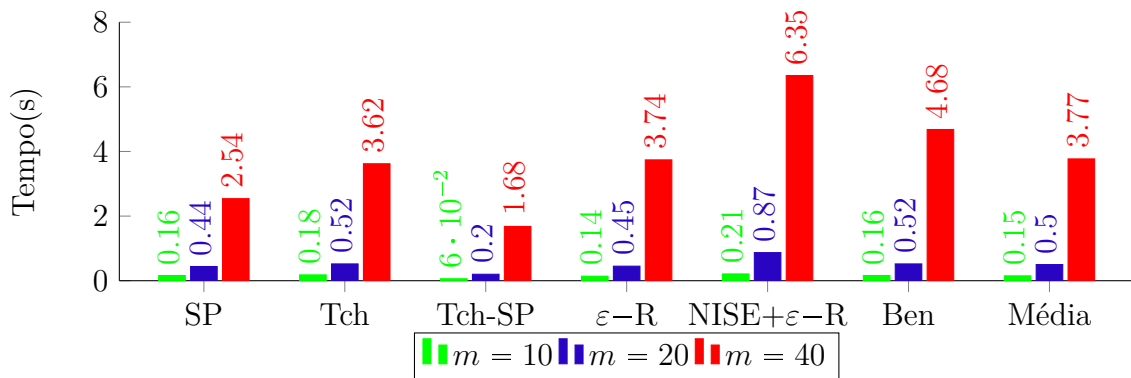


Figura 45 – Comparação dos tempos computacionais (em seg.) utilizados para obter $\tilde{\mathcal{Z}}$

À título de ilustração, apresentamos na Tabela (41) no Anexo 1 os resultados do método GC.

³ Foram omitidos os resultados do Método Sylva & Crema nas duas primeiras figuras em virtude da discrepância dos valores. A média que aparece nessas figuras não considera este método.

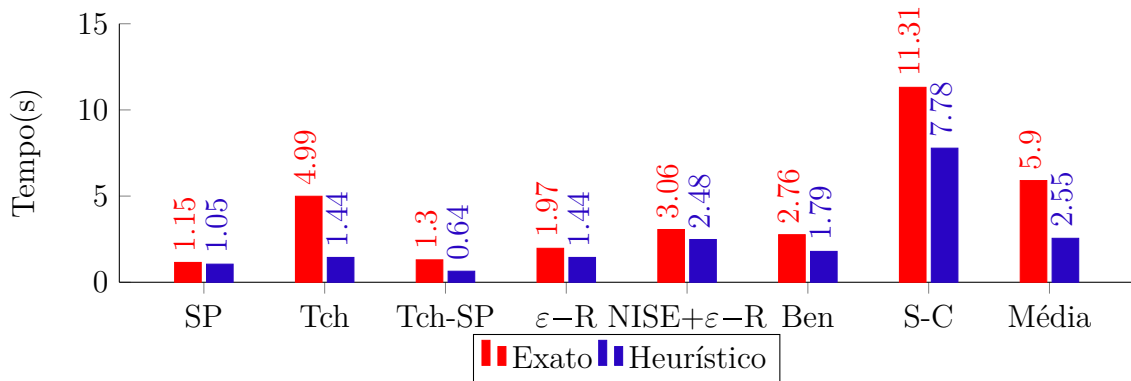


Figura 46 – Comparação dos tempos computacionais (em seg.) utilizados para obter \bar{Z} e Z^*

6.2.1 Testes com colunas adicionais

A grande parte dos exemplares para PCUIMs práticos assemelham-se com as classes presentes experimentadas na fase 2. A formulação com os padrões de corte dados por Gilmore e Gomory pode atender e dar uma resposta razoável ao gestor. No entanto, existe uma preocupação com a qualidade, quantidade e dispersão destas soluções eficientes, uma vez que geramos a fronteira de Pareto tendo em vista apenas $p = m$ colunas.

Um fator que pode influenciar demasiadamente a fronteira de Pareto de nosso problema é a quantidade de padrões de corte fornecidos ao modelo. Se o gestor possui um tempo maior na sua tomada de decisão, torna-se interessante analisar a ideia de fornecermos mais padrões de corte e verificar como a fronteira de Pareto se modifica em função do tempo de processamento. Contudo, cada subproblema a ser resolvido usará duas variáveis e restrições adicionais. Isso torna a resolução dos subproblemas mais custosa, tornando o procedimento multiobjetivo mais caro computacionalmente.

Outro fator motivacional a realizar alguns testes adicionais com mais padrões de corte é o fato de que o método de GC não gerar colunas que levam em conta o *setup*. O método utilizado apenas “prioriza” minimizar z_1 , sem considerar colunas promissoras para a segunda função-objetivo. A fim de contornar este inconveniente e gerar “boas” soluções eficientes com um *setup* baixo, foram feitos alguns testes com $p = 2 \cdot m$ e $p = 3 \cdot m$ colunas, onde as m primeiras são geradas por Gilmore e Gomory. As remanescentes foram obtidas pelo Procedimento (12) apresentado no Anexo 1 e discutido com detalhes no trabalho proposto em (ALIANO; MORETTI, 2015)

Uma maneira de gerar colunas que priorizem o *setup* consiste em alocar o maior número de itens distintos em um padrão. Além disso, dado um conjunto de m colunas iniciais, fazemos uma contagem item a item do número de vezes que ele aparece nestes padrões iniciais. Organizamos os itens por ordem crescente de prioridade e construímos um padrão de corte baseado nessa classificação, tentando alocar primeiramente os itens

que foram menos vezes utilizado previamente. Ao terminar a construção do padrão, esta contagem é novamente feita e novos padrões são construídos. Com essa ideia, este método heurístico é capaz de colocar à disposição do modelo um número maior de escolhas que darão uma maior variedade de soluções considerando os dois objetivos conflitantes de nosso problema.

Nesta subseção, visamos comparar como os diferentes métodos de escalarizações sofrem interferência quando o número de colunas aumenta. Para tanto, geramos instâncias para o PCUIM com $m = 40$ itens pertencentes à classe 19 com $p = 80$ e 120 colunas. Para cada quantidade de colunas utilizado, resolvemos 20 problemas testes e medimos o tempo computacional médio (t) para obter a fronteira de Pareto exata com respeito às colunas utilizadas (\mathcal{Z}^*), o tempo médio (t_s) para resolver cada subproblema e o número de soluções eficientes encontrado (σ_1^5). O resumo desses resultados estão disponíveis na Tabela (26).

Tabela 26 – Resultados computacionais comparando todos os métodos e com mais padrões de corte

p	Técnica	t	cv	t_s	cv	σ_1^5	cv
$p = 80$	SP	115.86	1.04	2.55	1.02	10.04	0.00
	Tch	126.41	1.03	2.92	1.00	13.00	0.57
	Tch-SP	87.92	0.98	1.78	0.97	19.70	0.15
	ε -R	153.73	1.02	3.67	0.98	17.00	0.03
	NISE+ ε -R	215.94	1.02	3.61	0.99	19.20	0.12
	Ben	141.14	1.02	3.30	0.98	20.50	0.23
	S-C	225.58	0.84	4.45	0.72	18.50	0.09
	Média	145.39	0.99	3.18	0.95	16.99	0.17
$p = 120$	SP	191.46	0.98	3.26	1.03	10.60	0.09
	Tch	202.71	0.85	5.65	0.88	15.00	0.44
	Tch-SP	109.90	0.70	2.33	0.69	20.60	0.13
	ε -R	239.71	0.72	5.35	0.69	16.67	0.05
	NISE+ ε -R	313.08	0.72	4.88	0.70	19.80	0.11
	Ben	226.24	0.72	4.40	0.70	20.60	0.19
	S-C	532.50	0.94	9.48	0.84	18.87	0.08
	Média	259.37	0.80	5.03	0.79	17.59	0.15

A Tabela (26) comprova uma maior rapidez do Método Tchebycheff Sem Ponderações e uma acentuada demora do Método Sylva & Crema, sustentando a ideia de que a técnica que propomos possui uma formulação para os subproblemas fazendo com que o CPLEX tenha mais agilidade em resolver. Para os testes realizados, para $p = 120$ a Soma Ponderada consumiu 13.84 seg. por subproblema ao passo que o Tchebycheff Sem Ponderações levou 9.34.

Outra característica observada foi o alto tempo de processamento à medida que o número de padrões de corte aumentou, passando de 600 a 1000 segundos quando p passa de 80 a 120 padrões de corte. No entanto, o uso de mais padrões de corte não traduz no maior número de soluções eficientes, e sim, interfere somente na qualidade das mesmas.

Para $p = 80$ combinações, foram encontradas em média 17.0 soluções eficientes, ao passo que para $p = 120$ obtivemos apenas 0.59 soluções adicionais.

A Figura (47) fazemos uma comparação de nove fronteiras de Pareto obtidas pelos métodos (a) Soma Ponderada, (b) Métrica de Tchebycheff e (c) Tchebycheff Sem Ponderações, usando $p = 40, 80$ e 120 padrões para um mesmo exemplar de um PCUIM. Naturalmente, conforme o número de combinações de corte aumenta, fronteiras de Pareto mais inferiores vão sendo determinadas, de sorte que a posterior (com mais colunas) vai dominando a anterior (com menos colunas).

Outra característica relevante é que, embora a Soma Ponderada obtenha pouco mais da metade das soluções eficientes, os pontos não dominados suportados obtidos distribuem-se de forma regular ao longo da fronteira de Pareto, com maior concentração na parte central da mesma. Isso é devido a característica das fronteiras de Pareto no PCUIM: íngreme nas proximidades do primeiro ponto lexicográfico (valor mínimo z_1) e achatada nas proximidades do outro ponto lexicográfico. O Método Métrica de Tchebycheff clássico consegue determinar um número maior de soluções em relação à Soma Ponderada, que são os pontos não suportados, como ilustra a mesma Figura (47). A capacidade do Método Tchebycheff Sem Ponderações, ε -Restrito, Benson e Sylva & Crema está na propriedade de captar todas as soluções eficientes e aquelas cuja imagem estão situadas em regiões mais próximas aos pontos lexicográficos, inclusive as não suportadas.

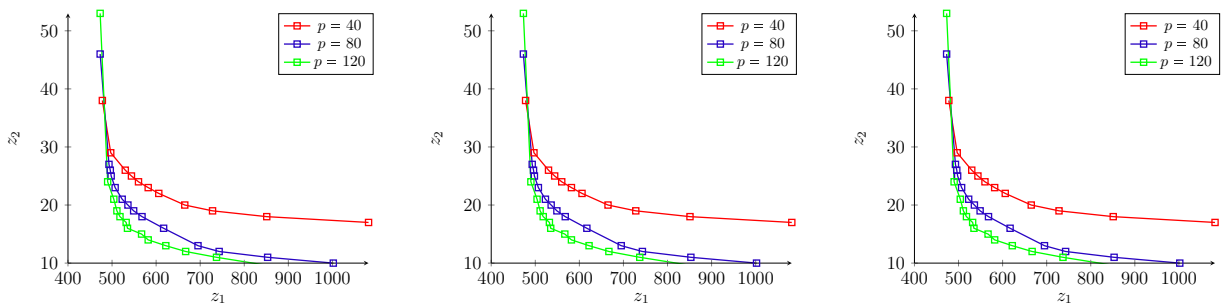


Figura 47 – Fronteiras de Pareto obtidas pela Soma Ponderada (esq.), Métrica de Tchebycheff (centro) e Tchebycheff Sem Ponderações (dir.)

Outro pequeno teste adicional que realizamos foi comparar as abordagens exata e heurística ao gerar 80, 120, 160 e 200 colunas (sendo as 40 iniciais dada pelo método de GC). Para cada número de colunas considerados, 20 exemplares foram resolvidos e calculadas as Métricas σ^5 e σ^6 para as abordagens exata e heurística. A intenção deste teste é comparar os métodos ε -Restrito e o Tchebycheff Sem Ponderações quanto ao número de soluções encontradas, tempo de processamento e número de subproblemas gerados. A Tabela (27) resume estes resultados⁴.

⁴ Para uma maior simplicidade dos resultados omitimos os coeficientes de variação.

Tabela 27 – Comparação das técnicas ε -Restrito e Tchebycheff Sem Ponderações nas abordagens exata e heurística com $p = 80, 120, 160$ e 200 padrões de corte

p	Métrica	Exata		Heurística	
		ε -R	Tch-SP	ε -R	Tch-SP
$p = 80$	σ^5	20.1	19.9	18.4	19.7
	σ^6	21.8	36.8	21.1	39.2
	t_s	4.94	0.78	3.14	0.56
$p = 120$	σ^5	19.8	19.5	17.7	17.9
	σ^6	22.5	21.1	36.4	39.4
	t_s	9.40	1.60	6.21	1.06
$p = 160$	σ^5	19.7	19.4	17.9	19.1
	σ^6	23.7	37.8	21.1	39.2
	t_s	12.23	2.00	7.74	1.46
$p = 200$	σ^5	20.4	20.5	19.5	20.2
	σ^6	22.5	39.8	21.4	40.8
	t_s	21.6	4.40	12.5	2.04

Como detectamos anteriormente, os resultados mostram numa primeira análise que as fronteiras heurísticas tem um número de pontos ligeiramente menor que a original. Embora o ε -Restrito use menos subproblemas (praticamente a metade em relação ao Tchebycheff Sem Ponderações), eles são mais difíceis de serem resolvidos que o subproblemas do Tchebycheff Sem Ponderações, como mostra os valores de t_s . É claro que a metade dos subproblemas de Tchebycheff Sem Ponderações são infactíveis e a otimização é efetivamente realizada em metade dos subproblemas que ele determina. Por exemplo, para $p = 200$ colunas, cada subproblema restrito puramente inteiro é otimizado em 21.6 segundos, ao passo que no Tchebycheff Sem Ponderações cada um é resolvido em 4.4 segundos em média. Essa compensação no custo computacional de cada subproblema deixa o Método Tchebycheff Sem Ponderações para exemplares com mais colunas ligeiramente mais rápido.

Outra análise resultante é a dificuldade crescente à medida que o número de colunas é incrementado, não porque o número de soluções eficientes aumenta, mas porque cada Problema de Programação Linear Inteiro envolver mais variáveis e restrições. Essa variação no desempenho pode ser vista nas Figuras (48) para o ε -Restrito e (49) para o Tchebycheff Sem Ponderações, que mostram o tempo computacional médio t com as abordagens exata e heurística. Essa análise quantifica como o número de colunas encarece computacionalmente os algoritmos envolvidos.

