



Tiago Tiburcio da Silva

*Um estudo sobre limites duais para o problema  
integrado de dimensionamento de lotes e  
sequenciamento da produção*

Dissertação de Mestrado  
Pós-Graduação em Matemática

Silva, Tiago Tiburcio da.

Um estudo sobre limites duais para o problema integrado de dimensionamento de lotes e sequenciamento da produção / Tiago Tiburcio da Silva. -- São José do Rio Preto, 2015

119 f. : il., tabs.

Orientador: Maria do Socorro Nogueira Rangel

Dissertação (mestrado) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Matemática. 2. Otimização matemática. 3. Métodos de relaxação (Matemática) 4. Problema de dimensionamento de lotes. 5. Algoritmos. I. Rangel, Maria do Socorro Nogueira. II. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 518.734

Ficha catalográfica elaborada pela Biblioteca do IBILCE  
UNESP - Câmpus de São José do Rio Preto

Tiago Tiburcio da Silva

*Um estudo sobre limites duais para o  
problema integrado de dimensionamento de  
lotes e sequenciamento da produção*

Orientador:

Prof<sup>a</sup>. Dra. Maria do Socorro Nogueira Rangel

UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"  
INSTITUTO DE BIOCÊNCIAS, LETRAS E CIÊNCIAS EXATAS  
CAMPUS DE SÃO JOSÉ DO RIO PRETO

São José do Rio Preto

13 de agosto de 2015

Tiago Tiburcio da Silva

*Um estudo sobre limites duais para o problema integrado de dimensionamento de lotes e sequenciamento da produção*

Dissertação apresentada para obtenção do título de Mestre em Matemática, área de Análise Aplicada, junto ao Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de São José do Rio Preto.

BANCA EXAMINADORA

---

Prof<sup>ª</sup>. Dra. Maria do Socorro Nogueira Rangel  
Professor Doutor  
UNESP - São José do Rio Preto  
Orientador

---

Prof. Dr. Silvio Alexandre de Araújo  
Professor Doutor  
UNESP - São José do Rio Preto

---

Prof<sup>ª</sup>. Dra. Deisemara Ferreira  
Professor Doutor  
UFSCar - Sorocaba

São José do Rio Preto, 13 de agosto de 2015.

Aos meus amados pais, Abigail e Valdecir.

À minha noiva Aline e amigos.

*Dedico.*

# Agradecimentos

A Deus, por sua infinita bondade e misericórdia que tem tido comigo, e pelo rico dom da vida.

A minha orientadora Socorro a quem eu devo meu ingresso no mundo acadêmico. Obrigado pelas conversas com quem tive o prazer de desfrutar, me esclarecendo questões sobre o meu trabalho, por me lembrar do frescor que é preciso ter na pesquisa acadêmica e na vida, e por seus preciosos conselhos pessoais. Tenho muito orgulhoso em ser seu orientando. Quando a procurei, no início da minha caminhada, tinha apenas uma intuição. Hoje tenho imensa admiração. Muito obrigado!

Aos meus pais Abigail e Valdecir, e minha irmã Juliana por terem me ensinado que tudo é possível. Não pouparam esforços para me ajudar, sempre ao meu lado, me apoiando, me incentivando, vibrando com minhas conquistas. Devo tudo à vocês. Sou grato a Deus por ter essa família linda e unida.

A toda a minha família pelo incentivo e carinho ao longo de todos esses anos.

A minha noiva Aline, por estar todo o tempo ao meu lado, incondicionalmente. Nos momentos mais difíceis, que não foram raros, sempre me fazendo acreditar que chegaria ao fim desta difícil, mas gratificante etapa. Sou grato por cada gesto carinhoso, cada sorriso, e ansioso por estar ao seu lado o resto da minha vida. Te Amo!

Aos meus amigos desde a graduação Leonardo, Lidia, Cinthia, Mariana, Rafaella, Ana Maria, Luiz Fernando, em especial o Rafael. Por compartilharmos os sufocos das provas, trabalhos, aulas, mas por todos os momentos de alegria, felicidades e viagens que fizemos juntos. Aos amigos Well, Tavin. Muito obrigado!

As companheiras da salinha da pós-graduação e grandes amigas Gislaine, Michelli, por todos os momentos de alegria, pelos conselhos e ajuda.

Ao irmão que Deus me deu, Simão Pedro, por muitas e muitas vezes puxar minha orelha, me aconselhar na vida, por acreditar em mim, com seu jeito todo espivado, e por todos os momentos de diversão e alegria que partilhamos.

A todos os professores, colegas, pessoas e funcionários do IBILCE, em especial ao Getúlio e Thiago.

Ao CNPq, pela credibilidade e apoio financeiro a este projeto.

E finalmente a todos que contribuíram direta ou indiretamente na realização deste trabalho. Sintam-se todos agradecidos.

“Posso todas as coisas naquele que me fortalece”

Filipenses 4.13

# Resumo

A Matemática está presente no nosso dia-a-dia seja pra dizer as horas, contar dinheiro, prever o tempo. Sob o aspecto empresarial ela também se faz presente na hora de tomar decisões, por exemplo. Muitas empresas de manufatura lidam com decisões diariamente no setor de produção, dimensionando lotes e sequenciando sua produção. Entretanto, o mais comum é tomar essas decisões de forma independente, sendo que poderiam ser tomadas simultaneamente, pois agregariam melhores resultados. Neste trabalho integramos essas decisões utilizando um modelo matemático que agrega ao problema de dimensionamento de lotes, o sequenciamento da produção modelando a exclusão de subsequências através das restrições do tipo MTZ e MCF. Também estudamos essas duas formulações considerando a variável de preparo explicitamente e implicitamente resultando em quatro formulações matemáticas diferentes para o problema integrado de dimensionamento de lotes e sequenciamento da produção. Concluimos que a formulação MCF com variável de preparo explícita é mais forte que as outras formulações estudadas e que as soluções das instâncias das formulações baseadas nas restrições do tipo MTZ são bastante influenciadas pelos planos de cortes e pré-processamento inclusos no *solver* CPLEX. Nosso objetivo é derivar limitantes primais e duais para o problema integrado de dimensionamento de lotes e sequenciamento da produção. Para a obtenção dos limitantes primais foi proposta uma heurística gulosa. Para obter os limites duais foram estudadas a relaxação Lagrangeana e a relaxação Lagrangeana/Surrogate e os métodos usados para resolução dos duais associados foram o Algoritmo de Subgradiente e Algoritmo de Volume. O método que obteve melhor desempenho foi o dual Lagrangeano/Surrogate resolvido pelo Algoritmo de Subgradiente para a formulação com restrições do tipo MTZ e variável explícita de preparo.

Palavras-Chave: problema integrado de dimensionamento de lotes e sequenciamento da produção, relaxação Lagrangeana, relaxação Lagrangeana/Surrogate.



# Abstract

Mathematics is present in our daily routine to tell time, count money, predict the weather. Many manufacturing companies deal with daily decisions in the manufacturing sector, lot-sizing and sequencing their production. However, the most usual is to take these decisions considering two independent problems, and not simultaneously, as it adds better results. In this work we integrate these decisions through a mathematical model that adds to the problem of lot sizing, sequencing decisions using constraints of the type MTZ and MCF. We also study these two formulations, considering the set up decisions explicitly and implicitly resulting in four different mathematical formulations for the integrated problem. We conclude that the MCF formulation with the explicit set up variable is stronger than the other formulations studied and the solutions of the instances of formulations based on constraints of MTZ type are strongly influenced by the cutting planes and pre-processing included in the solver CPLEX. We aimed to derive primal and dual bounds for the integrated problem of lot sizing and sequencing of production. To obtain the primal bound we proposed a greedy heuristic. The dual bounds were obtained studying the Lagrangean and the Lagrangean / Surrogate relaxation and the methods used to solve the dual associates were the subgradient algorithm and Volume algorithm. The method with better performance was the dual Lagrangian / Surrogate solved by subgradient Algorithm for formulation with constraints MTZ type and explicit set up variable.

Keywords: lot-sizing and scheduling problems, Lagrangean relaxation, Lagrangean/Surrogate relaxation.

# Sumário

<b>Introdução</b>	<b>7</b>
<b>1 O Problema PIDS</b>	<b>9</b>
1.1 Descrição do problema . . . . .	9
1.2 Modelos matemáticos . . . . .	10
<b>2 Limitantes primais e duais para PI</b>	<b>18</b>
2.1 Limitantes primais . . . . .	18
2.2 Limitantes duais . . . . .	19
2.2.1 Planos de corte . . . . .	20
2.2.2 Relaxação Lagrangeana . . . . .	21
2.2.3 Relaxação Surrogate . . . . .	32
2.2.4 Relaxação Lagrangeana/Surrogate . . . . .	34
<b>3 Solução do problema dual Lagrangeano</b>	<b>36</b>
3.1 Conceitos básicos . . . . .	36
3.2 Algoritmo do Subgradiente . . . . .	43
3.3 Algoritmo de Volume . . . . .	48
<b>4 Relaxações para o PIDS</b>	<b>54</b>
4.1 Uma heurística gulosa para o PIDS . . . . .	54
4.2 Relaxação Lagrangeana para o PIDS . . . . .	56
4.2.1 Algoritmo de Subgradiente para resolução do Dual Lagrangeano . . . . .	58
4.2.2 Algoritmo de Volume para resolução do Dual Lagrangeano . . . . .	58
4.3 Relaxação Lagrangeana/Surrogate para o PIDS . . . . .	61
4.3.1 Algoritmo de Subgradiente para resolução do Dual Lagrangeano/Surrogate . . . . .	63
<b>5 Estudo computacional</b>	<b>66</b>
5.1 Geração das Instâncias . . . . .	66
5.2 Comparação entre os modelos . . . . .	67
5.3 Comparação entre os limites duais . . . . .	78
5.4 Considerações finais . . . . .	85

<b>6</b>	<b>Conclusões e perspectivas futuras</b>	<b>87</b>
<b>A</b>	<b>Modelos para o PIDS</b>	<b>96</b>
A.1	Modelo MTZ1S1M (ver descrição na seção 1.2) . . . . .	96
A.2	Modelo MTZ1S1MP (ver descrição na seção 1.2) . . . . .	97
A.3	Modelo MCF1S1M (ver descrição na seção 1.2) . . . . .	98
A.4	Modelo MCF1S1MP (ver descrição na seção 1.2) . . . . .	99
<b>B</b>	<b>Gráficos - instâncias dados14 – 0</b>	<b>100</b>
<b>C</b>	<b>Gráficos - instâncias dados18 – 0</b>	<b>104</b>
<b>D</b>	<b>Gráficos - instâncias dados22 – 0</b>	<b>108</b>