Esta fase das simulações mostrou exemplares práticos do PCUIM e o emprego

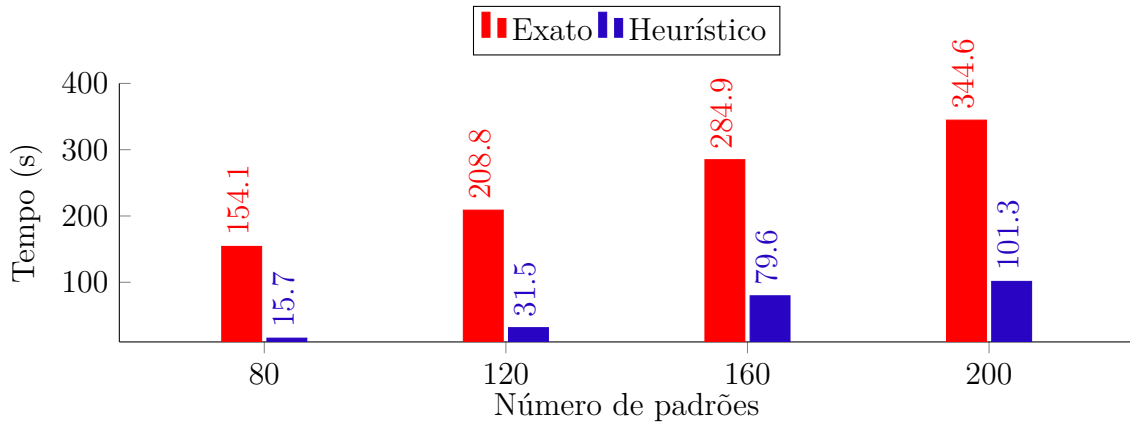


Figura 48 – Comparação dos tempos computacionais levados pelo ε -Restrito para $p = 80$, 120, 160 e 200 colunas

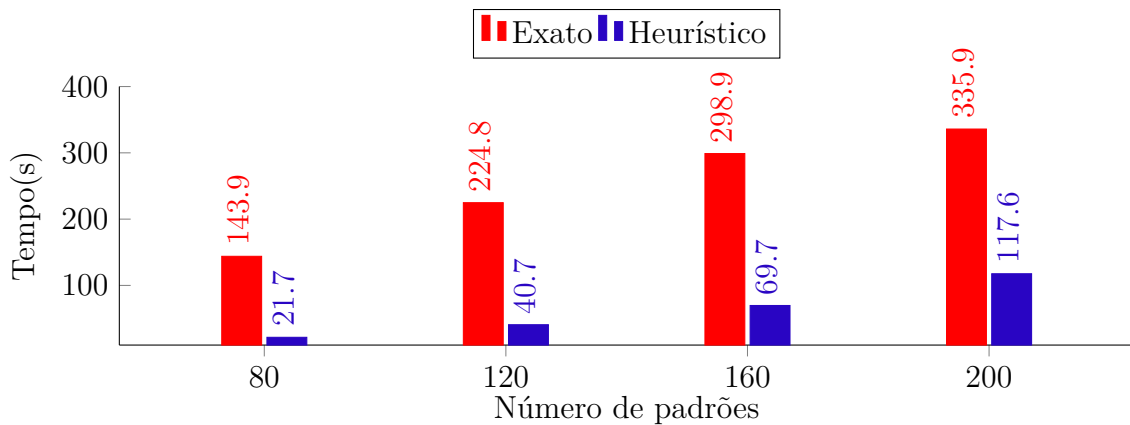


Figura 49 – Comparação dos tempos computacionais levados pelo Tchebycheff Sem Ponderações para $p = 80$, 120, 160 e 200 colunas

de sete métodos de escalarizações para solucioná-lo. As técnicas mostraram promissoras nesta aplicação, com vantagem no método proposto, o Tchebycheff Sem Ponderações, capaz de determinar todas as soluções para o PCUIM num tempo bastante competitivo com os consagrados na literatura. Mostramos também uma estratégia de aceleração para obter o conjunto eficiente resultando em uma economia no tempo de processamento. Finalmente, a última subseção apresentou alguns testes para $p = 4 \cdot m$ e $5 \cdot m$ colunas, com a intenção de obter um conjunto eficiente de melhor qualidade.

A próxima seção ilustrará simulações com exemplares de maiores dimensões.

6.3 Resultados computacionais com os padrões de corte a priori - Fase 3

Na terceira fase de testes computacionais testamos os algoritmos desenvolvidos/adaptados para o PCUIM utilizando instâncias de maior dimensão. Como visto

anteriormente, pela definição das três fronteiras, temos que $\tilde{Z} \leq Z^* \leq \bar{Z}$. Baseado nos experimentos realizados na fase 2, notamos uma proximidade entre as fronteiras Z^* e \bar{Z} . Além disso, conjecturamos que as fronteiras \bar{Z} e Z^* são próximas, então nós esperamos que as fronteiras \bar{Z} e \tilde{Z} também estejam relativamente próximas. Os resultados apresentados desta seção atestam e validam esta afirmação, mediante aos extensos resultados computacionais obtidos.

Os sete métodos foram novamente aplicados, e o intuito foi calcular a distância entre \bar{Z} e \tilde{Z} , já que Z^* não foi calculada em virtude do alto custo computacional envolvido nestas instâncias. A ideia consiste em termos um limitante superior para o erro que se comete ao obter a fronteira heurística \bar{Z} .

Para evitar um número muito grande de classes de problemas, as instâncias nessa fase foram geradas da seguinte forma:

- O número de itens m demandados foi igual a 50, 75 e 100.
- Os itens foram considerados em pequenos (P) e grandes (G) apenas.
- O número de bobinas-mestre b foi igual a 1 ou 5.

Os demais parâmetros como a demanda e o número de facas foi gerado como na Fase 2. A Tabela (16) ilustra estas classes.

Tabela 28 – Classes para o PCUIM - Fase 3

Classe	m	Tipo do item	b
1	50	P	1
2	50	P	5
3	50	G	1
4	50	G	5
5	75	P	1
6	75	P	5
7	75	G	1
8	75	G	5
9	100	P	1
10	100	P	5
11	100	G	1
12	100	G	5

A comparação entre as fronteiras heurística e parcialmente relaxada foram feitas pelas Métricas σ^1 , σ^3 , σ^4 e σ^5 . A Tabela (29) ilustra os resultados médios para σ^1 . Por exemplo, para $m = 50$ itens, a média geral da distância entre uma solução não-dominada parcialmente relaxada e a mais próxima desta, contida na fronteira heurística é de 3.73. Para $m = 75$ e $m = 100$, essas distâncias são de 5.14 e 6.51, respectivamente. Embora

pareça uma distância razoável, se tomarmos esses valores em termos da amplitude de variação dos objetivos, representam 0.6%, 0.6% e 0.1% respectivamente.

Tabela 29 – Resultados computacionais para σ^1 - Fase 3

Classe	SP		Tch		Tch-SP		$\varepsilon-R$		NISE+ $\varepsilon-R$		Ben		S-C		Média
	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	$\bar{\sigma}^1$	cv	
1	7.74	0.12	7.14	0.12	5.71	0.13	5.71	0.13	5.71	0.13	5.71	0.13	5.7	0.13	6.2
2	7.89	0.12	7.06	0.14	5.74	0.15	5.74	0.15	5.75	0.15	5.74	0.15	5.75	0.15	6.24
3	1.07	0.77	1.75	0.48	1.18	0.67	1.18	0.68	1.18	0.68	1.18	0.67	1.18	0.67	1.25
4	1.25	0.76	1.29	0.75	1.23	0.69	1.22	0.70	1.22	0.70	1.23	0.69	1.23	0.69	1.24
Média	4.49	0.20	4.31	0.21	3.47	0.23	3.46	0.23	3.46	0.23	3.47	0.23	3.46	0.24	3.73
5	10.21	0.13	8.66	0.12	6.84	0.15	6.85	0.15	6.85	0.15	6.85	0.15	6.8	0.01	7.58
6	11.36	0.15	9.51	0.12	7.63	0.14	7.63	0.14	7.63	0.14	7.63	0.14	7.6	0.13	8.43
7	1.38	0.57	1.39	0.55	1.34	0.47	1.33	0.47	1.34	0.47	1.34	0.47	1.34	0.47	1.35
8	4.74	0.45	4.76	0.48	3.8	0.41	3.81	0.40	3.81	0.40	3.81	0.40	3.8	0.41	4.07
Média	6.92	0.21	6.08	0.22	4.9	0.21	4.9	0.21	4.91	0.21	4.9	0.22	3.35	0.25	5.14
9	13.77	0.13	10.95	0.10	8.75	0.11	8.75	0.12	8.75	0.12	8.74	0.11	8.79	0.11	9.79
10	14.62	0.14	11.12	0.09	9.27	0.12	9.27	0.12	9.27	0.11	9.26	0.12	9.16	0.10	10.28
11	1.6	0.67	1.66	0.65	1.64	0.54	1.56	0.60	1.64	0.54	1.64	0.54	1.65	0.54	1.63
12	5.23	0.54	5.23	0.50	4	0.40	3.98	0.40	3.95	0.44	4	0.41	4.02	0.42	4.34
Média	8.8	0.22	7.24	0.20	5.92	0.19	5.89	0.20	5.9	0.20	5.91	0.19	5.9	0.19	6.51
Média Geral	6.74	0.21	5.88	0.21	4.76	0.21	4.75	0.21	4.76	0.21	4.76	0.21	4.75	0.20	5.13

Em termos dos métodos de escalarização, notamos que as medidas foram bastantes parecidas, sendo praticamente independente do tipo de técnica empregado. Outra consideração interessante consiste na notável diferença entre esses distanciamentos em instâncias que diferem apenas no tamanho dos itens. Por exemplo, para $m = 50$, considerando apenas as classes 1 e 2, $\bar{\sigma}^1$ é quase cinco vezes maior que nas classes 3 e 4, que possuem itens grandes. Essa tendência aproximadamente se manteve para as demais instâncias com 75 e 100 itens, o que podemos conjecturar que os PCUIMs com itens grandes possuem uma relaxação mais forte do que as que têm itens pequenos.

As amplitudes dos objetivos foram calculadas e disponibilizados na Tabela (30), sendo os resultados expressos apenas para a fronteira heurística $\bar{\mathcal{Z}}$. As mesmas conclusões que foram feitas na fase 2 podem ser feitas neste momento. Em média, as classes com $m = 50$ itens possuem quase 20 soluções eficientes, sendo essa média aumentada em quase duas unidades quando os itens são pequenos e diminuída em quase três unidades para as classes com itens grandes. A mesma conclusão é feita para as classes com $m = 75$ e $m = 100$ onde as classes com itens pequenos têm cerca de 43% de soluções eficientes a mais que as classes com itens grandes. A mesma situação ocorre no objetivo z_1 , onde nas classes com itens pequenos há praticamente o dobro da variabilidade em relação àquelas com itens grandes. Isso já era uma conclusão prévia e esperada, afinal, classes com itens pequenos tendem a ter os padrões de corte com mais itens diferentes alocados. Assim, as colunas da matriz A^* serão compostas por vetores com poucas entradas nulas, dando maior possibilidade de combinar esses padrões para atender toda a demanda. Essa maior possibilidade de combinação permite obter soluções de eficientes com baixo *setup* e alta frequência de padrões a serem cortados. Por outro lado, instâncias com itens grandes tendem a ter padrões de cortes parecidos com os padrões homogêneos, isto é, com muitas entradas nulas, de sorte que haverá poucas combinações de padrões que possam cobrir

as restrições de demanda, tendo como consequência imediata a baixa variabilidade no *setup*. Esse aspecto influencia o desempenho dos métodos multiobjetivos. Naturalmente, as classes com itens pequenos são mais difíceis de serem resolvidas relativamente as que contém itens grandes, uma vez que é necessário um maior número de subproblemas para a completa enumeração da fronteira. Outro aspecto que merece destaque é a maior amplitude de z_1 em relação à z_2 , justificando a implementação dos métodos ε -Restrito e Benson, que minimizam z_1 e impõem z_2 como restrição.

Tabela 30 – Resultados computacionais para σ_{12}^3 e σ_{22}^3 - Fase 3

Classe	SP, Tch, Tch-SP, ε -R, NISE+ ε -R, Ben				S-C	
	σ_{12}^3	σ_{22}^3	σ_{12}^3	σ_{22}^3	Média σ_{*2}^3	Média σ_{*2}^3
1	661.2	21.30	661.1	21.15	661.1	21.23
2	822.7	22.20	822.3	22.31	822.5	22.25
3	460.4	16.40	460.2	16.20	460.3	16.30
4	360.1	18.62	360.1	18.38	360.1	18.50
Média	576.1	19.63	575.93	19.54	576.01	19.57
5	1001.3	32.72	1000.7	32.64	1001.3	32.68
6	1104.6	30.80	1104.6	30.70	1104.6	30.75
7	590.4	23.69	590.3	23.57	590.3	23.63
8	703.2	23.80	702.8	23.90	703.4	23.85
Média	849.88	27.75	849.6	27.71	849.74	27.73
9	1504.1	42.98	1504.1	42.78	1504.13	42.88
10	1605.4	43.16	1605.3	43.24	1605.35	43.20
11	790.21	28.30	790.1	28.10	790.15	28.20
12	660.36	32.38	660.31	32.42	660.35	32.40
Média	4403.8	36.65	4403.48	36.67	4403.64	36.67
Média Geral	854.91	28.02	854.93	27.96	854.92	27.99

A Tabela (31) ilustra a média das razões entre as áreas delimitadas pelas fronteiras \tilde{Z} e \tilde{Z}^* . Como $\tilde{Z} \leq \tilde{Z}^*$, então essa razão é sempre menor que um. Quanto mais próxima de um, mais próximas estão as soluções não dominadas entre essas fronteiras.

Tabela 31 – Resultados computacionais para $\frac{\sigma_2^4}{\sigma_1^4}$ - Fase 3

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	$\frac{\sigma_2^4}{\sigma_1^4}$	cv	
1	0.977	0.01	0.972	0.01	0.945	0.03	0.945	0.03	0.946	0.03	0.946	0.03	0.949	0.03	0.954
2	0.981	0.01	0.975	0.01	0.964	0.02	0.964	0.02	0.964	0.02	0.964	0.02	0.967	0.02	0.969
3	0.993	0.01	0.992	0.01	0.987	0.01	0.994	0.00	0.99	0.01	0.99	0.01	0.994	0.00	0.991
4	0.994	0.01	0.992	0.01	0.986	0.01	0.993	0.01	0.989	0.01	0.989	0.01	0.993	0.01	0.991
Média	0.986	0.01	0.983	0.01	0.971	0.02	0.974	0.02	0.973	0.02	0.973	0.02	0.976	0.01	0.976
5	0.981	0.01	0.981	0.01	0.979	0.02	0.964	0.02	0.963	0.02	0.963	0.02	0.966	0.02	0.971
6	0.979	0.01	0.977	0.01	0.963	0.01	0.963	0.01	0.965	0.01	0.965	0.01	0.966	0.01	0.968
7	0.996	0.00	0.993	0.01	0.989	0.01	0.992	0.01	0.991	0.01	0.991	0.01	0.991	0.01	0.992
8	0.985	0.01	0.98	0.01	0.976	0.02	0.988	0.01	0.978	0.01	0.978	0.01	0.978	0.01	0.98
Média	0.985	0.01	0.983	0.01	0.977	0.01	0.977	0.01	0.974	0.01	0.974	0.01	0.975	0.01	0.978
9	0.981	0.01	0.977	0.01	0.964	0.02	0.964	0.02	0.965	0.02	0.965	0.02	0.965	0.02	0.969
10	0.981	0.01	0.978	0.01	0.966	0.01	0.966	0.01	0.966	0.01	0.966	0.01	0.969	0.01	0.971
11	0.997	0.00	0.995	0.01	0.993	0.00	0.995	0.01	0.994	0.01	0.994	0.00	0.995	0.01	0.995
12	0.988	0.01	0.978	0.01	0.967	0.03	0.976	0.01	0.976	0.01	0.976	0.01	0.976	0.01	0.977
Média	0.987	0.01	0.982	0.01	0.973	0.02	0.975	0.01	0.975	0.01	0.975	0.01	0.976	0.01	0.978
Média Geral	0.986	0.01	0.982	0.01	0.973	0.02	0.975	0.01	0.974	0.01	0.974	0.01	0.976	0.01	0.986

Em linhas gerais, os resultados comprovam uma proximidade entre essas fronteiras, de sorte que que o erro ao aproximar Z^* por \tilde{Z} ser majorado por um bom limitante. Há dois fatores associados à essa proximidade: o primeiro, a boa qualidade da relaxação

linear e segundo, a heurística HADS que mostrou ser capaz de gerar soluções inteiras de considerável precisão.

Outro aspecto que vem ajudar a confirmar nossas conjecturas já idealizadas anteriormente, são as diferenças entre as classes com itens pequenos e grandes comparadas com as respectivas relaxações. Em média, as classes com itens pequenos estão mais longe da relaxação que as com itens grandes. Para $m = 50$, a área ocupada pela fronteira heurística em média é 96.1% da fronteira relaxada, considerando as classes com itens pequenos; na mesma instância, considerando classes com itens grandes, essa razão atinge 99.1%.

Tabela 32 – Resultados computacionais para σ_1^5 - Fase 3

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	σ_1^5	cv	
1	10.75	0.17	15.45	0.15	20.85	0.13	20.85	0.13	20.9	0.13	20.85	0.13	20.85	0.13	18.64
2	10.72	0.12	16.54	0.11	21.71	0.09	21.71	0.09	21.71	0.09	21.71	0.09	21.67	0.10	19.4
3	9.03	0.19	7.82	0.22	15.47	0.21	15.47	0.21	15.47	0.21	15.47	0.21	15.47	0.21	13.46
4	9.21	0.19	8.01	0.24	15.78	0.23	15.72	0.22	15.72	0.22	15.72	0.22	15.78	0.23	13.71
Média	9.93	0.16	11.95	0.16	18.45	0.16	18.44	0.16	18.45	0.16	18.44	0.16	18.44	0.16	16.3
5	14.35	0.16	22.85	0.08	32.75	0.10	32.8	0.10	32.75	0.10	32.75	0.10	32.25	0.09	28.64
6	13.9	0.14	23.1	0.10	31.05	0.10	31	0.10	31.05	0.10	31	0.10	30.5	0.09	27.37
7	12.4	0.18	11.85	0.21	24.3	0.17	24.3	0.17	24.3	0.17	24.2	0.17	24.25	0.17	20.8
8	11.63	0.13	12.44	0.17	24.68	0.16	24.63	0.16	24.68	0.16	24.68	0.16	24.37	0.15	21.01
Média	13.07	0.15	17.56	0.13	28.19	0.13	28.18	0.13	28.19	0.13	28.16	0.13	27.84	0.12	24.46
9	15.8	0.11	29.85	0.10	42.25	0.08	42.25	0.08	42.25	0.08	42.25	0.08	38.5	0.08	36.16
10	15.54	0.10	30.56	0.11	40.55	0.09	40.55	0.09	40.55	0.09	40.5	0.09	38.75	0.09	35.29
11	14.77	0.13	15.54	0.18	33.01	0.14	32.9	0.14	32.89	0.15	32.95	0.14	32.69	0.14	27.82
12	12.85	0.12	17.33	0.20	33.95	0.13	34.27	0.14	34.93	0.14	33.9	0.13	32.98	0.12	28.6
Média	14.74	0.11	23.32	0.13	37.44	0.11	37.49	0.11	37.65	0.11	37.4	0.11	35.73	0.10	31.97
Média Geral	12.58	0.14	17.61	0.14	28.03	0.12	28.04	0.13	28.1	0.13	28	0.12	27.34	0.12	24.24

A Tabela (33) ilustra a média de soluções não dominadas na fronteira \bar{Z} . Os resultados são muito similares em relação à última coluna da Tabela (31), que ilustra a amplitude média de z_2 . Os métodos Tchebycheff Sem Ponderações, ε -Restrito, Benson, NISE híbrido com ε -Restrito e Sylva & Crema fornecem praticamente todas as soluções não dominadas do problema. Algumas pequenas diferenças entre eles aconteceu devido à limitação do tempo computacional para resolver os subproblemas e uso do filtro de Pareto posterior. Em contrapartida, a Soma Ponderada e Métrica de Tchebycheff mais uma vez falharam ao tentar esboçar todas as soluções eficientes. Quantitativamente, a Soma Ponderada obteve 42% das soluções ao passo que Métrica de Tchebycheff encontrou pouco mais que 60%.

Existe um aumento praticamente linear com m do número de soluções encontradas. Para $m = 50$ itens, obtivemos em média 16.3 soluções, para $m = 75$, $16.6 \times 1.5 = 24.46$ soluções e finalmente para $m = 100$ itens aproximadamente 32.0 soluções.

A Figura (50) apresenta estes resultados na forma de um histograma. Observamos que o Método Sylva & Crema teve seu desempenho muito afetado com a dimensão do problema, por dois motivos: (i) os subproblemas são mais demorados de serem otimizados e (ii) o maior número de soluções não dominadas, acarretando o surgimento de variáveis binárias e restrições conforme novas soluções são encontradas. Acreditamos que essas

restrições adicionais “atrapalham” o CPLEX na sua otimização, devido ao *big-M* das restrições novas. Essas constatações mostram e justificam o porquê o Sylva & Crema demorou 142% a mais com relação a média.

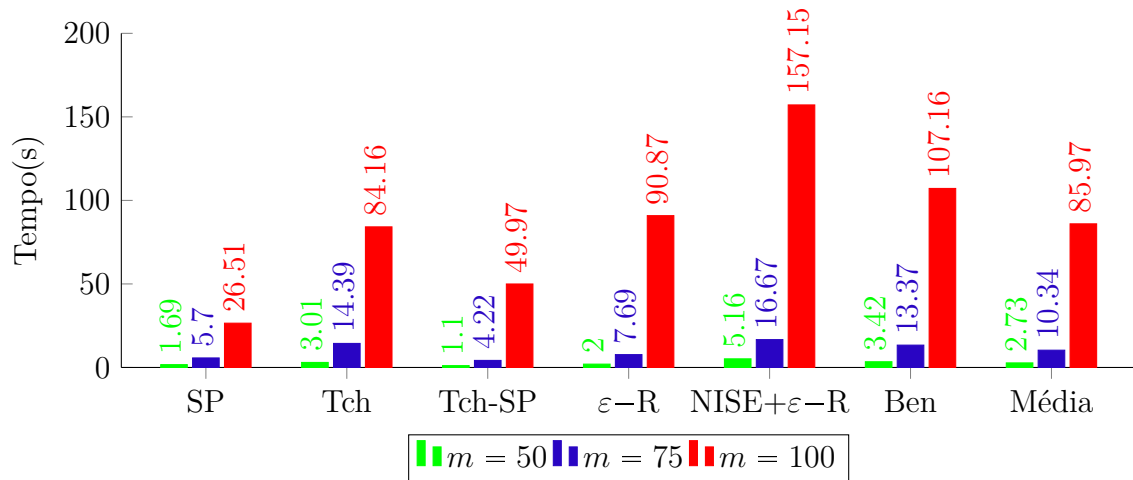


Figura 50 – Comparação dos tempos computacionais (em seg.) utilizados para obter \bar{Z} e Z^*

Em relação aos demais métodos, vemos que o NISE híbrido com ε -Restrito foi uma proposta neste trabalho que consumiu mais tempo que os demais, visto que o seu método ser uma Soma Ponderada mais elaborada (NISE), capaz de encontrar todas as soluções não dominadas e suportadas, e o ε -Restrito. Se apenas as soluções não suportadas fossem obtidas, este método seria mais rápido que a Soma Ponderada, pois ele necessita de apenas $2 \cdot |\mathcal{Z}_s| - 1$ subproblemas para as calcular, onde \mathcal{Z}_s denota a cardinalidade do conjunto das soluções suportadas. Isso é plausível quando observamos a proporção de soluções suportadas do PCUIM nas classes que geramos neste trabalho: em torno de 75%. O fator que deixa este método híbrido mais lento é a posterior aplicação do Método ε -Restrito, tentando resgatar as de soluções que NISE não gerou. Esse duplo processo deixou o método híbrido um pouco mais lento. Uma primeira proposta para esta hibridação mais competitiva consiste em melhorar a implementação do NISE ou combinar o NISE com o Tchebycheff Sem Ponderações.

Embora o ε -Restrito e o Método Benson tenham uma filosofia de pesquisa parecida, Benson foi o terceiro mais lento dentre os sete. As duas restrições de igualdade que relacionam os desvios l_i , deixam os subproblemas mais complicados de serem otimizados. Presumimos que a eficiência do Benson está associada com o ponto inicial z^0 , isto é, quanto mais próximo z^0 está do vetor ótimo do subproblema, mais forte será a sua formulação. Embora teoricamente qualquer escolha para z^0 à direita do vetor ótimo seria plausível, isso tornaria a formulação para o Problema de Programação Linear Inteiro mais longe de seu envoltório convexo. Isso nos motivou a escolher o ponto z^0 de tal forma que ele tenha na sua primeira componente, Δ unidades a menos que a primeira componente do vetor

ótimo encontrado na iteração anterior e, na sua segunda componente, Δ unidades a mais. Testes isolados ilustraram que essa estratégia, se não fosse adotada, teria deixado Benson cerca de 15% mais lento.

Como destacamos, o Método Soma Ponderada foi o mais rápido dentre os sete métodos. Da forma como implementamos, ele gera a mesma quantidade de subproblemas que os métodos Métrica de Tchebycheff, ε -Restrito, Benson e Sylva & Crema. No entanto foi o que gerou a menor quantidade de soluções eficientes. Uma medida para comparar e testar a performance deste e dos outros métodos consiste determinar quantas soluções eficientes são obtidas por segundo de execução. A Figura (51) ilustra esses resultados e fornece uma boa ideia de custo \times benefício destes métodos. A Soma Ponderada por exemplo, foi o método com o segundo melhor custo \times benefício, determinando em média 1.11 soluções eficientes por segundo, superando inclusive o ε -Restrito. O método mais eficiente foi o Tchebycheff Sem Ponderações, onde determina cerca de 1.53 soluções por segundo, confirmando os resultados das fases anteriores. Sem dúvida, essa análise mostra uma medida mais justa e ampla do desempenho dessas metodologias.

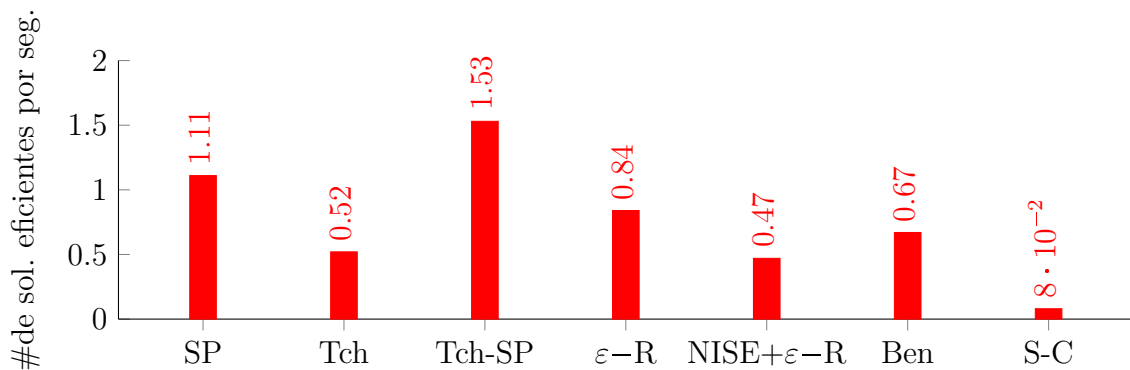


Figura 51 – Comparação dos custos-benefícios dos métodos de escalarização: $|\tilde{Z}| \times \text{Tempo}^{-1}$

A Tabela (34) ilustra os tempos computacionais utilizados pela HADS para arredondar a Fronteira de Pareto \tilde{Z} de cada método. Naturalmente, o tempo de processamento deste método depende especificamente do número de soluções não dominadas a ele fornecido e da dimensão da matriz de restrições, A^* . Para $m = 50$, em média há 16.30 soluções a serem arredondadas e o tempo consumido foi em média de 1.15 segundos. Para $m = 75$ itens, temos 24.46 soluções e o tempo demandado foi de 5.65 segundos, isto é, 0.23 segundos por solução. Finalmente, para $m = 100$ itens, o tempo médio para arredondar uma solução sobe para 0.29 segundo. Em suma, a HADS mostrou uma boa performance (que podem ser constatadas pela Métrica σ^5) e utilizou um baixo tempo computacional. Ressaltamos que outras técnicas de arredondamento, apresentadas no Capítulo 4, foram testadas e comparadas, mas a HADS apresentou resultados mais competitivos.