# Lista de Figuras

1.1	Plano de produção e sequenciamento. . . . .	14
1.2	Dados de uma instância do problema PIDS. . . . .	16
1.3	Plano de produção e sequenciamento. . . . .	17
1.4	Estoque da produção. . . . .	17
2.1	Conjuntos factíveis $Q$ , $\text{conv}(Q)$ e $\text{conv}(V)$ , e soluções do problema inteiro, do problema dual Lagrangeano e da relaxação linear. . . . .	25
2.2	Gráficos das funções $Z(\pi, x^i)$ , $i = 1, 2, \dots, 8$ . . . . .	28
2.3	Interpretação geométrica da relaxação Lagrangeana [62]. . . . .	32
2.4	Comportamento da função Lagrangeano/Surrogate em função de $t$ [64]. . . . .	35
3.1	$f(x) =  x $ . . . . .	39
3.2	Os elementos $y^1$ e $y^2$ são subgradientes de $f$ em $x$ , isto é, para todo $z$ tem-se que $f(z) \geq f(x) + \langle y^i, z - x \rangle$ , $i = 1, 2$ . . . . .	40
3.3	Gráfico de $f(x) =  x_1  + 2 x_2 $ . . . . .	41
3.4	Pseudocódigo do Método de Descida. . . . .	42
3.5	Pseudocódigo do Algoritmo de Subgradiente . . . . .	43
3.6	Um passo do método de feixes em que foi gerado um ponto $x^{k+1}$ com valor de $f$ menor de $f(x^k)$ . . . . .	49
3.7	Pseudocódigo do Método de Feixes. . . . .	50
3.8	Pseudocódigo do Algoritmo de Volume . . . . .	52
3.9	Algoritmo para a escolha de $\alpha$ . . . . .	53
4.1	Pseudocódigo para a heurística gulosa . . . . .	57
4.2	Pseudocódigo do Algoritmo de Subgradiente para a formulação MTZ1S1MP - Dual Lagrangeano. . . . .	58
4.3	Pseudocódigo do Algoritmo de Volume para a formulação MTZ1S1MP - Dual Lagrangeano. . . . .	61
4.4	Pseudocódigo do Algoritmo de Subgradiente para a formulação MTZ1S1MP - Dual Lagrangeano/Surrogate. . . . .	63
4.5	Algoritmo de busca unidimensional para o dual Lagrangeano/Surrogate. . . . .	65

B.1	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados14-0 - Modelo MTZ1S1MP. . . . .	100
B.2	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MTZ1S1MP. . . . .	101
B.3	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados14-0 - Modelo MTZ1S1MP. . . . .	101
B.4	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados14-0 - Modelo MCF1S1MP. . . . .	102
B.5	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MCF1S1MP. . . . .	102
B.6	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados14-0 - Modelo MCF1S1MP. . . . .	103
C.1	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados18-0 - Modelo MTZ1S1MP. . . . .	104
C.2	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MTZ1S1MP. . . . .	105
C.3	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados18-0 - Modelo MTZ1S1MP. . . . .	105
C.4	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados18-0 - Modelo MCF1S1MP. . . . .	106
C.5	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MCF1S1MP. . . . .	106
C.6	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados18-0 - Modelo MCF1S1MP. . . . .	107
D.1	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados22-0 - Modelo MTZ1S1MP. . . . .	108
D.2	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MTZ1S1MP. . . . .	109
D.3	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados22-0 - Modelo MTZ1S1MP. . . . .	109
D.4	Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradi- entes - Dados22-0 - Modelo MCF1S1MP. . . . .	110
D.5	Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MCF1S1MP. . . . .	110
D.6	Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados22-0 - Modelo MCF1S1MP. . . . .	111

# Lista de Tabelas

5.1	Parâmetros usados na geração das instâncias. . . . .	67
5.2	Classes de instâncias [56]. . . . .	67
5.3	Resultados para as formulações MTZ1S1M e MTZ1S1MP (estratégia CPLEX). .	69
5.4	Planos de cortes gerados para instâncias dos modelos MTZ1S1M e MTZ1S1MP, aqui renomeadas por M1 e M2, respectivamente. . . . .	71
5.5	Limitante superior obtido antes da ramificação da árvore - Modelo MTZ1S1MP.	72
5.6	Resultados para as formulações MCF1S1M e MCF1S1MP (estratégia CPLEX). .	74
5.7	Planos de cortes gerados para instâncias dos modelos MCF1S1M e MCF1S1MP, aqui renomeadas por MCF e MCFP, respectivamente. . . . .	75
5.8	Limitante superior obtido no nó raiz antes da ramificação da árvore - Modelo MCF1S1MP. . . . .	76
5.9	Limitantes superiores obtidos do CPLEX ( <i>Branch-and-bound</i> - 3.600 segundos).	77
5.10	Comparações entre os Métodos - Modelo MTZ1S1MP - 600 segundos. . . . .	80
5.11	Comparações entre os Métodos - Modelo MTZ1S1MP - 3.600 segundos. . . . .	81
5.12	Comparações entre os Métodos - Modelo MCF1S1MP- 600 segundos. . . . .	83
5.13	Comparações entre os Métodos - Modelo MCF1S1MP - 3.600 segundos. . . . .	84
5.14	Valores dos $GAP_C^{SU}$ - Modelo MTZ1S1MP - 3.600 segundos. . . . .	85
5.15	Valores médios dos $GAP_C^{SU}$ e seus respectivos desvios padrões. . . . .	86
5.16	Comparações entre os limitantes inferiores obtidos pelo dual lagrangeano/surrogate e a relaxação linear. . . .	86

# Introdução

Todos usam a Matemática todos os dias: tanto para dizer as horas, prever o tempo, contar dinheiro, como para auxiliar na tomada de decisões. Em situações mais complicadas surge a necessidade de uma teoria matemática para ajudar o ser humano a não só resolver os seus problemas, mas escolher, dentre diversas alternativas, a melhor delas. À este processo denominamos Otimização. A otimização, pode ser definida como: *"o conjunto de métodos capazes de determinar as melhores configurações possíveis para a construção ou o funcionamento de sistemas de interesse para o ser humano"* [75].

Sob o ponto de vista empresarial, a Otimização também é de grande valia. Grandes empresas hoje em dia, tem em seu ferramental, a Matemática, a Otimização, que muitas vezes auxiliam na identificação de problemas que a olho nu não são capazes de serem vistos, e os resolve de tal forma que os benefícios consequentes dessa resolução eram inimagináveis.

Mais especificamente, no setor de manufaturas, a indústria tem sido pressionada a tornar seus processos mais eficientes, posicionando-se frente à concorrência de uma forma mais agressiva. As decisões de produção nesse setor de manufatura objetivam determinar o uso mais eficiente dos recursos para atender a produção dos itens requeridos pelos clientes. Estas decisões são tomadas geralmente em dois níveis: um nível de dimensionamento de lotes e um nível de programação [5]. No nível de dimensionamento de lotes, o objetivo é determinar um plano de produção de curto prazo, ou seja, quanto de cada item produzir a fim de atender as demandas previstas, sob as condições e capacidades operacionais existentes. No nível de programação, o objetivo é determinar a ordem em que os itens serão produzidos de modo a reduzir tempos e custos [84].

Na prática, essas decisões ainda são tomadas de forma independentes mas deveriam ser tomadas conjuntamente. Na teoria, diversos modelos foram formulados a fim de integrar as decisões sobre o sequenciamento da produção às decisões de dimensionamento de lotes [18], [19], [22], [43].

Uma proposta para integrar essas decisões, é o Problema Integrado de Dimensionamento de Lotes e Sequenciamento da Produção (PIDS), é utilizar as restrições de eliminação de subrotas do problema do caixeiro viajante na formulação matemática para o problema de dimensionamento de lotes, para obter o sequenciamento dos lotes. Neste trabalho usaremos duas propostas: a primeira acrescentando inequações de eliminação de subsequências propostas por Miller, Tucker e Zemlin (MTZ) [63], e a segunda, é uma formulação do tipo *Multi Commodity Flow* (MCF),

introduzida por Claus [13]. Uma questão importante é como determinar limitantes primais e duais para as formulações apresentadas. Para tanto, propomos três métodos (dual Lagrangeano resolvido pelo algoritmo de subgradiente, dual Lagrangeano resolvido pelo algoritmo de volume e dual Lagrangeano/Surrogate resolvido pelo algoritmo de subgradiente) para obtenção dos limitantes duais e uma heurística gulosa para os limitantes primais. A obtenção de bons limitantes é importante, por exemplo, quando não temos o CPLEX para resolução do problema.

No Capítulo 1, é apresentada uma breve revisão do PIDS e é proposta uma nova formulação em que é explicitada a variável de preparo. Nos Capítulos 2 e 3, fazemos um estudo sobre limites duais e primais para problemas de otimização linear inteiro e a teoria envolvida para obtê-los. No Capítulo 4, apresentamos uma heurística gulosa para determinar limitantes primais para o PIDS, e alguns métodos para determinar limitantes duais para o PIDS. No Capítulo 5 testamos computacionalmente as técnicas desenvolvidas, e por fim, no Capítulo 6 são apresentadas as conclusões e trabalhos futuros.



# Capítulo 1

## O Problema PIDS

Neste capítulo apresentamos uma descrição e a motivação do problema PIDS para este estudo, uma breve revisão bibliográfica, e apresentamos os modelos estudados, assim como uma reformulação destes modelos.

### 1.1 Descrição do problema

As decisões de produção no setor de manufatura, por exemplo, objetiva determinar o uso mais eficiente dos recursos utilizados para atender a demanda dos itens requeridos pelos clientes. Estas decisões são tomadas geralmente em dois níveis: um nível de dimensionamento e um nível de programação dos lotes [5]. No nível de dimensionamento, o objetivo é determinar um plano de produção, ou seja, quanto de cada item produzir a fim de atender as demandas previstas, sob as condições e capacidades operacionais existentes. No nível de programação, o objetivo é ordenar essa produção de modo a reduzir tempos e custos de trocas entre os itens [84].

Na prática, essas decisões ainda são tomadas de forma independentes mas deveriam ser tomadas de forma conjunta. Como exposto por Wolosewicz *et al* [84], os modelos matemáticos para resolver os problemas de dimensionamento de lotes e sequenciamento da produção, consideram capacidade agregada, não garantindo que o plano de produção seja factível quando levado ao nível de programação. Se essa capacidade for superestimada, o plano de produção não representará a realidade em termos de recursos disponíveis, causando atrasos e clientes não satisfeitos. Por outro lado, se a capacidade for subestimada, os itens serão produzidos num tempo menor, acarretando maior custo de estocagem. Se torna então muito importante lidar simultaneamente com estes dois problemas.

Um outro exemplo em que é essencial esta integração, é quando as decisões de sequenciamento não estão ligadas diretamente com os itens, mas sim com a sequência de produção de determinados materiais que serão utilizados para produzir um grupo de itens demandados. Neste caso podemos diminuir a quantidade de trocas, agrupando itens que usam o mesmo

material.

O PIDS consiste em determinar quanto de cada item produzir e em que ordem serão produzidos, de modo a atender as demandas previstas e reduzir custos e tempos de produção, ou seja, determinar simultaneamente a quantidade a ser produzida em cada máquina, qual o período e em que ordem.