Tabela 33 – Tempos computacionais médios (em seg.) para obter as fronteiras \bar{Z} - Fase 3

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	
1	1.82	0.47	3.53	0.41	1.16	2.60	2.12	0.48	4.91	0.39	3.52	0.47	38.83	0.42	8.38
2	2.24	0.18	4.42	0.14	1.72	0.13	3.14	0.39	7.88	0.12	5.6	0.37	63.79	0.37	12.96
3	1.02	0.23	1.25	0.25	0.55	0.62	1.01	0.32	2.45	0.54	1.52	0.23	12.9	0.52	3.08
4	1.68	0.12	2.84	0.13	0.95	0.88	1.73	0.01	5.39	0.41	3.05	0.07	23.85	0.23	5.93
Média	1.69	0.25	3.01	0.23	1.1	1.01	2	0.32	5.16	0.31	3.42	0.31	34.84	0.37	7.59
5	9.63	0.87	26.77	0.83	7.78	4.89	14.16	0.79	31.62	0.64	23.85	0.70	331.24	0.82	68.31
6	7.55	0.80	23.29	0.90	6.03	4.49	10.97	0.69	21.45	0.67	17.72	0.61	278.68	0.88	55.22
7	2.28	0.46	3.71	0.53	0.67	7.73	2.52	0.49	5.86	0.56	5.85	0.76	49.57	0.62	10.8
8	3.34	0.13	3.8	0.08	0.05	25.35	3.09	0.03	7.76	0.12	6.08	0.03	56.95	0.01	12.33
Média	5.7	0.69	14.39	0.79	4.22	4.24	7.69	0.65	16.67	0.58	13.37	0.60	179.11	0.76	36.66
9	17.76	0.48	79.6	0.48	17.17	3.74	31.23	0.49	68.05	0.56	50.91	0.50	1009.8	0.29	193.69
10	71.89	0.46	224.68	0.40	170.19	0.91	309.44	0.49	516.03	0.18	326.91	0.46	1074.45	0.40	410.24
11	3.8	0.42	9.18	0.53	3.32	3.80	6.05	0.53	10.81	0.56	17.6	0.50	266.7	0.84	47.5
12	12.59	0.17	23.19	0.04	9.22	0.92	16.77	0.15	33.7	0.14	33.21	0.10	528.94	0.42	101.32
Média	26.51	0.42	84.16	0.40	49.97	1.21	90.87	0.47	157.15	0.23	107.16	0.44	719.97	0.40	188.19
Média Geral	11.3	0.46	33.85	0.45	18.43	1.43	33.52	0.48	59.66	0.26	41.32	0.45	311.31	0.47	77.48

Ilustramos nas Figuras (52) e (53) as fronteiras de Pareto \tilde{Z} (vermelha), Z^* (alaranjada) e \bar{Z} (amarela) de um PCUIM pertencente as Classes 9 e 11, respectivamente.

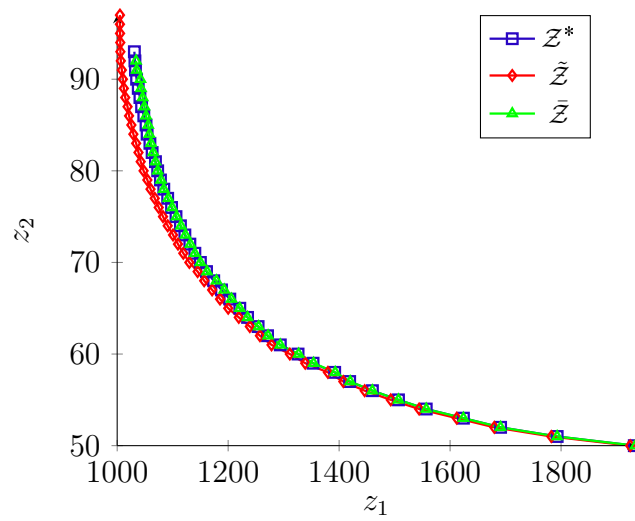


Figura 52 – Fronteira de Pareto Z^* , \tilde{Z} e \bar{Z} para um exemplar da classe 9

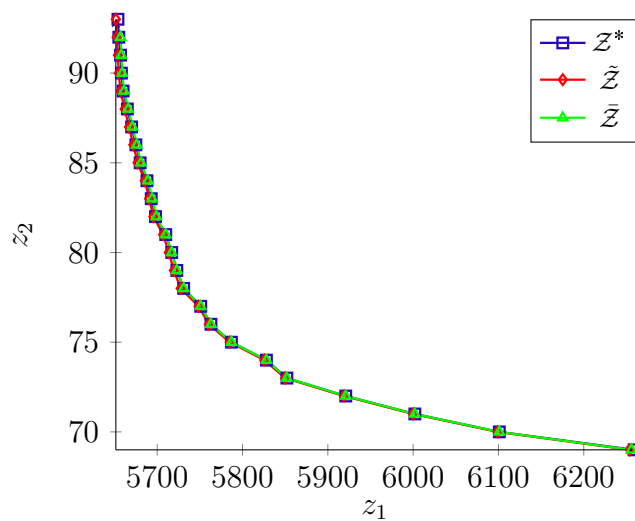


Figura 53 – Fronteira de Pareto Z^* , \tilde{Z} e \bar{Z} para um exemplar da classe 11

As tabelas e resultados ilustrados confirmaram-se nestas figuras, onde há uma imperceptível diferença entre estas fronteiras que obtivemos. Além disso, as evidências numéricas com respeito às classes com itens pequenos e grandes são percebidas nestas figuras. Os PCUIMs com itens pequenos tendem a ter uma fronteira de Pareto de maior amplitude e com maior cardinalidade; por outro lado, exemplares com classes com itens grandes possuem uma fronteira de menor amplitude, menor cardinalidade e uma maior proximidade entre as três fronteiras. Destacamos ainda, que o arredondamento das soluções eficientes que estão mais à esquerda e na parte superior, tendem a gerar soluções inteiras dominadas. Isso acontece porque estas soluções parcialmente relaxadas estão muito

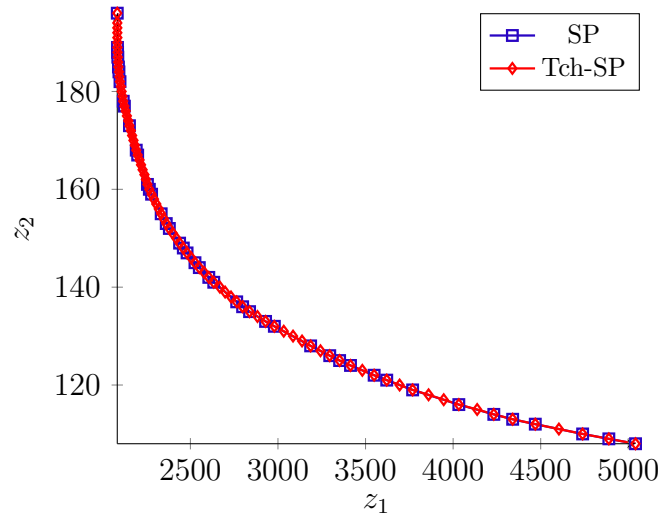


Figura 54 – Fronteiras de Pareto típicas determinadas pela Soma Ponderada e Tchebycheff Sem Ponderações para um exemplar com $m = 200$

próximas umas das outras. O mesmo fato não se verifica para os pontos localizados na parte inferior à direita.

Para finalizar esta seção, a Figura (54) traz uma comparação entre os métodos Soma Ponderada e Tchebycheff Sem Ponderações numa particular simulação com $m = 200$, mostrando como a fronteira de Pareto pode ser representada de diferentes formas à medida que um procedimento de escalarização é utilizado.

Tabela 34 – Tempos computacionais médios (em segundos) utilizados pela HADS - Fase 3

Classe	SP		Tch		Tch-SP		ε -R		NISE+ ε -R		Ben		S-C		Média
	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	<i>t</i>	<i>cv</i>	
1	2.03	0.71	1.38	0.54	2.77	0.30	2.69	0.13	3.18	0.25	2.9	0.25	3.1	0.14	2.45
2	1.51	0.80	1.23	0.63	1.54	0.64	2.12	0.31	1.66	0.35	1.49	0.30	1.57	0.63	1.49
3	0.68	1.52	0.19	2.26	0.32	1.09	0.18	0.22	0.34	1.23	0.17	2.06	0.35	1.29	0.25
4	0.55	0.68	0.37	1.03	0.39	1.00	0.39	1.08	0.53	1.04	0.51	0.96	0.4	1.20	0.41
Média	1.11	0.81	0.79	0.75	1.26	0.51	1.35	0.27	1.43	0.41	1.27	0.39	1.36	0.43	1.15
5	2.77	0.85	3.71	0.33	8.64	0.15	8.43	0.14	8.26	0.13	8.26	0.16	8.47	0.23	6.76
6	4.71	0.37	10.32	0.09	17.58	0.05	17.26	0.06	17.58	0.11	17.58	0.11	17.23	0.14	14.31
7	0.41	5.35	0.28	3.96	0.57	2.16	0.6	2.58	0.49	3.94	0.7	2.49	0.76	1.22	0.52
8	0.89	0.94	0.49	1.33	1.09	0.53	1.37	0.67	1.2	1.00	1.29	0.81	1.2	0.85	1.02
Média	2.22	0.81	3.7	0.26	6.97	0.14	6.92	0.17	6.88	0.22	6.96	0.22	6.92	0.23	5.65
9	7.72	0.28	10.28	0.18	17.67	0.10	17.47	0.12	16.77	0.10	16.5	0.10	17.72	0.12	14.39
10	7.09	0.51	10.67	0.16	19.06	0.08	19.11	0.12	19.09	0.10	18.87	0.07	18.54	0.14	15.61
11	0.61	0.35	0.49	0.47	0.79	0.43	0.73	0.75	1.09	0.72	0.87	0.67	1.04	1.08	0.76
12	3.52	0.23	3.9	0.17	8.36	0.07	8.12	0.08	8.22	0.10	8.26	0.23	7.9	0.17	6.67
Média	4.73	0.36	6.34	0.18	11.47	0.09	11.36	0.12	11.29	0.12	11.13	0.12	11.3	0.16	9.36
Média Geral	2.68	0.54	3.61	0.25	6.57	0.14	6.54	0.15	6.53	0.18	6.45	0.17	6.52	0.20	5.39

6.4 Resultados computacionais com os coeficientes nas funções-objetivo diferente de 1 - Fase 4

Nesta última fase dos experimentos realizados, descrevemos os resultados que foram obtidos com os padrões de corte dados a priori, como nas fases 2, 3 e 4, mas considerando que as funções-objetivos do PCUIM não tenham custo unitários c_{jk}^i iguais à 1, $i = 1, 2$. Nas simulações realizadas, $c_{jk}^1 \in [0, 10]$ e $c_{jk}^2 \in [0, 100]$. Dito de outra forma,

$$z_1 = \sum_{k=1}^b \sum_{j=1}^p c_{jk}^1 \cdot x_{jk} \quad (6.1)$$

e

$$z_2 = \sum_{k=1}^b \sum_{j=1}^p c_{jk}^2 \cdot y_{jk}. \quad (6.2)$$

O simples fato desses coeficientes tomarem valores reais e diferentes de 1, fazem com que a fronteira de Pareto para o PCUIM modifique drasticamente, como na cardinalidade, amplitude e na distribuição das soluções ao longo da mesma. Com isso, as projeções de cada ponto não dominado nos eixos coordenados do espaço de critério podem não ser inteiros, nem seu distanciamento variar de uma em uma unidade. Dessa forma os algoritmos ε -Restrito, Tchebycheff Sem Ponderações e Benson necessitam ter seus parâmetros alterados para (i) ou conseguirem obter todas as soluções eficientes (ii) minimizar o número de pontos não encontrados.

O parâmetro Δ que controla o tamanho de passo nos métodos ε -Restrito e Benson e o deslocamento dos pontos de referência no Tchebycheff Sem Ponderações, foi definido para estas três abordagens da seguinte forma:

$$\Delta = \frac{z_2^+ - z_2^-}{\theta}, \quad (6.3)$$

onde $\theta > 2$ é um parâmetro a ser definido pelo usuário e denota um limitante superior do número de soluções eficientes que é desejado. Noutras palavras, o tamanho de passo a ser definido pela equação (6.3) é interpretado como se a projeção da fronteira de Pareto no eixo z_2 fosse dividida em θ sub-intervalos de mesmo comprimento, esperando uma solução eficiente cuja imagem esteja em cada um desses sub-intervalos. Conforme θ é aumentado, o tamanho de passo é diminuído, e assim, a qualidade da aproximação melhora, esperando que mais soluções sejam encontradas.

Conforme pôde ser visualizado nas Figuras (52) e (53), por vezes um POM pode ter muitas soluções não dominadas, tornando algumas regiões da fronteira “congestionadas” de pontos devido a proximidade entre elas. Isso pode vir atrapalhar o decisor numa tomada de decisão, no momento de optar por uma alternativa entre as disponíveis. Nessa linha de raciocínio, por vezes é interessante ter uma representação da fronteira de Pareto de tal

modo que as soluções fiquem suficientemente espaçadas, facilitando a escolha de uma ou outra alternativa. O papel do parâmetro $\Delta(\theta)$ é exatamente fornecer esse espaçamento desses pontos não dominados e ao mesmo tempo, agilizar o tempo de processamento do método para eleger estas representações.

Fixado o parâmetro $\Delta(\theta)$ para as três técnicas, serão comprovados mediante os experimentos computacionais que o Método de Tchebycheff Sem Ponderações melhor se adaptou, em relação às duas outras técnicas em análise, no sentido do custo \times benefício.

Os três métodos, ε -R, Tchebycheff Sem Ponderações e Benson foram aplicados em 20 subproblemas nos quais foram divididos em três categorias distintas. Em todas classes e subproblemas, os PCUIMs possuem apenas itens pequenos e uma única bobina-mestre. Essas categorias são as seguintes:

- **Categoria 1:** m foi fixado em 40 itens. Além disso, utilizamos $p = 2 \cdot m$ colunas, onde m foram obtidas pelo método GC e as m remanescentes foram geradas pelo Algoritmo (12). O parâmetro θ tomou os valores 50, 75 e 100.
- **Categoria 2:** θ foi fixado em 100 e utilizamos $p = m$ colunas obtidas pelo método GC. O número de itens variou entre 20, 40 e 60.
- **Categoria 3:** θ foi fixado em 100 e utilizamos $m = 40$ itens. O número de colunas p utilizadas variou entre m , $2 \cdot m$ e $3 \cdot m$, sendo as m primeiras geradas pelo método de GC e as remanescentes pelo Algoritmo (12).

Os outros dados para o PCUIM, como demanda e número de facas foram gerados como nas subseções anteriores. Cada um dos 20 subproblemas foram resolvidos pelas três técnicas de escalarizações e duas fronteiras Z^* e \bar{Z} foram obtidas e comparadas entre si. Essa comparação foi realizada mediante o cálculo da médias das Métricas σ^1 , σ^2 , σ^4 e σ^5 . Além disso, calculamos as quantidades σ_1^6 e σ_2^6 , que denotam o número de subproblemas que cada técnica de escalarização usou para obter as fronteiras de Pareto Z^* e \bar{Z} , respectivamente.

Primeiramente, na Tabela (35), exibimos os resultados para a Categoria 1 dos experimentos. A primeira métrica mensura as distâncias médias absolutas entre cada ponto não dominado das duas fronteiras, heurística e exata. Conforme θ aumentou, esse distanciamento foi diminuindo, pois uma melhor precisão para estas fronteiras vão sendo determinadas. Embora essa distância foi de 52.4, em média, esse valor em termos relativos corresponde menos de 0.1% da amplitude dos objetivos. A Métrica σ^2 apresenta a proporção de soluções coincidentes entre essas fronteiras e os resultados se confirmam como nas seções anteriores. Há cerca de 95.4% de soluções exatamente coincidentes, em média. Além

dessas constatações, a razão das áreas entre essas fronteiras foi bastante próxima de um, como podemos observar pela razão $\frac{\sigma_2^4}{\sigma_1^4}$.

Tabela 35 – Resultados computacionais médios para a Categoria 1, com $\theta = 50, 75$ e 100 - Fase 4

	Técnica	σ^1		σ^2		$\frac{\sigma_2^4}{\sigma_1^4}$		σ_1^5		σ_2^5		σ_1^6		σ_2^6	
		média	cv	média	cv	média	cv	média	cv	média	cv	média	cv	média	cv
$\theta = 50$	ε -R	74.79	0.32	0.09	0.09	0.98	0.01	17.30	0.12	16.70	0.14	50.00	0.00	50.00	0.00
	Tch-SP	63.87	0.42	0.06	0.07	0.99	0.01	16.70	0.16	15.50	0.15	30.41	0.17	28.45	0.17
	Ben	75.86	0.29	0.08	0.09	0.99	0.01	17.20	0.15	16.40	0.11	50.00	0.00	50.00	0.00
$\theta = 75$	ε -R	55.55	0.31	0.02	0.03	0.99	0.01	27.61	0.11	25.63	0.11	75.00	0.00	75.00	0.00
	Tch-SP	42.91	0.38	0.03	0.04	0.99	0.00	26.00	0.13	24.40	0.12	49.00	0.14	45.80	0.13
	Ben	56.88	0.30	0.02	0.03	0.99	0.00	26.20	0.11	25.40	0.10	75.00	0.00	75.00	0.00
$\theta = 100$	ε -R	32.67	0.32	0.06	0.04	0.99	0.02	30.08	0.20	29.50	0.18	100.00	0.00	100.00	0.00
	Tch-SP	37.08	0.65	0.07	0.05	0.99	0.01	31.13	0.19	28.00	0.19	56.03	0.20	53.02	0.21
	Ben	33.26	0.32	0.06	0.05	0.99	0.01	30.10	0.20	29.40	0.16	100.00	0.00	100.00	0.00

O número de soluções obtidas por cada técnica é o dado mais relevante nestas análises. Como notamos pelas Métricas σ_1^5 e σ_2^5 , os resultados mostraram um equilíbrio entre os três métodos de escalarizações comparados. Outra análise que pode ser feita, é a melhora na qualidade da fronteira de Pareto quanto ao número de soluções, regulada pelo parâmetro θ ; contudo, observamos uma estabilização do número de soluções obtidas em função do aumento desse parâmetro. Basta observar que para $\theta = 50$, temos 17 soluções; para $\theta = 75$, 25 soluções, isto é, 54% a mais. Para $\theta = 100$, obtivemos pouco mais de 30 soluções, ou seja, apenas 20% a mais em relação à $\theta = 75$. Isso significa que para valores de θ superiores à 100, os métodos começarão ficar mais custosos computacionalmente e o número de soluções resultantes tenderá a se estabilizar. É claro que essa saturação depende fortemente da instância do problema (m) e do número de padrões levados em conta (p). Concluimos que, do ponto de vista custo \times benefício, $\theta = 75$ foi a melhor escolha nesses exemplares.

A Figura (55) ilustra três fronteiras sendo modificadas conforme θ varia.

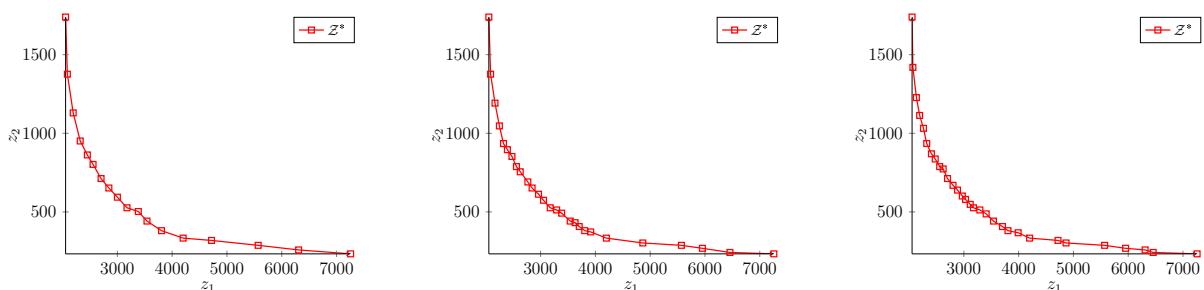


Figura 55 – Diferentes aproximações da Fronteira de Pareto para $\theta = 50$ (18 soluções) à esq., $\theta = 75$ (27 soluções) ao centro e $\theta = 100$ (31 soluções) à dir.

Outro fato observado nesses resultados, é o número de subproblemas gerados a cada escolha de θ . Da forma como os métodos ε -Restrito e Benson foram definidos, o

número de subproblemas foi igual à θ . No entanto, o Método Tchebycheff Sem Ponderações necessitou, em média, 60% de subproblemas a menos que essas duas técnicas, de modo que a quantidade de soluções eficientes obtidas foi praticamente a mesma. Esta foi a grande vantagem do método que propomos em relação aos outros, de sorte que quantidade de subproblemas gerados não depender da topologia da fronteira de Pareto e nem da sua amplitude.

A Tabela (36) mostra os resultados para a segunda categoria de experimentos, com a intenção de avaliar como a dimensão do problema afetou o número de soluções obtidas e a quantidade de problemas gerados.

Notamos uma diminuição dos valores σ^1 em relação aos resultados na Categoria 1. Por exemplo, se $m = 40$, a média dos distanciamentos ficou abaixo de 9.0. O principal fator para essa redução está associado ao número de padrões fornecido, isto é, a HADS tende a produzir mais soluções diferentes da ótima quanto mais colunas a matriz A^* possuir.

Tabela 36 – Resultados computacionais médios para a Categoria 2, com $m = 20, 40$ e 60 - Fase 4

	Técnica	σ^1		σ^2		$\frac{\sigma_2^4}{\sigma_1^4}$		σ_1^5		σ_2^5		σ_1^6		σ_2^6	
		média	cv	média	cv	média	cv	média	cv	média	cv	média	cv	média	cv
20 m	ε -R	6.46	1.29	0.11	0.10	0.97	0.01	15.00	0.17	14.80	0.16	100.00	0.00	100.00	0.00
	Tch-SP	4.48	0.67	0.11	0.09	0.99	0.01	13.90	0.17	13.80	0.16	24.80	0.20	24.60	0.18
	Ben	6.46	1.29	0.12	0.09	0.98	0.01	15.00	0.17	14.80	0.16	100.00	0.00	100.00	0.00
40 m	ε -R	9.02	0.43	0.04	0.05	0.99	0.01	30.30	0.20	30.10	0.17	100.00	0.00	100.00	0.00
	Tch-SP	8.41	0.24	0.04	0.06	0.99	0.01	27.50	0.22	27.10	0.22	52.03	0.23	51.20	0.24
	Ben	9.06	0.43	0.04	0.05	0.99	0.01	30.30	0.20	30.10	0.17	100.00	0.00	100.00	0.00
60 m	ε -R	16.24	0.28	0.03	0.03	0.99	0.01	30.50	0.15	30.40	0.13	100.00	0.00	100.00	0.00
	Tch-SP	13.01	0.36	0.03	0.03	0.99	0.01	29.30	0.15	29.10	0.12	55.60	0.16	55.20	0.13
	Ben	16.36	0.29	0.03	0.03	0.99	0.01	30.33	0.16	30.40	0.13	100.00	0.00	100.00	0.00

Em relação ao número de pontos não dominados em função da instância, notamos que há uma tendência de um crescimento até $m = 40$; para $m = 60$ itens, o número de soluções não dominadas encontrado foi praticamente o mesmo. Concluimos que o parâmetro θ escolhido é grande o suficiente para a instância com $m = 60$ itens, de modo que os métodos encontram a mesma quantidade de soluções que nas instâncias com $m = 40$. Isso ocorre devido ao fato dos os pontos não dominados estarem mais próximos uns dos outros. Nesse caso, θ necessita ser reduzido para a geração de um conjunto eficiente mais representativo.

Assim como na Categoria 1, o comportamento dos métodos em relação ao número de subproblemas gerados foi parecido. O Método ε -Restrito e Benson necessitam uma média aproximada de 4.47 subproblemas para gerar uma solução eficiente, ao passo que o Método Tchebycheff Sem Ponderações necessitou de apenas 1.9 subproblemas. Isso claramente é um aspecto que ajudou este procedimento ser mais rápido que os demais procedimentos.

A Figura (56) mostra como a fronteira é modificada em função de m .

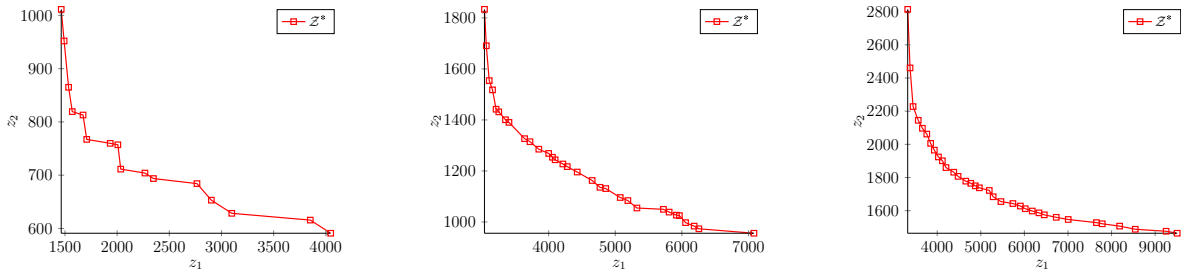


Figura 56 – Diferentes aproximações da Fronteira de Pareto conforme m é modificado: $m = 20$ (15 soluções) à esq., $m = 40$ (31 soluções) ao centro e $m = 60$ (33 soluções) à dir.

Por último, apresentamos na Tabela (37) os resultados computacionais para a Categoria 3, onde θ e m foram fixos e p variou. Em problemas práticos, muitas vezes o cliente oferece um conjunto de padrões de corte para serem cortados e essa quantidade pode variar, não restrito à apenas m . Os métodos de escalarizações podem ter suas eficiências modificadas à medida que essa quantidade de padrões se alterar. Nesta categoria, geramos as m colunas por GC, obtivemos a fronteira de Pareto correspondente à estes padrões; posteriormente m e $2 \cdot m$ padrões adicionais foram gerados de maneira aleatória e as fronteiras correspondentes foram sendo calculadas com as três metodologias de escalarizações. Naturalmente, à medida que p aumenta, mais inferior ficam as fronteiras, assim como o erro e a proporção de pontos não coincidentes entre \bar{Z} e Z^* .

Assim como na Categoria 2, o número de soluções obtidas para $p = 60$ foi mais reduzido que para $p = 40$. Isso mostra que a escolha para $\theta = 100$ é insuficiente para descobrir novas soluções eventualmente existentes, em virtude dos pontos não dominados estarem mais próximos entre si. A Figura (57) traz uma ilustração de três fronteiras obtidas conforme p aumenta.

Tabela 37 – Resultados computacionais médios para a Categoria 3. com $p = m$, $2 \cdot m$ e $3 \cdot m$ - Fase 4

	Técnica	σ^1		σ^2		$\frac{\sigma_2^4}{\sigma_1^4}$		σ_1^5		σ_2^5		σ_1^6		σ_2^6	
		média	cv	média	cv	média	cv	média	cv	média	cv	média	cv	média	cv
$p = m$	ε -R	10.76	0.62	0.03	0.03	0.99	0.01	26.60	0.14	26.10	0.13	100.00	0.00	100.00	0.00
	Tch-SP	5.66	0.35	0.02	0.03	0.99	0.01	28.60	0.20	28.50	0.17	54.20	0.16	54.12	0.19
	Ben	10.76	0.62	0.03	0.03	0.98	0.02	26.60	0.14	26.10	0.13	100.00	0.00	100.00	0.00
$p = 2m$	ε -R	20.81	0.48	0.12	0.09	0.98	0.01	31.20	0.13	31.20	0.13	100.00	0.00	100.00	0.00
	Tch-SP	21.85	0.35	0.07	0.03	0.98	0.01	33.40	0.11	32.80	0.14	62.60	0.12	64.40	0.13
	Ben	21.23	0.45	0.12	0.10	0.98	0.03	33.20	0.11	30.50	0.10	100.00	0.00	100.00	0.00
$p = 3m$	ε -R	31.83	0.54	0.11	0.07	0.98	0.01	26.50	0.13	26.50	0.13	100.00	0.00	100.00	0.00
	Tch-SP	45.01	0.68	0.06	0.03	0.91	0.02	29.20	0.14	26.53	0.13	55.80	0.13	48.78	0.15
	Ben	33.14	0.53	0.10	0.05	0.99	0.02	28.02	0.11	25.60	0.12	100.00	0.00	100.00	0.00

Em todas elas, notamos uma maior concentração de soluções na parte intermediária da fronteira e mais esparsa nas caudas, pois durante a execução do método, o

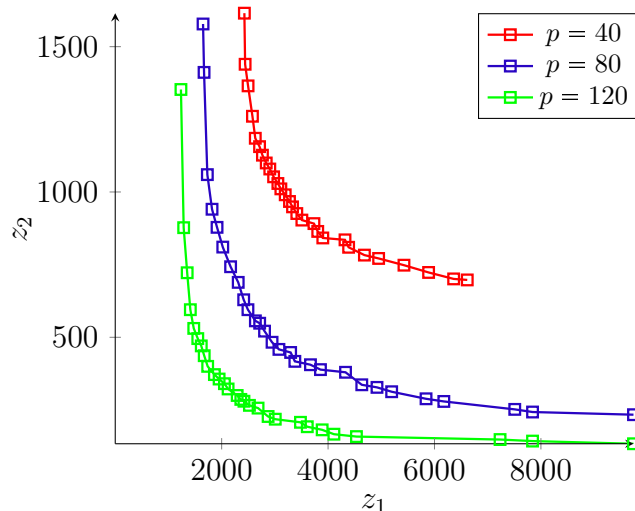


Figura 57 – Diferentes fronteiras de Pareto conforme p varia

valor de Δ sempre foi fixo. No entanto, o tamanho de passo poderia variar ao longo da execução, priorizando determinadas regiões de mais interesse.