Várias são as técnicas propostas pelos pesquisadores para resolver o problema integrado de dimensionamento de lotes e sequenciamento da produção. Dauzère-Pérès e Lasserre [14] [15] formularam um modelo considerando problemas de sequenciamento e capacidade exata, e resolveram iterando entre o plano de dimensionamento e o de programação. Sikora, Chhajer e Shaw [74] abordaram o problema de forma separada, resolvendo o problema período por período, do primeiro ao último período do horizonte de planejamento, sendo que em cada período usaram uma heurística tipo Silver-Meal [73] para utilizar a capacidade o quanto for possível. Algumas recentes revisões bibliográficas podem ser encontradas em Zhu e Wilhelm [87], e Jans e Degraeve [44] [40].

O problema integrado de dimensionamento de lotes e sequenciamento da produção com preparo dependente da sequência tem sido tratado frequentemente com modelos que subdividem o período em subperíodos [22]. Em muitos casos, esta estratégia condiz com a realidade, em que cada subperíodo seria o período de 1 dia, por exemplo, sendo que o período seria 1 semana.

O modelo proposto por Fleischmann e Meyr [22], tem sido utilizado para formular várias aplicações práticas: Araujo *et al* [3] [4] lidaram com problemas em pequenas fundições, explorando esquemas *relax-and-fix* [85] implementados em um horizonte rolante de planejamento; Araújo *et al.* [3] propôs variações para o método de busca local; Ferreira *et al.* [20] e Toledo *et al.* [77] trataram o problema em uma indústria de bebidas, onde sincronizam os estágios de preparação da bebida e o engarrafamento, sendo que Ferreira *et al.* [20] propõem várias estratégias *relax-and-fix* e Toledo *et al.* [77] usam um algoritmo genético para sua resolução; Toso *et al.* [78] trataram o problema quanto a produção de compostos para alimentação animal propondo heurísticas *relax-and-fix* para sua resolução. Wolosewicz *et al.* [84] propôs um modelo para o problema integrado de dimensionamento de lotes e sequenciamento da produção com sequência fixa de operações nas máquinas, e o método de resolução proposto é uma heurística Lagrangeana.

## 1.2 Modelos matemáticos

Wolosewicz *et al* [84] afirmam que os problemas de dimensionamento de lotes e sequenciamento da produção tem que ser lidados simultaneamente, e Clark *et al* [11] afirma que quando os modelos de dimensionamento de lotes são incorporados aos de sequenciamento da produção, há uma influência considerável na produção.

O problema PIDS consiste em determinar simultaneamente a quantidade a ser produzida em cada máquina, qual o período e em que ordem.

O modelo MTZ1S1M proposto por Maldonado, Rangel e Clark [57] consiste em utilizar as restrições MTZ de eliminação de subrotas para eliminar subsequências não desejadas [63]. Já o modelo MCF1S1M, proposto em [58] utiliza as restrições tipo *Multi Commodity Flow* (MCF).

Na descrição dos modelos, iremos considerar um problema com um total de  $J$  itens e  $T$  períodos no horizonte de planejamento, e os índices  $i, j = 1, 2, \dots, J$  para representar os itens e  $t = 1, 2, \dots, T$  para representar os períodos.

Os parâmetros e variáveis de decisão utilizados para modelar o problema são apresentados em seguida:

# *Parâmetros:*

- $h_j$  : custo de estocar uma unidade do item  $j$ ;
- $g_j$  : custo de atrasar uma unidade do item  $j$ ;
- $s_{ij}$  : custo de troca de produção de um item  $i$  para o item  $j$ ;
- $p_j$  : tempo de produzir um item  $j$ ;
- $b_{ij}$  : tempo de troca de um item  $i$  para o item  $j$ ;
- $d_{jt}$  : demanda do item  $j$  no período  $t$ ;
- $C_t$  : tempo total disponível para produção dos itens no período  $t$  (capacidade da máquina);

# *Variáveis de decisão:*

- $x_{jt}$  : quantidade produzida do item  $j$  no período  $t$ ;
- $I_{jt}^+$  : quantidade estocada do item  $j$  no período  $t$ ;
- $I_{jt}^-$  : quantidade em atraso do item  $j$  no período  $t$ ;
- $z_{ijt} : \begin{cases} 1, & \text{se há troca do item } i \text{ para o item } j \text{ no período de tempo } t; \\ 0, & \text{caso contrário.} \end{cases}$
- $u_{jt}$  : ordem numérica em que o item  $j$  é produzido no período  $t$ ;

O modelo MTZ1S1M é descrito em (1.1) – (1.12).

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt} \quad (1.1)$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t \quad (1.2)$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t \quad (1.3)$$

$$x_{jt} \leq \frac{C_t}{p_j} \left( \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right), \quad \forall j, \forall t \quad (1.4)$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t \quad (1.5)$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t \quad (1.6)$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t \quad (1.7)$$

$$u_{jt} \geq u_{it} + 1 - J(1 - z_{ijt}), \quad \forall i, \forall j, i \neq j, \forall t \quad (1.8)$$

$$1 \leq u_{jt} \leq J \quad \forall j, t \quad (1.9)$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t \quad (1.10)$$

$$I_{j0}^+ = 0, \quad \forall j \quad (1.11)$$

$$I_{j0}^- = 0, \quad \forall j \quad (1.12)$$

A função objetivo (1.1) minimiza custos de estoque, atraso e troca entre os itens. As restrições (1.2) são as restrições de balanceamento do estoque com a produção, garantindo que a demanda de cada item  $j$  em cada período de tempo  $t$  seja atendida. As restrições (1.3) são as restrições de capacidade, ou seja, impedem que o tempo de produção ultrapasse a capacidade da máquina em cada período  $t$ . As restrições (1.4) garantem que só haverá produção do item  $j$  no período  $t$  se a máquina estiver preparada para produzir o item  $j$ . As restrições (1.5) – (1.7) modelam o sequenciamento dos itens. As restrições (1.5) garantem que o item fantasma  $i_0$  é o primeiro a ser produzido em cada período. Qualquer custo relacionado ao item  $i_0$  é considerado nulo e não interfere na solução do modelo (ver [16]). As restrições (1.6) nos dizem que sempre que um item  $k$  for produzido, então um item  $i$  anterior foi produzido e será produzido um item  $j$  depois de  $k$ , ou seja, estas restrições conservam o fluxo de produção. As restrições (1.7) nos asseguram que cada item é produzido no máximo uma vez por período  $t$ .

As restrições (1.8) e (1.9) são as restrições MTZ que eliminam subsequências, ou seja, não permitem que hajam subsequências desconexas no sequenciamento dos itens. As restrições (1.11) e (1.12) consideram que não há atraso nem estoque no início do horizonte de planejamento, e as restrições (1.10) são as restrições de domínio das variáveis.

Oncan *et al* [66] analisaram e compararam diversas formulações para o Problema do Caixeiro Viajante (PCV). Eles chegaram a conclusão de que a relaxação linear do modelo quando usadas as restrições MTZ são ruins, enquanto que, quando usadas as restrições MCF a relaxação linear se aproxima muito do valor ótimo do problema PCV. As restrições MCF tem o papel de eliminar subsequências desconexas e tem ordem polinomial. A ideia principal das restrições MCF é assegurar que, sempre há um caminho do produto inicial  $s$  para qualquer outro produto  $r$  no período de tempo  $t$ , obtendo assim o sequenciamento dos itens [12], [56].

Para descrever o modelo MCF1S1M, precisamos incluir uma nova variável:

$$\bullet \ m_{rijt} = \begin{cases} 1, & \text{se o fluxo da commodity } r \text{ flui do nó } i_0 \text{ para o nó } r \text{ pelo arco } (i,j); \\ 0, & \text{caso contrário,} \end{cases}$$

onde  $r \in \{1, 2, \dots, J\}$ . Uma *commodity* pode ser entendida como uma mercadoria a ser distribuída.

A variável  $m_{rijt}$  sendo 1, nos diz que se a *commodity*  $r$  é incluída na sequência de produção, então o item  $i$  precede o item  $j$  no sequenciamento, pois a *commodity*  $r$  passou pelo arco  $(i, j)$ .

**Exemplo 1.2.1.** Para entendermos a variável  $m_{rijt}$ , considere um período  $t$  onde temos que produzir 5 itens e todos eles foram produzidos e sequenciados conforme a Figura 1.1.

No início da produção temos todas as *commodities* disponíveis, e elas irão fluir pelo circuito sendo depositas nos nós (itens) conforme são requisitadas.

A *commodity* 1, por exemplo, irá sair do nó 0, fluir pelo nó 3 e será depositada no nó 1. Assim, teremos:

$$m_{103t} = m_{131t} = 1,$$

$$m_{115t} = m_{154t} = m_{142t} = m_{120t} = 0.$$

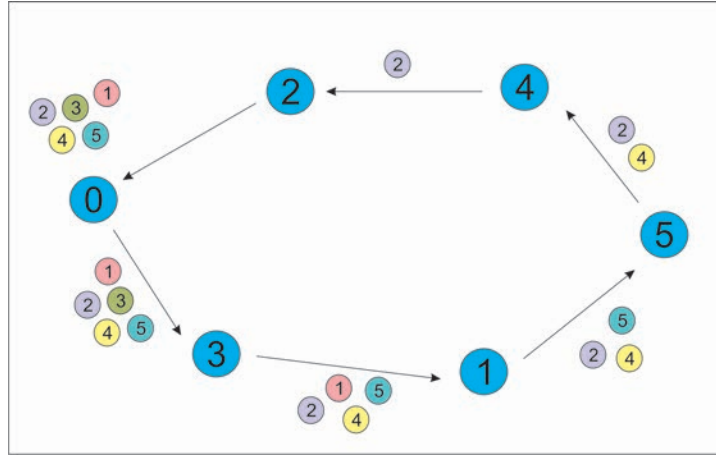


Figura 1.1: Plano de produção e sequenciamento.

O modelo MCF1S1M é descrito por (1.1) – (1.7), (1.11), (1.12) e (1.13) – (1.16).

$$\sum_{j \in J} m_{ri_0jt} - \sum_{j \in J} m_{rji_0t} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{jrt}, \quad \forall r, \forall t \quad (1.13)$$

$$\sum_{\substack{j=i_0 \\ j \neq r}}^J m_{rjrt} - \sum_{\substack{j=i_0 \\ j \neq r}}^J m_{rrjt} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{jrt}, \quad \forall r, \forall t \quad (1.14)$$

$$\sum_{\substack{i=i_0 \\ i \neq j}}^J m_{rijt} = \sum_{\substack{i=i_0 \\ i \neq j}}^J m_{rjit}, \quad \forall r, \forall t; j \neq r, \forall t, \quad (1.15)$$

$$0 \leq m_{rijt} \leq z_{ijt}, \quad \forall i, j = i_0, 1, \dots, J, \forall r, \forall t. \quad (1.16)$$

As restrições (1.13) garantem que todas as commodities saíam do nó inicial  $i_0$ , ou seja, para algum  $j$  a *commodity*  $r$  atravessará o arco  $(i_0, j)$ , e que não existirão commodities disponíveis quando incluirmos o último arco  $(k, i_0)$ , para algum item  $k$ . As restrições (1.14) garantem que a *commodity*  $r$  seja depositada em apenas um nó. As restrições (1.15) são as de conservação de fluxo, e as restrições (1.16) nos dizem que haverá travessia no arco  $(i, j)$  se houver troca de produção do item  $i$  para o item  $j$ . As restrições (1.16) também são restrições de domínio das variáveis.