A seguir ilustramos os tempos computacionais para resolver cada PCUIM, em média. Os números indicados por t^* e \bar{t} denotam os tempos (em seg.) para a obtenção das fronteiras \mathcal{Z}^* e $\bar{\mathcal{Z}}$, respectivamente. Além disso, medimos t_s^* e \bar{t}_s , que são os tempos médios necessários para resolver cada subproblema inteiro e parcialmente relaxado, respectivamente. Os resultados, separados por cada categoria e técnica, estão dispostos nas tabelas que seguem⁵. Fazendo uma comparação entre os tempos levados pelas abordagens exata e heurística, em média, temos que a abordagem heurística consumiu pouco mais 40% do tempo computacional. Nessa categoria de experimentos, apenas 5% das soluções eficientes foram perdidas com o arredondamento.

Tabela 38 – Tempos computacionais médios (em segundos) para a Categoria 1. com $\theta = 50$. 75 e 100 - Fase 4

	Técnica	t^*		\bar{t}		t_s^*		\bar{t}_s	
		média	cv	média	cv	média	cv	média	cv
$\theta = 50$	ε -R	76.57	0.49	28.26	0.40	1.48	0.50	0.57	0.41
	Tch-SP	42.41	0.49	18.64	0.48	1.32	0.45	0.58	0.38
	Ben	97.15	0.38	34.81	0.38	1.89	0.39	0.70	0.38
$\theta = 75$	ε -R	117.23	0.46	58.31	0.30	1.53	0.47	0.78	0.30
	Tch-SP	77.25	0.40	35.12	0.39	1.59	0.39	0.70	0.33
	Ben	172.11	0.44	75.02	0.32	2.26	0.44	1.01	0.32
$\theta = 100$	ε -R	188.51	0.57	73.35	0.34	1.85	0.57	0.74	0.34
	Tch-SP	83.30	0.52	39.11	0.43	1.39	0.43	0.66	0.32
	Ben	257.99	0.45	92.73	0.30	2.55	0.45	0.93	0.31

A Figura (58) faz uma ilustração do custo \times benefício de cada procedimento

⁵ Os resultados para \bar{t} englobam o tempo computacional para resolver os subproblemas parcialmente relaxados somado com o tempo para realizar os arredondamentos das soluções.

nas abordagens exata (a) e heurística (b).

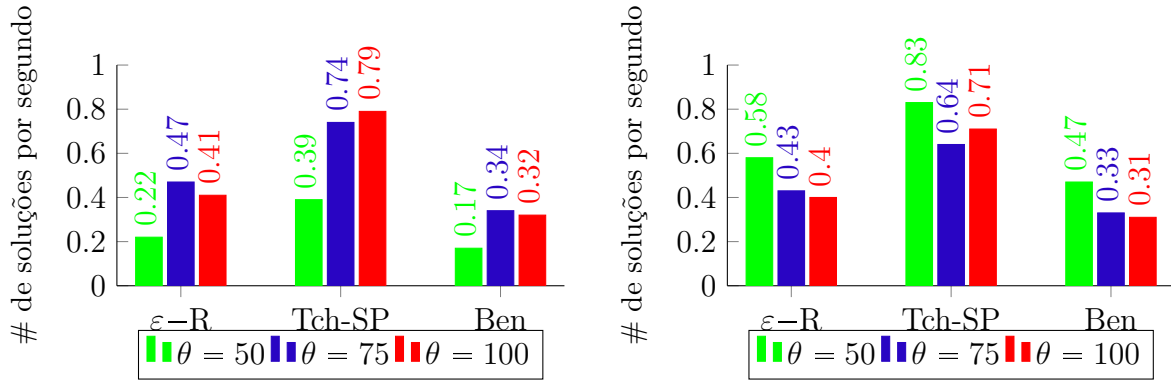


Figura 58 – Relação do número de soluções obtidas por segundo - Categoria 1

Sem dúvida, Tchebycheff Sem Ponderações foi o que conseguiu o maior número de soluções por segundo, seguido do ε -Restrito e Benson. Outro fator importante é que na abordagem exata, a maior eficiência se concentrou para os maiores valores para θ , ao passo que na abordagem heurística a maior eficiência por foi dada para $\theta = 50$.

Na Tabela (39), detalhamos os tempos computacionais demandados médios na segunda categoria destes experimentos. A tendência se repete, com Tchebycheff Sem Ponderações mais rápido que os demais métodos e o Benson o mais lento. Para $m = 60$ itens, tanto na abordagem exata quanto heurística, Tchebycheff Sem Ponderações conseguiu consumir quase a metade do tempo do ε -Restrito. As duas últimas colunas atestam que essa diferença entre essas duas técnicas não está tão ligada ao fato dos subproblemas serem mais demorados de serem otimizados, mas sim, pelo fato do Tchebycheff Sem Ponderações necessitar de quase metade do número de subproblemas para obter quase a mesma quantidade de soluções. Com relação ao ε -Restrito e Benson, ambos geraram a mesma quantidade de subproblemas, mas os subproblemas de Benson são ligeiramente mais demorados de serem resolvidos em relação ao ε -Restrito.

Tabela 39 – Tempos computacionais médios (em seg.) para a Categoria 2. com $m = 20, 40$ e 60 - Fase 4

	Técnica	t^*		\bar{t}		t_s^*		\bar{t}_s	
		média	cv	média	cv	média	cv	média	cv
$m = 20$	ε -R	7.29	0.46	6.48	0.50	0.72	0.47	0.62	0.52
	Tch-SP	1.43	0.34	1.29	0.42	0.53	0.44	0.41	0.51
	Ben	8.26	0.45	7.91	0.53	0.84	0.46	0.84	0.54
$m = 40$	ε -R	17.37	0.33	12.51	0.27	1.72	0.33	1.23	0.28
	Tch-SP	8.52	0.50	6.05	0.34	1.56	0.33	1.13	0.20
	Ben	21.64	0.39	15.34	0.28	2.11	0.40	1.54	0.29
$m = 60$	ε -R	24.94	0.28	15.30	0.17	2.44	0.29	1.52	0.18
	Tch-SP	12.95	0.33	7.24	0.18	2.35	0.37	1.26	0.18
	Ben	33.20	0.38	18.50	0.16	3.34	0.38	1.86	0.16

A Figura (59) fornece uma análise comparativa entre a performance destes métodos. Notamos que ela pouco varia à medida que m se modifica, tanto na abordagem exata quanto heurística. Outro fato importante é que todos os métodos tiveram performances melhorada quando a estratégia heurística foi adotada. No ε -Restrito, a esse valor é varia de 0.2 para 0,48 soluções por segundo, significando um aumento de 140%. Já no Tchebycheff Sem Ponderações, essa melhora foi de 118% e no Benson igual à 267%.

A Tabela (6.4) apresenta os resultados para os tempos computacionais demandados para as experimentações variando-se o número de padrões de corte. Observamos um acentuado aumento no custo computacional à medida que p cresce. Isso é uma característica dos subproblemas considerados, visto que sua formulação exigir duas variáveis (uma inteira e outra binária) para cada padrão de corte. As mesmas conclusões retiradas anteriormente podem servir para esta classe de experimentos, como por exemplo, o ganho em termos computacionais ao se utilizar a formulação parcialmente relaxada. Por exemplo, para $m = 60$ padrões de corte, houve uma economia de 54, 51 e 59% do tempo computacional, respectivamente ao ε -Restrito, Tchebycheff Sem Ponderações e Benson.

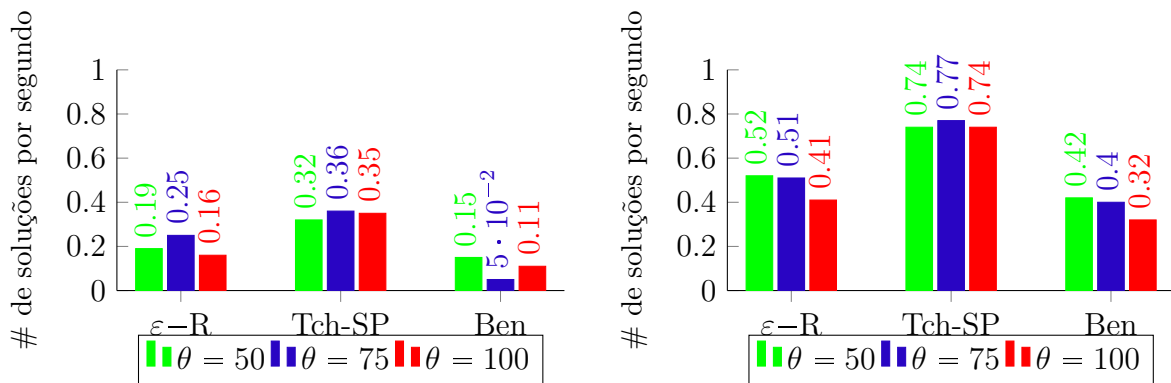


Figura 59 – Relação do número de soluções obtidas por segundo - Categoria 2

Tabela 40 – Tempos computacionais médios (em seg.) para a Categoria 3, com $p = m$, $2 \cdot m$ e $3 \cdot m$ - Fase 4

	Técnica	t^*		\bar{t}		t_s^*		\bar{t}_s	
		média	cv	média	cv	média	cv	média	cv
$p = m$	ε -R	19.72	0.54	11.56	0.27	0.19	0.55	0.11	0.28
	Tch-SP	10.79	0.57	5.37	0.22	0.19	0.49	0.09	0.16
	Ben	26.50	0.57	14.04	0.26	0.26	0.57	0.14	0.27
$p = 2 \cdot m$	ε -R	255.94	0.40	102.29	0.27	2.55	0.40	1.03	0.27
	Tch-SP	154.07	0.26	70.57	0.32	2.38	0.17	1.07	0.27
	Ben	337.75	0.36	133.10	0.23	3.37	0.36	1.34	0.23
$p = 3 \cdot m$	ε -R	368.84	0.30	169.82	0.22	3.68	0.30	1.71	0.22
	Tch-SP	163.53	0.32	79.84	0.19	2.99	0.14	1.41	0.09
	Ben	498.13	0.22	203.70	0.16	4.98	0.22	2.05	0.16

Na Figura (60), apresentamos as performances desses métodos por unidade de tempo. Existe uma brusca queda quando p aumenta, devido ao maior custo computacional

em otimizar cada subproblema e ao fato do número pontos não dominados não sofrer acentuada interferência com respeito a p .

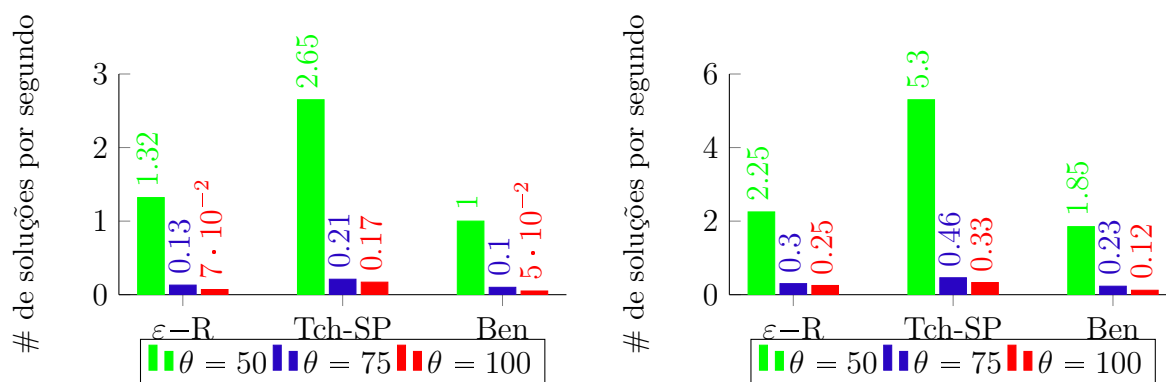


Figura 60 – Relação do número de soluções obtidas por segundo - Categoria 3

Finalizando esta seção, para o caso onde os coeficientes da função-objetivo são diferentes de um, os métodos tiveram suas performances modificadas. A experiência ao longo de todos estes experimentos demonstrou que há um excesso de soluções concentradas na parte central da fronteira e uma escassez de pontos nas “caudas” da fronteira. Isso torna uma tarefa bastante difícil a obtenção de todas as soluções não dominadas, principalmente pela elevada quantidade e o desconhecimento da distribuição exata destes pontos em relação aos demais. Na forma como foi construído os algoritmos, no ε -Restrito e Benson, a escolha para o parâmetro θ é uma tarefa mais difícil se comparado ao Tchebycheff Sem Ponderações, principalmente porque os pontos não dominados estarem mal distribuídos ao longo da fronteira. Além disso, a escolha de θ denota a escolha para o número de subproblemas que irão ser gerados. Valores pequenos podem gerar uma fronteira muito mal representada; por outro lado, valores grandes podem encarecer demasiadamente o procedimento. Essa má distribuição destes pontos podem gerar muitos subproblemas que fornecem pontos já obtidos de em iterações precedentes. No Método Tchebycheff Sem Ponderações, essa escolha não está atrelada ao número de subproblemas que irão ser otimizados. Além disso, garante substancialmente a precisão desejada da fronteira. Desta maneira, tal como foi definido, esse método de escalarização sempre gera soluções eficientes distintas para subproblemas distintos.