Observe que nas duas formulações, MTZ1S1M e MCF1S1M, a decisão sobre o preparo ou não da máquina para a produção de um item é dada implicitamente pelas variáveis de troca através da soma:

$$w_{jt} = \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt}, \quad \forall j, \forall t. \quad (1.17)$$

De acordo com as restrições (1.7) e (1.10), a variável  $w_{jt}$  pertence ao conjunto  $\{0, 1\}$ , para cada  $j$  e  $t$ , ou seja, se houver troca de um item  $i$  para um item  $j$  no período  $t$ , então a máquina tem que ser preparada para produção do item  $j$ . Assim,  $w_{jt} = 1$ . Caso contrário, a máquina não produzirá o item  $j$  e então não há preparo, ou seja,  $w_{jt} = 0$ . Desta maneira, podemos definir a variável binária de preparo  $w$ :

$$\bullet \quad w_{jt} = \begin{cases} 1, & \text{se a máquina deve ser preparada para produzir o item } j \text{ no período} \\ & \text{de tempo } t, \\ 0, & \text{caso contrário.} \end{cases}$$

Isto nos motiva a reformular os modelos MTZ1S1M e MCF1S1M considerando esta nova variável de preparo  $w$ .

O Modelo MTZ1S1MP é dado por (1.1) – (1.3), (1.17), (1.18) e (1.5) – (1.12),

$$x_{jt} \leq C_t \frac{w_{jt}}{p_j}, \quad \forall j, \forall t, \quad (1.18)$$

sendo que as restrições (1.18) nos dizem que haverá produção do item  $j$  apenas quando a máquina estiver preparada para produzi-lo.

Da mesma forma, podemos obter o modelo MCF1S1MP, cuja formulação é dada por (1.1) – (1.3), (1.5) – (1.7), (1.13) – (1.16), (1.17), (1.18) e (1.5) – (1.12).

No Apêndice A, são apresentadas as descrições completas dos modelos discutidos nesta seção.

**Exemplo 1.2.2.** *Para exemplificar os modelos, considere que queremos resolver uma instância do problema PIDS com os dados:*

---

- $J = 5$
- $T = 2$
- $h = \begin{pmatrix} 6 & 9 & 2 & 9 & 8 \end{pmatrix}$
- $g = \begin{pmatrix} 90 & 135 & 30 & 135 & 120 \end{pmatrix}$
- $p = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \end{pmatrix}$
- $b = \begin{pmatrix} 0 & 8 & 9 & 9 & 8 \\ 9 & 0 & 8 & 8 & 8 \\ 6 & 8 & 0 & 7 & 8 \\ 6 & 6 & 7 & 0 & 6 \\ 6 & 8 & 6 & 7 & 0 \end{pmatrix}$
- $s = \begin{pmatrix} 0 & 402 & 471 & 432 & 378 \\ 446 & 0 & 396 & 422 & 419 \\ 278 & 384 & 0 & 343 & 408 \\ 311 & 324 & 348 & 0 & 304 \\ 292 & 412 & 288 & 348 & 0 \end{pmatrix}$
- $C = \begin{pmatrix} 1716 & 1315 \end{pmatrix}$
- $d = \begin{pmatrix} 40 & 42 \\ 58 & 44 \\ 47 & 56 \\ 52 & 43 \\ 59 & 52 \end{pmatrix}$

---

Figura 1.2: Dados de uma instância do problema PIDS.

Escrevendo os modelos na sintaxe do AMPL [23] e utilizando o CPLEX [39] para resolver a instância gerada com os dados apresentados na Figura 1.2, obtemos a solução ilustrada na Figura 1.3. Na solução ótima obtida, a ordem de produção no período 1 é produzir os itens na sequência 4–5–3–1–2 e no período 2, a ordem é 5–4–2. Os itens 1 e 3 não foram produzidos no período 2, mas foi produzida uma quantidade maior destes itens no período 1, e estocados para atender a demanda do período 2, como podemos ver na Figura 1.4. Os números mostrados nos retângulos representam o tamanho do lote. Por exemplo, no período 1 são produzidas 59 unidades do item 5.



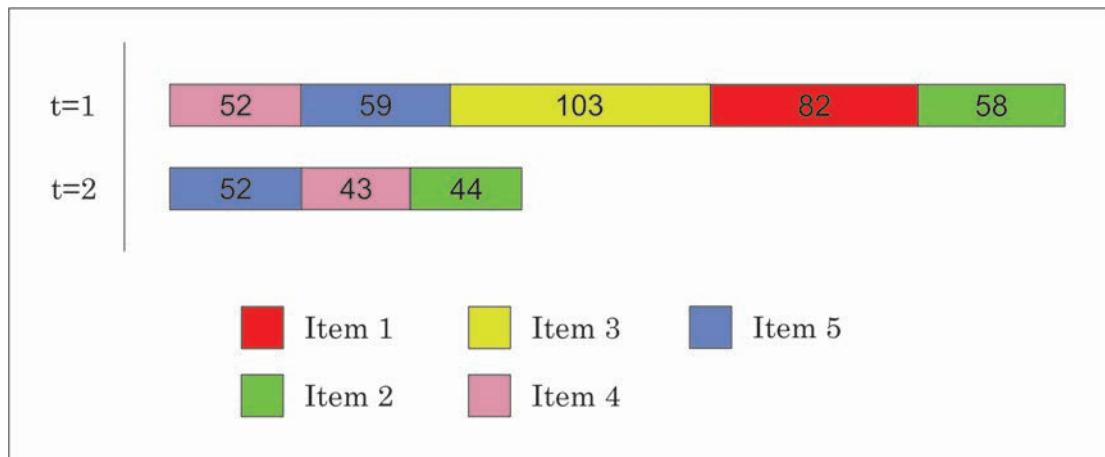


Figura 1.3: Plano de produção e sequenciamento.

$$I^+ = \begin{pmatrix} 42 & 0 \\ 0 & 0 \\ 56 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Figura 1.4: Estoque da produção.

# Capítulo 2

## Limitantes primais e duais para PI

Neste capítulo veremos algumas maneiras de derivar limites superiores e limites inferiores para um problema de otimização inteiro - PI (caso minimização). A partir dos valores destes limitantes podemos provar a otimalidade da solução, ou seja, quando os dois limites são iguais ou ainda quando a diferença entre eles é muito pequena. Podemos ainda dar um intervalo, denominado GAP, que é calculado pela diferença do limite superior pelo limite inferior, dividido pelo limite superior. Esse valor nos fornece uma medida de quão boa é a solução factível encontrada.

### 2.1 Limitantes primais

Suponhamos que o problema que queremos resolver seja da forma (PI)  $z = \min\{f(x) | x \in V\}$ . Toda solução factível  $x^* \in V$  provê um limitante primal (limitante superior, no caso de minimização), para o problema (PI). Entretanto, para muitos problemas inteiros encontrar uma solução factível não é trivial. Como exemplo, se tomarmos o problema de partição de conjuntos, saber se existe uma solução factível é um problema NP-completo [49].

Uma maneira rápida de obter uma solução factível para um PI é usando uma abordagem heurística [2], [55]. Uma heurística é um algoritmo desenvolvido a partir de um modelo cognitivo, usualmente utilizando regras baseadas na experiência dos desenvolvedores. Os métodos heurísticos se utilizam de estratégias, com o objetivo de encontrar uma boa solução, mesmo que não seja a ótima, em um tempo computacional razoável. Ginsberg [27] descreve os procedimentos heurísticos como “regras de ouro”. Podemos classificar as heurísticas em quatro grupos:

- *Métodos gulosos* - são métodos em que uma solução é construída elemento a elemento, escolhendo a cada iteração aquele que traz o maior retorno para a função objetivo;
- *Métodos de decomposição* - são métodos que consistem em dividir o problema original em vários subproblemas menores de tal forma que a resolução de todos esses subproblemas

se combinem e compõem uma solução para o problema maior;

- *Métodos de redução* - são métodos em que algumas variáveis tem seu valor fixado, de acordo com a estrutura do problema original;
- *Métodos de busca* - são métodos que partem de uma solução, que pode ser obtida por outro método, e buscam na sua vizinhança uma solução melhor.

Neste trabalho, desenvolvemos um Método Guloso para resolução do problema integrado de dimensionamento de lotes e sequenciamento da produção (ver seção 4.1).

## 2.2 Limitantes duais

Em geral, os algoritmos para resolver um problema de otimização inteiro (PI) geram duas sequências de valores. Uma sequência de limites duais  $\{\underline{z}\}$  (limites inferiores, no caso do critério de otimização ser minimização), tais que cada elemento  $\underline{z}_i$  desta sequência obedeça  $\underline{z}_i \leq z$ . E a outra de limites primais  $\{\overline{z}\}$  tais que  $\overline{z}_j \geq z$ , para todo elemento desta sequência. Desta forma, um algoritmo para resolver o problema PI gera as sequências:

$$\underline{z}_1 \leq \underline{z}_2 \leq \dots \leq \underline{z}_t \leq z \leq \overline{z}_s \leq \dots \overline{z}_2 \leq \overline{z}_1. \quad (2.1)$$

O algoritmo para quando

$$\overline{z}_s - \underline{z}_t \leq \varepsilon, \quad (2.2)$$

para algum valor de  $\varepsilon$  escolhido adequadamente [85].

A relaxação é uma das técnicas utilizadas para gerar limites duais e definida de acordo com a Definição 2.2.1 [26].

**Definição 2.2.1** (Relaxação). *Um problema (PR)  $z^R = \min\{g(x) \mid x \in W\}$  é uma relaxação de (PI)  $z = \min\{f(x) \mid x \in V\}$  se, e somente se,*

- (i)  $V \subseteq W$ , e
- (ii)  $g(x) \leq f(x), \forall x \in V$ .

A solução ótima de uma relaxação determina um limitante dual para o valor ótimo do problema original. Muitas das vezes a solução do problema relaxado é infactível para o problema original, mas pode ser usada como ponto de partida para métodos heurísticos.

Na Proposição 2.2.1, enunciamos algumas relações existentes entre os problemas originais e as relaxações associadas [62], [85].

**Proposição 2.2.1.** (i) Se (PR) é uma relaxação de (PI), então  $z^R \leq z$ ;  
(ii) Seja  $x^*$  a solução ótima do problema relaxado (PR). Se  $x^*$  é factível para o problema original (PI) e  $f(x^*) = g(x^*)$ , então  $x^*$  também é ótimo para (PI);  
(iii) Se o problema relaxado (PR) é infactível, então o modelo original (PI) também é infactível.

*Demonstração.* (i) Sendo  $x^*$  o ótimo de (PI), então  $x^* \in V \subseteq W$  e, assim,

$$z = f(x^*) \geq g(x^*). \quad (2.3)$$

Como  $x^* \in W$ , segue que  $g(x^*)$  é um limite superior para  $z^R$ , ou seja,

$$z^R \leq g(x^*). \quad (2.4)$$

Portanto, por (2.3) e (2.4), o resultado segue, ou seja,  $z^R \leq g(x^*) \leq z$ .

(ii) Como  $x^*$  é factível de (PI), ótimo para (PR) e  $f(x^*) = g(x^*)$ , segue que

$$z \leq f(x^*) = g(x^*) = z^R. \quad (2.5)$$

Mas, por (i), temos que  $z^R \leq z$ . Portanto, concluímos que  $z = z^R = f(x^*)$ .

(iii) Como (PR) é infactível, então  $W = \emptyset$ .

Por (PR) ser uma relaxação de (PI), temos que  $V \subseteq W = \emptyset$ .

Logo,  $V = \emptyset$ , ou seja, (PI) também é infactível.  $\square$

Uma das relaxações mais usadas é a relaxação linear definida na Definição 2.2.2 [85].

**Definição 2.2.2** (Relaxação Linear). *Considere o problema inteiro (PI)  $z = \min\{f(x) : x \in P \cap \mathbb{Z}^n\}$ , onde  $P = \{x \in \mathbb{R}_+^n : Ax \leq b\}$ . Então, a relaxação linear de (PI) é o problema (PL) dado por  $z_{PL} = \min\{f(x) : x \in P\}$ .*

Observe que como  $P \cap \mathbb{Z}^n \subset P$  e as funções objetivos de (PI) e (PL) são as mesmas, pela Definição 2.2.1, o problema (PL) é uma relaxação de (PI).

### 2.2.1 Planos de corte

Nesta seção, definiremos o que são inequações válidas e como elas podem ser usadas para gerar planos de corte, os quais serão utilizados no Capítulo 5.

Precisaremos das definições de poliedro e inequações válidas encontradas nas Definições 2.2.3 e 2.2.4, respectivamente [85].

**Definição 2.2.3** (Poliedro). *Um subconjunto  $P \subset \mathbb{R}^n$  descrito por um conjunto finito de restrições lineares  $P = \{x \in \mathbb{R}^n : Ax \leq b\}$  é chamado poliedro.*

**Definição 2.2.4** (Inequações Válidas). *Uma inequação da forma  $vx \leq v_0$  é uma inequação válida para o poliedro  $P \subset \mathbb{R}^n$ , se  $vx \leq v_0$  para todo  $x \in P$ .*

A ideia de utilizar inequações válidas reside no fato de que quando acrescentadas ao problema de otimização inteiro, estas diminuem a região factível da relaxação linear sem excluir soluções inteiras, aproximando-se assim do envoltório convexo. Esse problema é conhecido como problema de separação [69]. Essa ideia é utilizada no método de planos de cortes que consiste em tratar o problema de otimização linear inteiro como um problema de otimização linear contínuo e acrescentar a este problema, de forma iterativa inequações válidas (planos de cortes). Esses cortes são feitos de forma a eliminar a solução ótima da relaxação linear obtida a cada iteração, quando esta for inactível para o problema inteiro [60].

Existem muitas inequações válidas que podem ser usadas como planos de cortes [86], entre elas: Inequações GUB (*Generalized Upper Bounding*) [34], Inequações de cobertura (*Cover Cuts*) [60], Inequações de cobertura de fluxo (*Flow Cuts*) [35], [67], Inequações de cobertura de caminho de fluxo (*Flow Path*) [82], MIR (*Mixed Integer Rounding*) [59], Cliques [7], *Implied Bounds Cuts* [36], Inequações de Gomory [9], [31], [85]. Tais inequações são utilizadas por pacotes de otimização como o CPLEX 12.6 [39] que resolvem problemas gerais de otimização inteiro mistos.

## 2.2.2 Relaxação Lagrangeana

Considere o seguinte problema de otimização linear inteiro:

$$\begin{aligned} z_{PI} = \min \quad & cx \\ \text{s.a.} \quad & Ax \leq b, \\ & x \in \mathbb{Z}^n. \end{aligned} \tag{2.6}$$

Podemos subdividir o problema (2.6) da seguinte maneira:

$$\begin{aligned} z_{PI} = \min \quad & cx \\ \text{s.a.} \quad & A^1 x \leq b^1, \\ & A^2 x \leq b^2, \\ & x \in \mathbb{Z}^n, \end{aligned}$$

ou ainda,

$$\begin{aligned}
z_{PI} &= \min cx \\
\text{s.a. } &A^1x \leq b^1, \\
&x \in \mathbb{Q},
\end{aligned} \tag{2.7}$$

sendo  $\mathbb{Q} = \{x \in \mathbb{Z}^n | A^2x \leq b^2\}$  e vamos supor que as restrições  $A^1x \leq b^1$  são restrições que dificultam a resolução do problema (2.7) no sentido que sem elas, o problema seria de fácil resolução [62].

**Definição 2.2.5.** *Seja  $\pi \geq 0$  um vetor. A função  $Z(\pi, x) = cx + \pi(A^1x - b^1)$  é dita função Lagrangeana.*

**Definição 2.2.6.** *A relaxação Lagrangeana de (2.7) relativa às restrições  $A^1x \leq b^1$ , com multiplicadores de Lagrange  $\pi \geq 0$ , é o problema*

$$\begin{aligned}
Z_{RL}(\pi) &= \min Z(\pi, x) \\
\text{s.a. } &x \in \mathbb{Q}.
\end{aligned} \tag{2.8}$$

O problema (2.8) também é chamado de *problema Lagrangeano*.

**Proposição 2.2.2.** *O problema Lagrangeano (2.8) é uma relaxação do problema inteiro (2.7).*

*Demonstração.* De fato, tomando  $x$  factível para o problema (2.7), então  $x \in \mathbb{Q}$  e assim, factível para o problema (2.8).

Além disso, observe que

$$Z(\pi, x) = cx + \pi(A^1x - b^1) \leq cx,$$

para todo  $x$  factível em (2.7).

Assim, de acordo com a Definição 2.2.1, o resultado segue.  $\square$

Observe que como (2.8) é relaxação de (2.7), como demonstrado na Proposição 2.2.2, segue que  $Z_{RL}(\pi) \leq Z_{PI}, \forall \pi \geq 0$ , ou seja, para cada  $\pi \geq 0$ , temos um limitante inferior  $Z_{RL}(\pi)$  para  $Z_{PI}$ . Logo, o interessante seria encontrarmos o maior desses limitantes inferiores, ou seja, o equivalente a resolvermos o problema:

$$\begin{aligned}
Z_{DL} &= \max Z_{RL}(\pi) \\
&\pi \geq 0.
\end{aligned} \tag{2.9}$$

**Definição 2.2.7.** *O problema (2.9) é chamado dual Lagrangeano de (2.7) com relação às restrições  $A^1x \leq b^1$ .*

Observe que o problema (2.9) pode ser reescrito da seguinte maneira:

$$Z_{DL} = \max_{\pi \geq 0} \min_{x \in \mathbb{Q}} Z(\pi, x), \quad (2.10)$$

ou seja, o problema dual Lagrangeano é um problema não diferenciável irrestrito [8], [24], [62]. Desta maneira, técnicas de otimização não diferenciável serão necessárias para sua resolução.

A relaxação Lagrangeana também pode ser aplicada a problemas com restrições de igualdade. Suponhamos que o problema que queremos relaxar via multiplicadores de Lagrange seja da seguinte forma:

$$\begin{aligned} \min \quad & cx \\ \text{s.a.} \quad & A^1x = b^1, \\ & A^2x \leq b^2, \\ & x \in \mathbb{Z}^n. \end{aligned}$$

As restrições complicadoras são as restrições de igualdade  $A^1x = b^1$ . Observe que podemos reescrever esta restrição como sendo a interseção dos hiperplanos  $A^1x \leq b^1$  e  $-A^1x \leq -b^1$ . Agora, tomando os multiplicadores  $\mu \geq 0$  e  $\nu \geq 0$ , e dualizando estas duas restrições, temos

$$\begin{aligned} \min \quad & cx + \mu(A^1x - b^1) + \nu(-A^1x + b^1) \\ \text{s.a.} \quad & A^2x \leq b^2, \\ & x \in \mathbb{Z}^n, \end{aligned}$$

que pode ser reescrito convenientemente como

$$\begin{aligned} \min \quad & cx + \pi(A^1x - b^1) \\ \text{s.a.} \quad & A^2x \leq b^2, \\ & x \in \mathbb{Z}^n, \end{aligned}$$

sendo  $\pi = \mu - \nu$ . Neste caso,  $\pi$  é livre de sinal, pois é possível que  $\mu < \nu$ .

**Definição 2.2.8.** 1. Fixado um valor de  $\pi$ , denotaremos a solução do problema (2.8) em relação a este  $\pi$  fixado, por  $x(\pi)$ .

2. Dado um problema de otimização  $P$ , denotaremos por  $v(P)$  o valor ótimo de  $P$ .

**Proposição 2.2.3.** *Seja  $\pi \geq 0$ . Se  $x(\pi)$  satisfaz as seguintes condições*

- (i) *é ótimo para (2.8),*
- (ii) *é factível para (2.7), ou seja,  $A^1x(\pi) \leq b^1$ , e*
- (iii) *satisfaz as folgas complementares, ou seja,  $\pi(A^1x(\pi) - b^1) = 0$ ,*  
*então  $x(\pi)$  é ótimo para (2.7) e  $Z_{DL} = Z_{PI}$ .*

*Demonstração.* Por (i), temos que  $x(\pi)$  é ótimo para o problema (2.8), ou seja,

$$Z_{DL} \geq Z_{RL}(\pi) = cx(\pi) + \pi(A^1x(\pi) - b^1);$$

Por (iii), temos que  $\pi(A^1x(\pi) - b^1) = 0$ , ou seja,

$$cx(\pi) + \pi(A^1x(\pi) - b^1) = cx(\pi);$$

Ainda, por (ii), sendo  $x(\pi)$  factível para o problema (2.7), segue que

$$cx(\pi) \geq Z_{PI}.$$

Logo, concluímos que

$$Z_{DL} \geq Z_{RL}(\pi) = cx(\pi) \geq Z_{PI},$$

ou seja,  $Z_{DL} \geq Z_{PI}$ . Como sabemos que  $Z_{PI} \geq Z_{DL}$ , o resultado segue, ou seja,

$$Z_{DL} = Z_{PI}.$$

□

Observe que podemos ter  $x(\pi)$  ótimo para o problema (2.7) sem satisfazer as folgas complementares, já que a condição é apenas suficiente, e não necessária. Outro detalhe importante é que se as restrições a serem dualizadas são restrições de igualdade, então não necessitamos verificar as folgas complementares, uma vez que estas são automaticamente satisfeitas.