Uma simulação particular para a Categoria 1, quantifica essas diferenças mais acentuadamente. Com um $\theta = 400$, obtivemos 79 soluções eficientes com o ε -Restrito em 766 segundos, ao passo que no Tchebycheff Sem Ponderações as mesmas 79 soluções foram obtidas em 217 segundos, resolvendo-se apenas 153 subproblemas. Para quaisquer valores acima de 400, nenhuma outra solução além das 79 foi obtida.

Essa, com certeza, é uma maior competência do Método Tchebycheff Sem Ponderações que propomos e sem dúvida, é um dos motivos que o pontuam como mais promissor no PCUIM.

Parte III

Considerações finais

Capítulo 7

Conclusões e Perspectivas

Neste trabalho, tratamos o Problema de Corte Unidimensional Inteiro Multiobjetivo, ao minimizar dois objetivos conflitantes, a saber, a soma das frequências dos padrões de corte para atender a demanda mínima requisitada e o *setup*. Embora o Problema de Corte Unidimensional seja vastamente explorado na literatura, a atual versão bi-objetiva foi pouco retratada, principalmente, empregando técnicas clássicas de otimização multiobjetivo. Sua resolução, exploração e aprofundamento é de extrema importância para área de Pesquisa Operacional.

Nesta forma de abordarmos o problema, um conjunto de soluções ditas *eficientes* foi fornecido ao decisor e elas estabelecem um compromisso entre a quantidade de material a ser cortada e o número de possíveis trocas de padrões de corte. Os trabalhos correlatos até a presente data, consideram este segundo objetivo ponderando-o na função-objetivo. Contudo, esta maneira de resolver o problema tem suas limitações, pois a solução gerada pode ficar muito dependente dos pesos concedidos a estas funções-objetivo e, conseqüentemente, podemos ter uma solução que não corresponde às expectativas do decisor.

Primeiramente, trouxemos uma revisão de vários modelos matemáticos de Problemas de Corte Unidimensionais, explorando os aspectos mono e multiobjetivos, tendo várias bobinas mestres distintas, com ou sem *setup*, fornecendo-se ou não os padrões de corte a priori.

Esta última característica, leva à uma formulação não linear, que por sua vez, ainda é pouco estudada na literatura específica da área. Nesta tese, baseado nos trabalhos (HARJUNKOSKI et al., 1996) e (HARJUNKOSKI et al., 1998) linearizou-se este modelo e aplicou-se os métodos de escalarizações aqui desenvolvidos/adaptados. A vantagem desta abordagem é que ele garante a otimalidade global das soluções eficientes encontradas, sem se preocupar em com os padrões de corte, que já são automaticamente gerados pelo modelo. Contudo, devido ao aumento considerável no número de variáveis e restrições no processo de linearização, ocasiona sérias dificuldades em otimizar os subproblemas, o que torna este caminho difícil de ser percorrido em exemplares de mediano tamanho.

Além destas seis técnicas adaptadas, desenvolvemos uma nova técnica de escalarização e que pode ser aplicada não apenas neste problema, mas em qualquer problema de otimização combinatória biobjetivo. É inspirada e fundida nas características teóricas da Métrica de Tchebycheff, no Método *Non Inferior Set Estimation* e numa ideia de mudar iterativamente o ponto de referência à medida que um novo ponto não dominado é encontrado. O resultado foi o desenvolvimento de uma nova metodologia de escalarização, fruto destas ideias e que foi capaz de gerar a fronteira de Pareto para o PCUIM, gastando em torno de 50% do tempo computacional do ε -Restrito. Além disso, do ponto de vista matemático, este método foi competitivo com os demais métodos multiobjetivos mais clássicos da literatura, especialmente para o caso onde a imagem dos pontos não dominados não são inteiros. Até o momento, existia uma dificuldade em como definir novos subproblemas de modo que novas soluções eficientes diferentes fossem obtidas. O método aqui proposto supera esta dificuldade, onde baseado nas soluções já obtidas, redefine um novo subproblema capaz de obter uma nova solução, sem gerar qualquer variável adicional ou restrições. Isso ficou muito bem justificado pela série de experimentos computacionais realizados. Além dessa desejável característica, essa técnica possui um custo computacional mais reduzido comparado com os demais métodos consagrados na literatura. Isso porque os subproblemas de Programação Linear Inteira gerados possuem uma formulação e limitantes para a função-objetivo mais forte, fazendo com que o CPLEX tenha mais agilidade na ramificação e podas no Método *Branch-and-Bound*.

Além desta contribuição, um outro aspecto importante deve ser ressaltado com esta pesquisa. Quando o problema de corte é resolvido fornecendo-se os padrões de corte (caso linear), a condição de integralidade da frequência dos padrões pôde ser relaxada e ser integralizada por uma heurística especialmente aqui idealizada, num procedimento pós-otimização. A qualidade destas soluções, quando comparadas com as soluções do modelo inteiro original (5.11), foram surpreendentes, isto é, em torno de 91% de soluções coincidiram. As 9% remanescentes ficam à uma distância euclidiana de apenas 0.42 da original, em média. Os experimentos computacionais demonstraram que, em média, apenas 2% do número total de soluções eficientes foram perdidas. O aspecto surpreendente foi o tempo computacional utilizado por este procedimento heurístico, em torno de 43% do tempo do procedimento exato para instâncias de até 40 itens. A grande notoriedade destes resultados se deve à Heurística de Arredondamento por Direções Simplex, inédita e especialmente desenvolvida neste trabalho.

Assim sendo, é possível resolver este importante problema multiobjetivo, usando ferramentas clássicas de programação matemática para fornecer um conjunto de soluções eficientes, que pode ajudar o tomador de decisões na escolha de uma esquema de corte de materiais mais aplicável à sua realidade. Ao mesmo tempo, vários meios de aceleração destas técnicas clássicas foram propostos e tais ideias foram confirmadas e validadas.

7.1 Principais contribuições deste trabalho

Pontuamos algumas contribuições deste trabalho a seguir:

- Resolver o problema do corte de maneira não linear e considerar diferentes estratégias para sua linearização;
- Desenvolvimento, implementação e comparação de sete métodos clássicos de otimização multiobjetivo;
- Desenvolvimento de uma nova técnica de escalarização, que se mostrou mais promissora e eficaz;
- Desenvolvimento de uma estratégia heurística de relaxação para acelerar a obtenção do conjunto eficiente;
- Desenvolvimento de uma Heurística de Arredondamento específica para gerar soluções eficientes inteiras.

7.2 Trabalhos futuros

Este trabalho veio colaborar com alguns tópicos que ainda eram pouco explorados na área. No entanto, alguns aspectos merecem destaque e são passíveis de serem estudados em trabalhos futuros, a curto e longo prazo.

- Exploração dos modelos linearizados, proveniente das formulações que não fornecem os padrões de corte a priori e estudar de desigualdades válidas.
- Consideração de mais objetivos, como perdas, excesso de produção e adaptar as técnicas de escalarizações desenvolvidas neste trabalho nestas novas considerações.
- Aplicar estas técnicas no Problema de Corte Bidimensional biobjetivo.
- Aprimoramento da implementação do método combinado $\text{NISE}+\varepsilon$ -Restrito e desenvolver outras abordagens híbridas, como por exemplo o NISE com o Método Tchebycheff Sem Ponderações.
- Aprofundar e testar a HADS para outros problemas de otimização combinatória.
- Aplicar e comparar as técnicas de programação multiobjetivos deste trabalho em outros problemas clássicos, como Problema da Mochila biobjetivo, Problema do Caixeiro Viajante, Problema do Roteamento de Veículos, Problema de Rotação de Culturas, etc.

- Aplicar e fazer comparações com outros métodos de otimização biobjetivos, como os métodos dos trabalhos em (SAYIN; KOUVELIS, 2005) e (SCHANDL; KLAMROTH; WIECEK, 2001) e analisar qual a melhor estratégia do ponto de vista custo \times benefício.

Referências

- ALIANO, F. A.; MORETTI, A. C. Um Simulated Annealing no Problema do Corte Unidimensional Inteiro. In: FREIRE, I. L. (Ed.). *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*. São Carlos: SBMAC - Sociedade de Matemática Aplicada e Computacional, 2015. v. 3, n. 1, p. 1–8.
- BAZARAA, M. S.; JARVIS, J. J.; SHERALI, H. D. *Linear Programming and Network Flows*. [S.l.]: Wiley-Interscience, 2004. ISBN 0471485993.
- BEENSON, R. *Optimization with Respect to Multiple Criteria*. Tese (Doutorado) — University of Southern California, 1971.
- BELOV, G.; SCHEITHAUER, G. A Cutting Plane Algorithm for the One-Dimensional Cutting Stock Problem with Multiple Stock Lengths. *European Journal of Operational Research*, v. 141, p. 274–294, 2002.
- BENSON, H. Existence of Efficient Solutions for Vector Maximization Problems. *Journal of Optimization Theory and Applications*, Kluwer Academic Publishers-Plenum Publishers, v. 26, n. 4, p. 569–580, 1978.
- BOWMAN, V. On the Relationship of the Tchebycheff Norm and the Efficient Frontier of Multiple-Criteria Objectives. In: THIERIEZ, H. (Ed.). *Multiple Criteria Decision Making*. [S.l.]: Springer Verlag, Berlin, 1976. (Lecture Notes in Economics and Mathematical Systems, v. 130).
- BÉRUBÉ, J.-F.; GENDREAU, M.; POTVIN, J.-Y. An Exact ε -Constraint Method for Bi-Objective Combinatorial Optimization Problems: Application to the Traveling Salesman Problem with Profits. *European Journal of Operational Research*, v. 194, n. 1, p. 39 – 50, 2009.
- CARVALHO, J. M. V. Exact Solution of Bin-packing Problems Using Column Generation and Branch-&-Bound. *Annals of Operations Research*, v. 86, p. 629–659, 1999.
- CHALMET, L. G.; LEMONIDIS, L.; ELZINGA, D. J. An Algorithm for the Bi-Criterion Integer Programming Problem. *European Journal of Operational Research*, v. 25, n. 2, p. 292–300, 1986.
- CHEITHAUER, G.; TERNO, J. The Modified Integer Round-up Property of the One-dimensional Cutting Stock Problem. *European Journal of Operational Research*, v. 84, p. 562–571, 1995.
- CLÍMACO, J.; FERREIRA, C.; CAPTIVO, M. E. Multicriteria analysis: Proceedings of the xith international conference on mcdm, 1–6 august 1994, coimbra, portugal. In: _____. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. cap. Multicriteria Integer

Programming: An Overview of the Different Algorithmic Approaches, p. 248–258. ISBN 978-3-642-60667-0. Disponível em: <http://dx.doi.org/10.1007/978-3-642-60667-0_24>.

CLÍMACO, J.; FERREIRA, C.; CAPTIVO, M. Multicriteria Integer Programming: An Overview of the Different Algorithmic Approaches. In: CLÍMACO, J. (Ed.). *Multicriteria Analysis*. [S.l.]: Springer Berlin Heidelberg, 1997. p. 248–258. ISBN 978-3-642-64500-6.

COFFMAN JR., E. G.; GAREY, M. R.; JOHNSON, D. S. Approximation algorithms for np-hard problems. In: HOCHBAUM, D. S. (Ed.). Boston, MA, USA: PWS Publishing Co., 1997. cap. Approximation Algorithms for Bin Packing: A Survey, p. 46–93. ISBN 0-534-94968-1. Disponível em: <<http://dl.acm.org/citation.cfm?id=241938.241940>>.

COHON, J. L. Book; Book/Illustrated. *Multiobjective Programming and Planning*. [S.l.]: Mineola, N.Y. : Dover Publications, 2003. ISBN 0486432637 (pbk.).

DäCHERT, K.; GORSKI, J.; KLAMROTH, K. An Augmented Weighted Tchebycheff Method with Adaptively Chosen Parameters for Discrete Bicriteria Optimization Problems. *Computational Operational Research*, Elsevier Science Ltd., v. 39, n. 12, p. 2929–2943, 2012. ISSN 0305-0548.

DEB, K.; KALYANMOY, D. *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc., 2001. ISBN 047187339X.

DYCKHOFF, H. Cutting and Packing: A Typology of Cutting and Packing Problems. *European Journal of Operational Research*, v. 44, n. 2, p. 145 – 159, 1990. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/037722179090350K>>.

EHRGOTT, M. *Multicriteria Optimization*. [S.l.]: Springer-Verlag, 2000. (Lecture Notes in Economics and Mathematical Systems).

EHRGOTT, M.; GANDIBLEUX, X. A Survey and Annotated Bibliography of Multiobjective Combinatorial Optimization. *OR-Spektrum*, Springer-Verlag, v. 22, n. 4, p. 425–460, 2000. ISSN 0171-6468.

EHRGOTT, M.; RUZIKA, S. An Improved ε -Constraint Method for Multiobjective Programming. *Journal of Optimization Theory and Applications*, v. 138, p. 375–396, 2008.

EVANS, G. An Overview of Techniques for Solving Multiobjective Mathematical Problems. *Management Science*, v. 30, n. 11, p. 1268–1282, 1984.

FARLEY, A. A.; RICHARDSON, K. V. Fixed Charge Problems With Identical Fixed Charges. *European Journal of Research Operational*, v. 18, p. 245–249, 1984.

FOERSTER, H.; WASCHER, G. Pattern Reduction in One-dimensional Cutting Stock Problem. *International Journal of Production Research*, v. 38, p. 1647–1676, 2000.

GAU, T.; WASCHER, G. CUTGEN: A Problem Generator for the Standard One-Dimensional Cutting Stock Problem. *European Journal of Operational Research*, v. 84, n. 3, p. 572–579, 1995. Disponível em: <<http://EconPapers.repec.org/RePEc:eee/ejores:v:84:y:1995:i:3:p:572-579>>.

GEHRING, H.; BORTFELDT, A. A Genetic Algorithm for Solving the Container Loading Problem. *International Transactions in Operations Research*, v. 4, p. 401–418, 1998.