**Exemplo 2.2.1.** *Para exemplificar alguns conceitos mencionados neste texto, considere o seguinte problema de otimização inteiro [2]:*

$$\begin{aligned} Z_{PI} = \min & -7x_1 - 2x_2 \\ & -x_1 + 2x_2 \leq 4 \\ & 5x_1 + x_2 \leq 20 \\ & -2x_1 - 2x_2 \leq -7 \\ & -x_1 \leq -2 \\ & x_2 \leq 4 \\ & x \in \mathbb{Z}_+^2. \end{aligned} \tag{2.11}$$



Denominemos de  $V$  o conjunto factível do problema (2.11), e dualizemos a restrição  $-x_1 + 2x_2 \leq 4$  com o multiplicador de Lagrange  $\pi \geq 0$ . Desta forma, o subproblema Lagrangeano é dado pelo problema (2.12):

$$\begin{aligned}
 Z_{RL}(\pi) = \min & -7x_1 - 2x_2 + \pi(-x_1 + 2x_2 - 4) \\
 & 5x_1 + x_2 \leq 20 \\
 & -2x_1 - 2x_2 \leq -7 \\
 & -x_1 \leq -2 \\
 & x_2 \leq 4 \\
 & x \in \mathbb{Z}_+^2.
 \end{aligned} \tag{2.12}$$

Da mesma forma, denominemos por  $Q$  o conjunto factível do problema (2.12). Podemos ver na Figura 2.1 os conjuntos factíveis  $Q$ ,  $\text{conv}(Q)$  e  $\text{conv}(V)$ .

Observe que  $Q = \{x^1, x^2, \dots, x^8\} = \{(2, 2), (2, 3), (2, 4), (3, 1), (3, 2), (3, 3), (3, 4), (4, 0)\}$ .

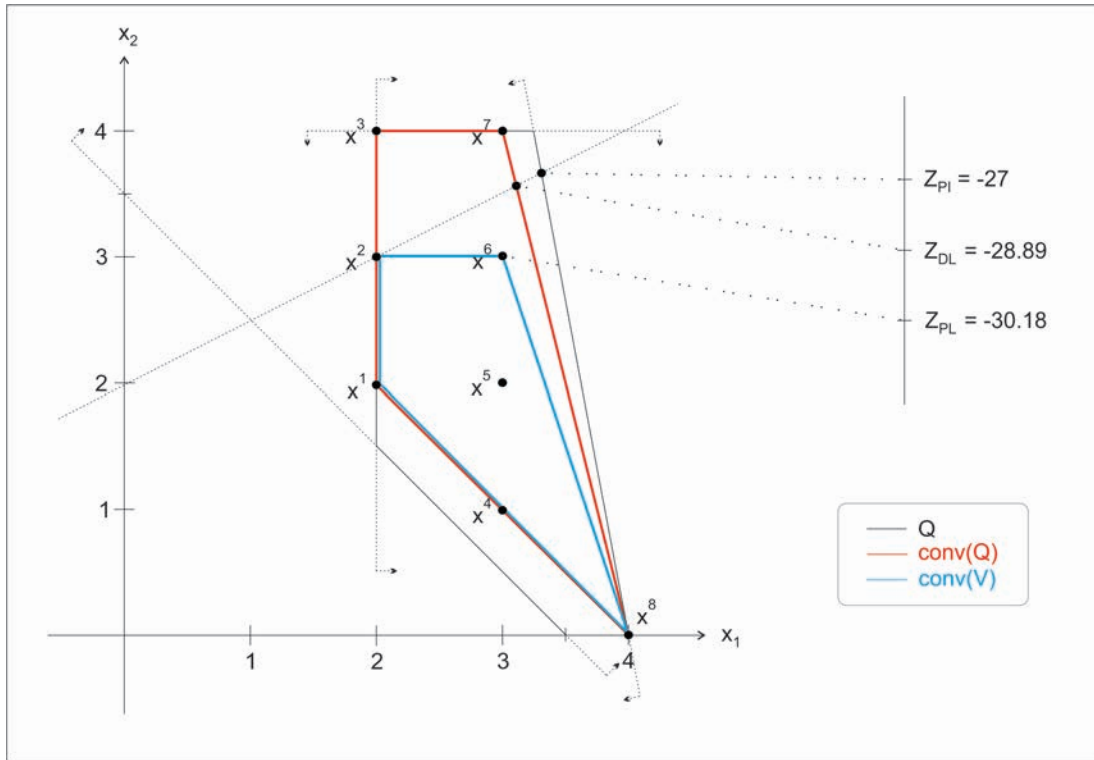


Figura 2.1: Conjuntos factíveis  $Q$ ,  $\text{conv}(Q)$  e  $\text{conv}(V)$ , e soluções do problema inteiro, do problema dual Lagrangeano e da relaxação linear.

A sigla (PL) usada na Figura 2.1 refere-se a relaxação linear do problema (PI).

Podemos tentar resolver o problema 2.12 usando duas abordagens:

1ª abordagem: Tomar a função  $Z(\pi, x)$  como uma função afim de  $x$  para  $\pi$  fixo.

Para  $\pi$  fixo, podemos obter o valor de  $Z_{RL}(\pi)$  resolvendo o seguinte problema de programação linear:

$$Z_{RL}(\pi) = \min_{x \in \text{conv}(Q)} Z(\pi, x)$$

onde  $\text{conv}(Q) = \{x \in \mathbb{R}_+^2 \mid -x_1 \leq -2, x_2 \leq 4, -x_1 - x_2 \leq -4, 4x_1 + x_2 \leq 16\}$ , que, para este exemplo, pode facilmente ser obtido usando a Figura 2.1.

Assim, o subproblema Lagrangeano se torna:

$$\begin{aligned} Z_{LR}(\pi) = \min & -7x_1 - 2x_2 + \pi(-x_1 + 2x_2 - 4) \\ & -x_1 - x_2 \leq -4 \\ & 4x_1 + x_2 \leq 16 \\ & -x_1 \leq -2 \\ & x_2 \leq 4 \\ & x \in \mathbb{R}_+^2. \end{aligned} \tag{2.13}$$

Para resolvermos o problema (2.13), como  $\pi$  é fixo, podemos usar uma heurística simples de busca, ir variando  $\pi$  até encontrarmos a solução ótima, já que a função  $Z_{LR}(\pi)$  é uma função linear por partes côncava.

Por exemplo,

$$\begin{aligned} \pi = 0 & \Rightarrow Z_{RL}(0) = \min \{-7x_1 - 2x_2 \mid x \in \text{conv}(Q)\} = Z(0, x^7) = -29 \\ \pi = 0.1 & \Rightarrow Z_{RL}(0.1) = \min \{-7.1x_1 - 1.8x_2 - 0.4 \mid x \in \text{conv}(Q)\} = Z(0.1, x^7) = -28.9 \\ \pi = 0.5 & \Rightarrow Z_{RL}(0.5) = \min \{-7.5x_1 - x_2 - 2 \mid x \in \text{conv}(Q)\} = Z(0.5, x^8) = -32 \\ \pi = 0.75 & \Rightarrow Z_{RL}(0.75) = \min \{-7.75x_1 - 0.5x_2 - 3 \mid x \in \text{conv}(Q)\} = Z(0.75, x^8) = -34 \\ \pi = 1 & \Rightarrow Z_{RL}(1) = \min \{-8x_1 - 4 \mid x \in \text{conv}(Q)\} = Z(1, x^8) = -36 \\ \pi = 1.5 & \Rightarrow Z_{RL}(1.5) = \min \{-8.5x_1 + x_2 - 6 \mid x \in \text{conv}(Q)\} = Z(1.5, x^8) = -40. \end{aligned}$$

Observe que  $\nabla Z_{RL}(\pi) = (-7 - \pi, -2 + 2\pi)$ , e assim para qualquer  $\pi > 1$ , a solução do problema é o ponto  $x^8$ , pois o vetor gradiente se encontra no 2º quadrante e o ponto  $x^8$  é o ponto mais distante com relação as retas perpendiculares ao vetor gradiente. Assim, fixemos  $x^8$  e verifiquemos o que acontece quando  $\pi \rightarrow +\infty$ . Considerando  $x^8$  fixo, temos  $Z_{RL}(\pi) = -4\pi - 32$  e assim,  $Z_{RL}(\pi) \rightarrow -\infty$ . Logo, o valor de  $\pi$  ótimo pertence ao intervalo  $[0, 1)$ , mais precisamente, próximo do valor 0.1, como podemos ver quando variamos o valor de  $\pi$ .

Fazendo mais algumas comparações entre os valores de  $\pi$ , chegamos que a solução ótima de  $\pi$  é 0, 11 e de  $x$  é  $x^7$ , cujo valor ótimo é  $-28,89$ .

2ª abordagem: Tomar a função  $Z(\pi, x)$  como uma função afim de  $\pi$  para  $x$  fixo. Para cada  $x^i$ ,  $i = 1, 2, \dots, 8$ , temos uma função  $Z_{RL}(\pi)$  :

$$\begin{aligned} Z(\pi, x^1) &= -18 - 2\pi \\ Z(\pi, x^2) &= -20 \\ Z(\pi, x^3) &= -22 + 2\pi \\ Z(\pi, x^4) &= -23 - 5\pi \\ Z(\pi, x^5) &= -25 - 3\pi \\ Z(\pi, x^6) &= -27 - \pi \\ Z(\pi, x^7) &= -29 + \pi \\ Z(\pi, x^8) &= -28 - 8\pi. \end{aligned}$$

Na Figura 2.1, podemos ver os gráficos das funções  $Z(\pi, x^i)$ ,  $i = 1, 2, \dots, 8$ , e em vermelho, a função  $Z_{RL}(\pi)$ .

Temos que

$$Z_{RL}(\pi) = \min_{\pi \geq 0} Z_{RL}(\pi, x) = \min_{\pi \geq 0} Z(\pi, x^i) \\ x \in Q \quad i = 1, 2, \dots, 8,$$

e, assim,

$$\begin{aligned} Z_{RL}(\pi) &= \min_{\pi \geq 0} \{-18 - 2\pi, -20, -22 + 2\pi, -23 - 5\pi, -25 - 3\pi, -27 - \pi, -29 + \pi, -28 - 8\pi\} \\ &= \begin{cases} -29 + \pi, & \text{se } 0 \leq \pi \leq 1/9 \\ -28 - 8\pi, & \text{se } \pi \geq 1/9. \end{cases} \end{aligned}$$

Claramente vemos que  $\pi = 1/9$  maximiza a função  $Z_{RL}(\pi)$ , e assim,  $Z_{DL} = Z_{RL}(1/9) = -28,89$ .