- GILMORE, P. C.; GOMORY, R. E. A Linear Programming Approach to the Cutting-Stock Problem. *Operational Research*, v. 9, p. 848–859, 1961.
- _____. A Linear Programming Approach to the Cutting-Stock Problem-part II. *Operational Research*, v. 11, p. 863–888, 1963.
- GLOVER, F. Heuristics for Integer Programming Using Surrogate Constraints. *Decision Sciences*, v. 8, n. 1, p. 156–166, 1977.
- GOLFETO, R. R.; MORETTI, A. C.; SALLES, L. L. N. A Grasp Metaheuristic for the Ordered Cutting Stock Problem. *Revista Chilena de Ingeniería (En línea)*, v. 16, p. 421–427, 2008.
- _____. A Genetic Symbiotic Algorithm Applied to the One-dimensional Cutting Stock Problem. *Pesquisa Operacional (impresso)*, v. 9, p. 365–382, 2009.
- HAESSLER, R. W. Controlling Cutting Pattern Changes in One-dimensional Trim Problems. *Operational Research*, v. 23, p. 483–493, 1975.
- HAIMES, Y. Y.; LASDON, L. S.; WISMER, D. A. On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. *IEEE Transactions on Systems*, 1971.
- HARJUNKOSKI, I.; ISAKSSON, J.; WESTURLUND, T.; SKRIFVARS, H. Different Formulations for Solving Trim Loss Problems in a Paper-converting Mill with ILP. *European Symposium on Computer Aided Process Engineering*, v. 20, p. 121–126, 1996.
- HARJUNKOSKI, I.; WESTERLUND, T.; PIRN, R.; SKRIFVAS, H. Different Transformations for Solving Non-convex Trim-loss Problems by MINLP. *European Journal of Operational Research*, v. 105, p. 594–603, 1998.
- IBM ILOG CPLEX 12.5 Optimizer. [S.l.: s.n.], 2010.
- KETTANI, O.; AOUNIÏD, B.; MARTEL, J.-M. The Double Role of the Weight Factor in the Goal Programming Model. *Comput. Oper. Res.*, Elsevier Science Ltd., Oxford, UK, UK, v. 31, n. 11, p. 1833–1845, 2004. ISSN 0305-0548. Disponível em: <[http://dx.doi.org/10.1016/S0305-0548\(03\)00142-4](http://dx.doi.org/10.1016/S0305-0548(03)00142-4)>.
- KLEIN, D.; HANNAN, E. An Algorithm for the Multiple Objective Integer Linear Programming Problem. *European Journal of Operational Research*, v. 9, n. 4, p. 378 – 385, 1982. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0377221782901825>>.
- KOLEN, A. W. J.; SPIEKSMAN, F. C. R. Solving a Bi-criterion Cutting Stock Problem with Open-ended Demand: A Case Study. Palgrave Macmillan Journals, v. 51, n. 11, p. 1238–1247, 2000. ISSN 01605682, 14769360. Disponível em: <<http://www.jstor.org/stable/254209>>.
- LEDUÍNO, L. S. N. *Modelo Não Linear para Minimizar o Número de Objetos Processados e o Setup num Problema de Corte Unidimensional*. Tese (Tese de Doutorado) — Universidade Estadual de Campinas, IMECC - Campinas, 2005.

- LUQUE, M. Modified Interactive Chebyshev Algorithm (mica) for Non-Convex Multiobjective Programming. *Optimization Letters*, Springer Berlin Heidelberg, v. 9, n. 1, p. 173–187, 2015. ISSN 1862-4472.
- MARCOTE, O. An Instance of the Cutting Stock Problem for which the Rounding Property does not Hold. *Operations Research Letters*, v. 4, p. 239–243, 1986.
- MATLAB. *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.
- MAVROTAS, G.; FLORIOS, K. An Improved Version of the Augmented ε -constraint Method (augmecon2) for Finding the Exact Pareto Set in Multi-Objective Integer Programming Problems. *Applied Mathematics and Computation*, v. 219, n. 18, p. 9652–9669, 2013.
- MIETTINEN, K. *Nonlinear Multiobjective Optimization*. [S.l.]: Kluwer Academic Publishers Dordrecht, 1999. v. 12. (International Series in Operations Research and Management Science, v. 12).
- MIETTINEN, K.; ESKELINEN, P.; LUQUE, M.; RUIZ, F. On the Use of Preferential Weights in Interactive Reference Point Based Methods. In: *Multiobjective Programming and Goal Programming*. [S.l.]: Springer Berlin Heidelberg, 2009, (Lecture Notes in Economics and Mathematical Systems, v. 618). p. 211–220. ISBN 978-3-540-85645-0.
- NEUMAYER, P.; SCHWEIGERT, D. Three Algorithms for Bicriteria Integer Linear Programs. *Operations-Research-Spektrum*, Springer-Verlag, v. 16, n. 4, p. 267–276, 1994.
- OZLEN, M.; BURTON, B. A.; MACRAE, C. A. G. Multi-Objective Integer Programming: An Improved Recursive Algorithm. *Journal of Optimization Theory and Applications*, Springer US, v. 160, n. 2, p. 470–482, 2014. ISSN 0022-3239.
- POLDI, K. C.; ARENALES, M. N. Heurísticas para o Problema do Corte Unidimensional Inteiro. *Pesquisa Operacional*, scielo, v. 26, p. 473 – 492, 12 2006. ISSN 0101-7438.
- _____. Heuristics for the One-dimensional Cutting Stock Problem with Limited Multiple Stock Lengths. *Computers and Operations Research*, v. 36, p. 2074 – 2081, 2009.
- PRZYBYLSKI, A.; GANDIBLEUX, X.; EHRGOTT, M. Computational Results for Four Exact Methods to Solve the Three-Objective Assignment problem. In: *Multiobjective Programming and Goal Programming*. [S.l.]: Springer Berlin Heidelberg, 2009. v. 618, p. 79–88. ISBN 978-3-540-85645-0.
- RALPHS, T. K.; SALTZMAN, M. J.; WIECEK, M. M. An Improved Algorithm for Solving Biobjective Integer Programs. *Annals of Operations Research*, Springer US, v. 147, n. 1, p. 43–70, 2006. ISSN 0254-5330. Disponível em: <<http://dx.doi.org/10.1007/s10479-006-0058-z>>.
- RUZIKA S.AND WIECEK, M. M. A Survey of Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, v. 126, n. 3, p. 473–501, 2003. ISSN 1573-2878.
- SAYIN, S.; KOUVELIS, P. The Multiobjective Discrete Optimization Problem: A Weighted Min-Max Two-Stage Optimization Approach and a Bicriteria Algorithm. *Management Science*, v. 51, n. 10, p. 1572–1581, 2005.

- SCHANDL, B.; KLAMROTH, K.; WIECEK, M. Norm-Based Approximation in Bicriteria Programming. *Computational Optimization and Applications*, Kluwer Academic Publishers, v. 20, n. 1, p. 23–42, 2001. ISSN 0926-6003. Disponível em: <<http://dx.doi.org/10.1023/A%3A1011267305444>>.
- SOLANKI, R. Generating the Noninferior Set in Mixed Integer Biobjective Linear Programs: An Application to a Location Problem. *Comput. Oper. Res.*, Elsevier Science Ltd., Oxford, UK, UK, v. 18, n. 1, p. 1–15, jan. 1991. ISSN 0305-0548. Disponível em: <[http://dx.doi.org/10.1016/0305-0548\(91\)90037-R](http://dx.doi.org/10.1016/0305-0548(91)90037-R)>.
- STEUER, R.; CHOO, E.-U. An Interactive Weighted Tchebycheff Procedure for Multiple Objective Programming. *Mathematical Programming*, Springer-Verlag, v. 26, n. 3, p. 326–344, 1983. ISSN 0025-5610. Disponível em: <<http://dx.doi.org/10.1007/BF02591870>>.
- SYLVA, J.; CREMA, A. A Method for Finding the Set of Non-dominated Vectors for Multiple Objective Integer Linear Programs. *European Journal of Operational Research*, v. 158, n. 1, p. 46–55, 2004. Disponível em: <[http://dx.doi.org/10.1016/S0377-2217\(03\)00255-8](http://dx.doi.org/10.1016/S0377-2217(03)00255-8)>.
- UMETANI, S.; YAGIURA, M.; IBARAKI, T. One-Dimensional Cutting Stock Problem to Minimize the Number of Different Patterns. *European Journal of Operational Research*, v. 146, n. 2, p. 388 – 402, 2003.
- VANCE, P. H. A Branch-&-Cut-&-Price Algorithm for One-Dimensional Stock Cutting and Two-Dimensional Two-Stage Cutting. *Computational Optimization and Applications*, v. 9, p. 211–228, 1998.
- VANDERBECK, F. An Exact Algorithm for IP Column Generation. *Operations Research Letters*, v. 19, p. 151–159, 1996.
- WASCHER, G.; GAU, G. Heuristics for the Integer One-dimensional Cutting Stock Problem: a Computational Study. *Operational Research Spektrum*, v. 18, p. 131–144, 1996.
- WASCHER, G.; HAUBNER, H.; SCHUMANN, H. An Improved Typology of Cutting and Packing Problems. *European Journal of Operational Research*, v. 183, n. 3, p. 1109 – 1130, 2007. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S037722170600292X>>.
- ÖZLEN, M.; AZIZOGLU, M. Multi-objective Integer Programming: A General Approach for Generating all Non-dominated Solutions. *European Journal of Operational Research*, v. 199, n. 1, p. 25–35, 2009.

Anexo I

Suponhamos que cada ponto $z^i \in \mathcal{Z} = \{z^1, \dots, z^n\} \subseteq \mathbb{R}^r$ seja a imagem de um ponto $x \in \mathcal{X}$ num POM onde todas as funções objetivo são minimizadas. Proposto em (DEB; KALYANMOY, 2001) e denominado de Filtro de Pareto, a saída do procedimento a seguir será o conjunto \mathcal{Z}^* de todos os pontos não-dominados.

Algoritmo 11: Filtro de Pareto

- 1: Entrada: conjunto \mathcal{Z}
 - 2: Inicialize $\mathcal{Z}^* = \{z^1\}$. Seja o contador de soluções $i = 2$
 - 3: Seja $j = 1$
 - 4: Compare a solução z^i com z^j de \mathcal{Z}^*
 - 5: Se $z^i \leq z^j$, exclua z^j de \mathcal{Z}^* ou atualize $\mathcal{Z}^* = \mathcal{Z} - \{z^j\}$. Se $j < |\mathcal{Z}^*|$, então incremente j por 1 e vá ao Passo 4. Senão, vá ao Passo 6. Alternativamente, se j -ésimo membro de \mathcal{Z}^* domina z^i , incremente i por 1 e vá ao Passo 3.
 - 6: Insira z^i em \mathcal{Z}^* , ou atualize $\mathcal{Z}^* = \mathcal{Z}^* \cup \{z^i\}$. Se $i < n$, incremente i por 1 e vá ao Passo 3. Senão, pare, e declare \mathcal{Z}^* como o conjunto não-dominado.
 - 7: Saída: \mathcal{Z}^*
-

O algoritmo descrito a seguir constrói heurísticamente um padrão de corte admissível para o PCUI.

Algoritmo 12: Construção de um Padrão de Corte Admissível

- 1: Entrada: m, ℓ, q, L e v
 - 2: $perda = L$
 - 3: **while** $perda > \eta$ **do**
 - 4: $a_j = 0^{m \times 1}$
 - 5: **for** $k = 1, \dots, m$ **do**
 - 6: $i = v_k$
 - 7: Gere um número aleatório $r \in [0, 1]$
 - 8: Calcule $g = \left\lfloor \frac{r \cdot perda}{l_i} \right\rfloor$
 - 9: **if** Se o número de facas não foi excedido **then**
 - 10: $a_{ij} = g$
 - 11: **end if**
 - 12: Atualize $perda = L - \ell^T a_j$
 - 13: **end for**
 - 14: **end while**
 - 15: Saída: \mathbf{a}_j
-

O vetor v denota a ordem em que os itens são alocados dentro do padrão a ser

construído. Esse vetor é constantemente atualizado e calculado da seguinte forma: os itens que aparecem com maior número de vezes têm maior preferência em ocupar as primeiras deste vetor, em contraste com os que têm maior frequência. Quando um novo padrão é inserido, este vetor é atualizado e o procedimento é iterado até que o número de padrões pré-determinado seja atendido.

Na Tabela (41) fornecemos os tempos computacionais médios para o algoritmo GC (em seg.) bem como o número de iterações necessários na fase 2.

Tabela 41 – Resultados computacionais para o algoritmo Gilmore e Gomory - Fase 2

Classe	Tempo (s)		Iterações	
	Média	λ	Média	λ
1	2.03	0.43	19.65	0.33
2	3.58	0.43	34.1	0.33
3	3.98	0.25	37.05	0.26
4	0.83	0.46	9.4	0.46
5	2.14	0.37	23.7	0.35
6	3.05	0.31	34.95	0.27
7	0.55	0.44	6.6	0.40
8	1.59	0.26	19.4	0.26
9	2.27	0.26	26.8	0.22
Média	2.22	0.34	23.52	0.30
10	3.92	0.35	37.35	0.31
11	4.69	0.29	48.2	0.27
12	7.41	0.27	72.75	0.26
13	3.1	0.43	32.85	0.37
14	7.04	0.38	73.15	0.34
15	9.03	0.37	91.6	0.34
16	1.64	0.37	20.3	0.35
17	4.34	0.26	48.45	0.22
18	4.97	0.28	55.4	0.24
Média	5.13	0.33	53.34	0.30
19	7.56	0.29	72.15	0.25
20	9.54	0.38	94.15	0.32
21	13.42	0.32	123.5	0.23
22	17.75	0.62	180.25	0.52
23	29.06	0.45	268.55	0.35
24	30.12	0.36	284.3	0.34
25	7.21	0.30	63.85	0.28
26	6.53	0.33	67.35	0.29
27	14.6	0.39	125.7	0.30
Média	15.09	0.40	142.2	0.34
Média Geral	7.48	0.38	73.02	0.33

Na Tabela (42) apresentamos os tempos computacionais médios para o algoritmo GC (em seg.) bem como o número de iterações necessários na fase 3.

Tabela 42 – Resultados computacionais para o algoritmo Gilmore e Gomory - Fase 3

Classe	Tempo (s)		Iterações	
	Média	λ	Média	λ
1	13.43	0.22	74.45	0.21
2	26	0.28	144.97	0.25
3	14.66	0.22	92.96	0.22
4	17.54	0.48	105.64	0.34
Média	17.91	0.31	104.5	0.26
5	29.74	0.29	139.6	0.27
6	44.08	0.31	217.1	0.31
7	30.52	0.34	175.7	0.29
8	56.14	0.36	268.88	0.28
Média	40.12	0.33	200.32	0.29
9	34.15	0.20	160.4	0.19
10	55.8	0.34	273.8	0.34
11	52.35	0.24	277.62	0.23
12	106.99	0.35	453.99	0.28
Média	62.32	0.30	291.45	0.27
Média Geral	40.12	0.31	198.76	0.28