Uma outra maneira de enxergarmos essa solução é a seguinte:

Sabemos que

$$Z_{DL} = \max_{\pi \geq 0} Z_{RL}(\pi) = \max_{\pi \geq 0} \min_{i=1,2,\dots,8} Z(\pi, x^i),$$

que é equivalente a

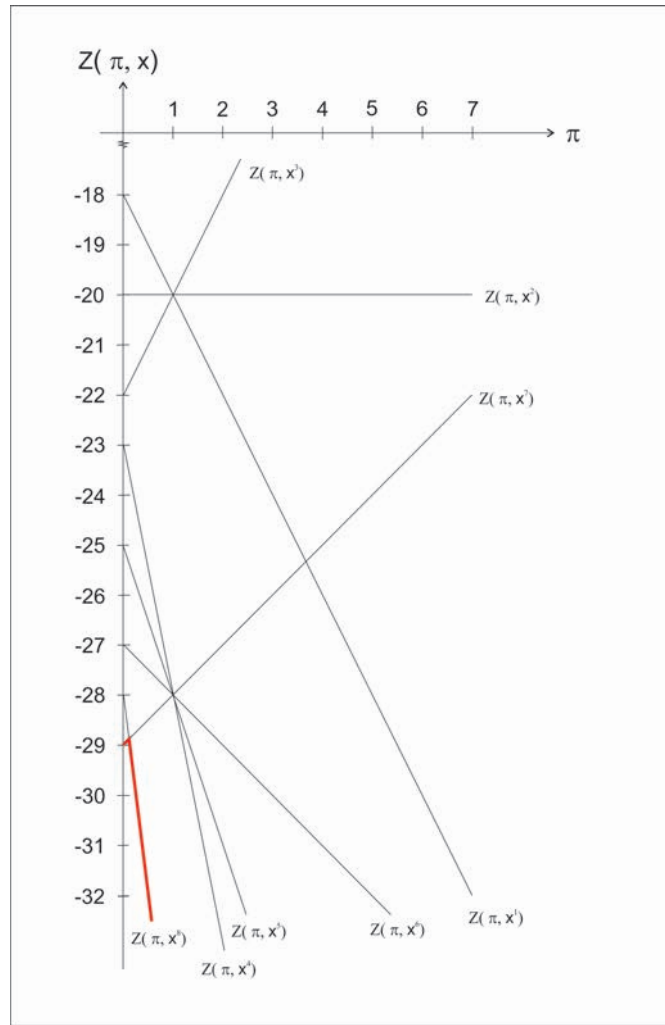


Figura 2.2: Gráficos das funções  $Z(\pi, x^i)$ ,  $i = 1, 2, \dots, 8$ .

$$Z_{DL} = \max_{\pi \geq 0} w$$

$$w \leq Z(\pi, x^i), \quad i = 1, 2, \dots, 8.$$

*Ou seja, necessitamos apenas resolver o problema de otimização linear:*

$$\begin{aligned}
Z_{DL} = \max \quad & w \\
& w \leq -18 - 2\pi \\
& w \leq -20 \\
& w \leq -22 + 2\pi \\
& w \leq -23 - 5\pi \\
& w \leq -25 - 3\pi \\
& w \leq -27 - \pi \\
& w \leq -29 + \pi \\
& w \leq -28 - 8\pi \\
& w \geq 0, \pi \geq 0.
\end{aligned}$$

A solução deste problema é  $\pi = 1/9$  e  $w = -28.89$ .

Observe que em todas as abordagens usadas para resolver o problema do Exemplo 2.2.1, foi preciso conhecer todos os pontos do conjunto  $Q$ , tornando essas práticas inviáveis para problemas de grande porte. No Teorema 2.2.1, apresentamos um resultado proposto por Geoffrion [26], que embasa toda a teoria de relaxação Lagrangeana.

**Teorema 2.2.1.** *O dual Lagrangeano (2.9) é equivalente ao problema*

$$\begin{aligned}
\min \quad & cx \\
s.a. \quad & A^1x \leq b^1, \\
& x \in \text{conv}(Q),
\end{aligned} \tag{2.14}$$

no sentido de que o valor ótimo do problema (2.14) é igual ao valor ótimo do problema (2.9). O problema (2.14) é dito relaxação primal (RP).

*Demonstração.* Suponhamos que  $Q$  seja não vazio e limitado.

Observe que:

$$\begin{aligned}
Z_{RL}(\pi) &= \min_{x \in Q} cx + \pi(A^1x - b^1) = \min_{x \in \text{conv}(Q)} cx + \pi(A^1x - b^1)
\end{aligned}$$

Sabemos que  $Z_{DL} = \max_{\pi \geq 0} Z_{RL}(\pi)$ .

Logo,

$$Z_{DL} = \max_{\pi \geq 0} \min_{x \in \text{conv}(\mathbb{Q})} cx + \pi(A^1x - b^1)$$

Como  $\mathbb{Q} \neq \emptyset$ , podemos reescrever  $\mathbb{Q}$  em função de seus pontos extremos como sendo

$$\mathbb{Q} = \{x^k \in \mathbb{Z}_+^n \mid k \in \mathcal{K}\},$$

onde  $\mathcal{K}$  é o conjunto de índices dos pontos extremos, ou seja, a cardinalidade de  $\mathcal{K}$  é a quantidade de pontos extremos de  $\mathbb{Q}$  uma vez que tomamos  $\mathbb{Q}$  limitado.

Desta forma, teremos que

$$\begin{aligned} \min \quad & cx + \pi(A^1x - b^1) = cx^k + \pi(A^1x^k - b^1), \\ \text{s.a.} \quad & x \in \text{conv}(\mathbb{Q}). \end{aligned}$$

para algum  $k \in \mathcal{K}$ , e

$$Z_{DL} = \max_{\pi \geq 0} \left[ \min_{k \in \mathcal{K}} cx^k + \pi(A^1x^k - b^1) \right].$$

Equivalentemente, temos que

$$\begin{aligned} Z_{LD} = \max \quad & \mu \\ \text{s.a.} \quad & \mu \leq cx^k + \pi(A^1x^k - b^1), k \in \mathcal{K} \\ & \pi \geq 0, \mu \text{ livre}, \end{aligned}$$

que ainda equivale a

$$\begin{aligned} Z_{LD} = \max \quad & \mu \\ \text{s.a.} \quad & \mu - \pi(A^1x^k - b^1) \leq cx^k, k \in \mathcal{K} \\ & \pi \geq 0, \mu \text{ livre}. \end{aligned} \tag{2.15}$$

Dualizando o problema (2.15) nas variáveis duais  $\alpha^k$ ,  $k \in \mathcal{K}$ , temos

$$\begin{aligned} Z_{LD} = \min \quad & \sum_{k \in \mathcal{K}} (cx^k) \alpha^k \\ \text{s.a.} \quad & - \sum_{k \in \mathcal{K}} (A^1x^k - b^1) \alpha^k \geq 0 \\ & \sum_{k \in \mathcal{K}} \alpha^k = 1 \\ & \alpha^k \geq 0, k \in \mathcal{K}, \end{aligned}$$

que equivale a

$$\begin{aligned} Z_{LD} &= \min \sum_{k \in \mathcal{K}} (cx^k) \alpha^k \\ \text{s.a. } &\sum_{k \in \mathcal{K}} (A^1 x^k) \alpha^k \leq b^1 \\ &\sum_{k \in \mathcal{K}} \alpha^k = 1 \\ &\alpha^k \geq 0, \quad k \in \mathcal{K}, \end{aligned}$$

que pode ser reescrito como:

$$\begin{aligned} Z_{LD} &= \min cx \\ &A^1 x \leq b^1 \\ &x \in \text{conv}(\mathbb{Q}), \end{aligned}$$

$$\text{uma vez que } \text{conv}(\mathbb{Q}) = \left\{ x \in \mathbb{R}_+^n \mid x = \sum_{k \in \mathcal{K}} \alpha^k x^k, \sum_{k \in \mathcal{K}} \alpha^k = 1, \alpha^k \geq 0, \quad k \in \mathcal{K} \right\}.$$

Portanto, o resultado segue. □

Guignard [62] observa que se a matriz de restrições não for formada por números racionais, então o Teorema 2.2.1 pode não ser verdadeiro.

**Definição 2.2.9.** *O problema (2.9) tem a propriedade de integralidade se  $\{x \mid A^2 x \leq b^2\} = \text{conv}\{x \mid A^2 x \leq b^2\}$ .*

Na Figura 2.3, podemos ver uma interpretação geométrica da relaxação Lagrangeana. Observando a Figura 2.3, se o problema (2.9) tem a propriedade de integralidade, então a relaxação primal não produz melhor limitante que a relaxação linear. Esta observação nos motiva ao seguinte corolário:

**Corolário 2.2.1.** *1. Se o problema (2.9) tem a propriedade de integralidade, então  $v(PL) = v(RP) = v(RL) \leq v(PI)$ .*

*2. Se o problema (2.9) não tem a propriedade de integralidade, ou seja,  $\{x \mid A^2 \leq b^2\} \supset \text{conv}\{x \mid A^2 \leq b^2\}$ , então  $v(PL) < v(RP) = v(RL) \leq v(PI)$ .*

Este resultado nos diz que o limitante obtido através da relaxação Lagrangeana é melhor ou igual ao obtido pela relaxação linear.

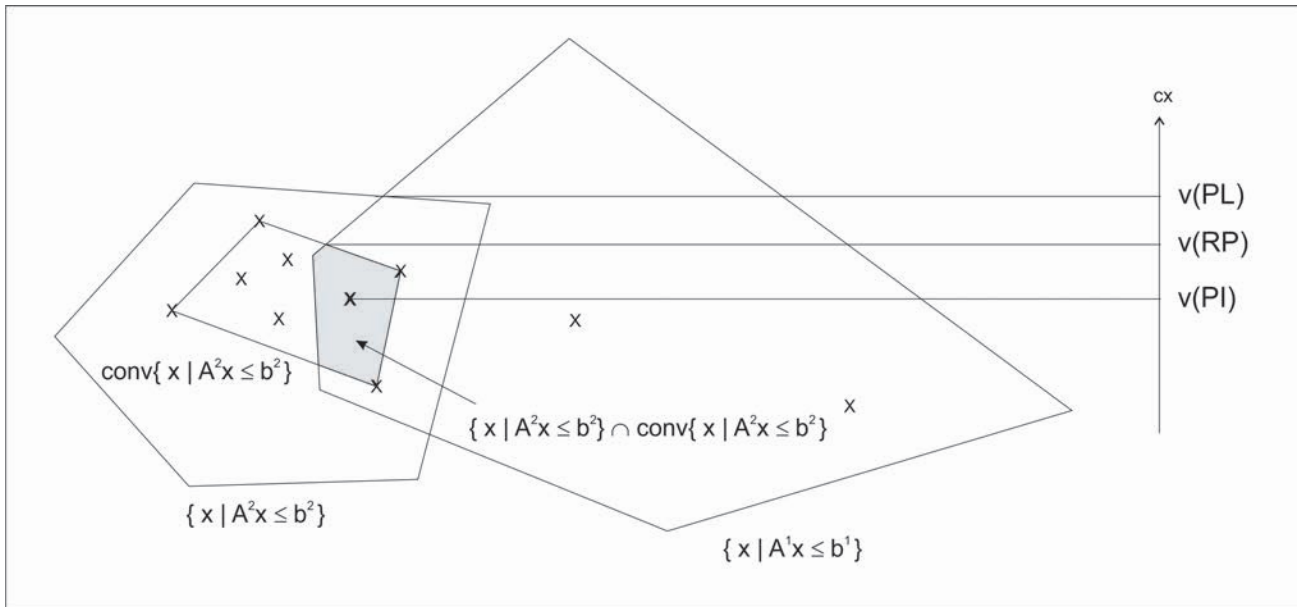


Figura 2.3: Interpretação geométrica da relaxação Lagrangeana [62].

### 2.2.3 Relaxação Surrogate

A relaxação Surrogate tem sido menos aplicada que a relaxação Lagrangeana. Ela apareceu pela primeira vez na literatura nos trabalhos de Glover [28] e [29]. Esta técnica consiste basicamente em substituir um conjunto de restrições do problema original (PI) por uma combinação linear destas restrições, dando origem a chamada restrição Surrogate.

Em 1975, Glover [30] apresentou a teoria da dualidade Surrogate, resumindo o trabalho feito anteriormente por Greenberg e Pierskalla [33]. Em 1979 Karwan e Rardin [46] desenvolveram um estudo relacionando o dual Lagrangeano e o dual Surrogate, para o caso de problemas inteiros.

Lorena, Freville e Plateau [51], Lorena e Lopes [52], e Lorena e Narciso [53] utilizaram a relaxação Surrogate contínua, ou seja, relaxa-se também as restrições de integralidade, aos problemas da mochila 0 – 1, cobertura de conjuntos e ao problema generalizado de atribuição, respectivamente. Em todos eles pode-se observar um melhor desempenho com relação ao uso da relaxação Lagrangeana.

A formulação do problema Surrogate é bastante simples. Considere o problema de otimização inteiro (PI) definido na Subseção 2.2.2:



$$\begin{aligned}
z_{PI} = \min \quad & cx \\
\text{s.a.} \quad & A^1 x \leq b^1, \\
& A^2 x \leq b^2, \\
& x \in \mathbb{Z}^n.
\end{aligned} \tag{2.16}$$

Considere um vetor de multiplicadores  $\mu = (\mu_1, \mu_2, \dots, \mu_k)$ , onde  $\mu_i \geq 0$ ,  $i = 1, 2, \dots, k$ . Uma relaxação Surrogate RS( $\mu$ ) do problema (2.16) pode ser formulada como

$$\begin{aligned}
z_{RS}(\mu) = \min \quad & cx \\
\text{s.a.} \quad & \mu(A^1 x - b^1) \leq 0, \\
& A^2 x \leq b^2, \\
& x \in \mathbb{Z}^n.
\end{aligned} \tag{2.17}$$

Desta forma, as  $k$  restrições escritas como  $A^1 x \leq b^1$ , são substituídas por apenas uma restrição  $\mu(A^1 x - b^1) \leq 0$ . Quando também relaxamos as restrições de integralidade, pondo  $x \in \mathbb{R}^n$ , obtemos a relaxação Surrogate contínua.

Galvão [25] diz que problemas resultantes de relaxação Surrogate são mais difíceis de resolver que os da relaxação Lagrangeana, devido a natureza de mochila que os problemas relaxados obtém. Entretanto, o dual Surrogate (DS), definido como

$$\begin{aligned}
Z_{DS} = \max \quad & Z_{RS}(\mu) \\
& \mu \geq 0,
\end{aligned} \tag{2.18}$$

fornece limites melhores ou iguais ao dual Lagrangeano, como veremos nos Teoremas 2.2.2 e 2.2.3.

**Teorema 2.2.2.** *Considere os problemas PI, RS( $\mu$ ) e DS definidos por 2.16, 2.17 e 2.18, respectivamente. Então,  $v(RS(\mu)) \leq v(PI)$ , e conseqüentemente,  $v(DS) \leq v(PI)$ .*

*Demonstração.* Claramente, para cada  $\mu \geq 0$ , temos que o problema Surrogate RS( $\mu$ ) é uma relaxação do problema (PI), conforme o item (i) da Proposição 2.2.1, pois toda solução factível de (PI) também é factível para RS( $\mu$ ).

Assim, ainda pela Proposição 2.2.1,

$$v(RS(\mu)) \leq v(PI), \quad \forall \mu \geq 0.$$

Como a desigualdade é válida para todo  $\mu \geq 0$ , segue que

$$\max_{\mu \geq 0} v(RS(\mu)) \leq v(PI).$$

Mas,  $\max_{\mu \geq 0} v(RS(\mu)) = v(DS)$ . Logo,

$$v(DS) \leq v(PI).$$

□

**Teorema 2.2.3.** *Considere os problemas (PI), (DL( $\pi$ )) e (DS( $\mu$ )) como definidos anteriormente. Então,  $v(DL(\pi)) \leq v(DS(\mu))$ . Além disso, se  $v(DL(\pi)) = v(DS(\mu))$ , então para qualquer  $\pi$  que resolva (DL( $\pi^*$ )), deve existir  $x^*$  tal que  $\pi^*(A^2x^* - b^2) = 0$ .*

*Demonstração.* Ver o livro de Parker e Rardin [68].

□

## 2.2.4 Relaxação Lagrangeana/Surrogate

A relaxação Lagrangeana/Surrogate foi proposta por Narciso [64]. Seu objetivo é combinar as relaxações Lagrangeana e Surrogate para conseguir melhores tempos computacionais na aplicação de heurísticas subgradientes. Esta relaxação consiste em relaxar via multiplicadores Surrogate um conjunto de restrições, e depois relaxar via multiplicadores de Lagrange esta única restrição. O multiplicador de Lagrange é unidimensional.

Considere o problema de otimização inteiro (PI) definido na Subseção 2.2.2:

$$\begin{aligned} z_{PI} = \min \quad & cx \\ \text{s.a.} \quad & A^1x \leq b^1, \\ & A^2x \leq b^2, \\ & x \in \mathbb{Z}^n. \end{aligned} \tag{2.19}$$

A relaxação Surrogate RS( $\mu$ ) do problema (2.19) (ver Subseção 2.2.3):

$$\begin{aligned} z_{RS}(\mu) = \min \quad & cx \\ \text{s.a.} \quad & \mu(A^1x - b^1) \leq 0, \\ & A^2x \leq b^2, \\ & x \in \mathbb{Z}^n. \end{aligned} \tag{2.20}$$

Relaxando a restrição Surrogate de RS( $\mu$ ) via multiplicadores de Lagrange, obtemos o sub-problema Lagrangeano/Surrogate (RLS( $t, \mu$ )):

$$\begin{aligned}
z_{RLS}(t, \mu) = \min \quad & cx + t\mu(A^1x - b^1) \\
\text{s.a.} \quad & A^2x \leq b^2, \\
& x \in \mathbb{Z}^n.
\end{aligned} \tag{2.21}$$

Da mesma forma que as outras relaxações, temos o dual Lagrangeano Surrogate (DLS):

$$\begin{aligned}
Z_{DLS} = \max \quad & Z_{RLS}(t, \mu) \\
& t \geq 0, \mu \geq 0,
\end{aligned} \tag{2.22}$$

Claramente, se substituirmos  $u = t\pi$  obtemos o dual Lagrangeano usual.

Fixando  $\mu^* \geq 0$ , e enxergando a função Lagrangeana/Surrogate em função do parâmetro  $t$ , ou seja,  $RLS(t, \mu^*) = (\mu^* A^1x - \mu^* b^1)t + cx$ , uma visualização do seu comportamento pode ser visto na Figura 2.4.

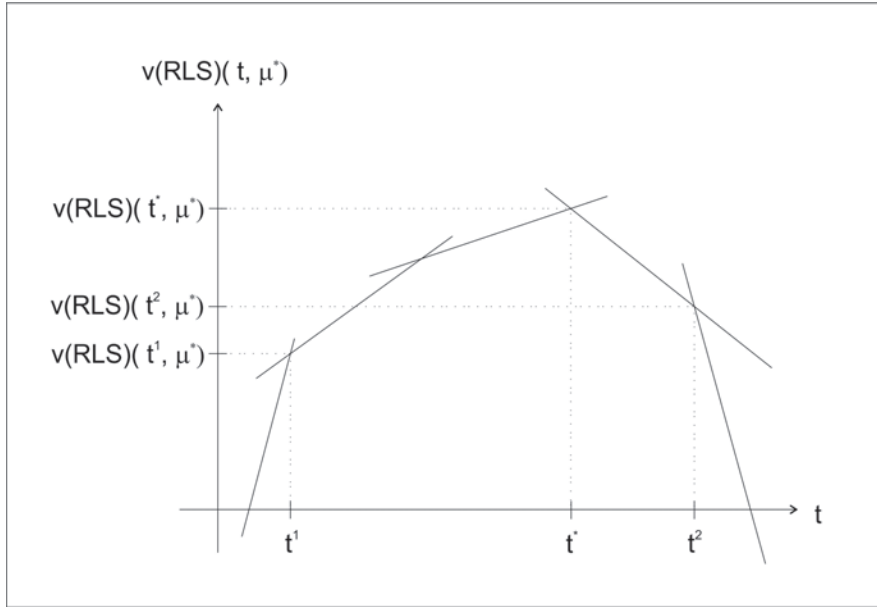


Figura 2.4: Comportamento da função Lagrangeano/Surrogate em função de  $t$  [64].

O objetivo então é encontrar  $t^*$  que maximize o valor de  $Z_{RLS}(t, \mu^*)$ . Existem vários métodos que encontram o valor de  $t$  que minimiza o valor da função  $v(RLS(t, \mu))$ , como dito por Narciso [64]. Entretanto, estes métodos são caros computacionalmente. O método usado neste trabalho é apresentado na subseção 4.3.1. O método consiste em fixado um valor para  $\mu = \mu^*$  e dado um valor inicial para  $t$ , buscar na vizinhança de  $t$  melhores limitantes para o problema (2.21).

# Capítulo 3

## Solução do problema dual Lagrangeano

Neste capítulo, introduziremos métodos para resolver o problema dual Lagrangeano, e apresentamos alguns conceitos para melhor entendimento dos métodos.

Como vimos no Exemplo 2.2.1, o problema dual Lagrangeano é linear por partes, não diferenciável e côncavo [41], [10]. Assim, é necessário um método de otimização não linear para resolver o problema dual Lagrangeano. Solodov [42] argumenta que problemas de otimização não diferenciáveis necessitam da utilização de técnicas de solução próprias, diferentes daquelas para o caso diferenciável. Para a minimização irrestrita sobre uma função convexa  $f$ , muitos métodos estão disponíveis: subgradientes, planos de corte, métodos de feixes, entre outros. Esses métodos tem suas vantagens e desvantagens, e as respectivas eficiências dependerão da natureza do problema a ser resolvido.

De uma maneira geral, os algoritmos de otimização não diferenciável, fundamentam-se na geração de iterados  $x^k$  através da busca de possíveis direções de descida  $d^k$  e tamanhos de passo  $s_k$ , e consequente atualização na forma  $x^{k+1} = x^k + s_k d^k$ .

### 3.1 Conceitos básicos

Nesta seção, faremos uma introdução a alguns conceitos da teoria de Análise Matemática e Otimização Não Linear necessária para o entendimento dos métodos apresentados neste Capítulo. Mais detalhes podem ser obtidos em [41], [42], [50].

De maneira informal, podemos definir uma sequência como uma sucessão interminável de termos. O interesse em sequências vem particularmente em saber como ela evolui, ou seja, como ela se comporta conforme seus termos vão sendo gerados. Formalmente, a definição de sequência é dada pela Definição 3.1.1:

**Definição 3.1.1** (Sequência). *Uma sequência é uma função  $x : \mathbb{N} \rightarrow \mathbb{R}$ . O valor  $x(n)$  será representado por  $x_n$  e chamado termo de ordem  $n$  da sequência.*

Intuitivamente, dizer que o número real  $a$  é limite de uma sequência  $(x_n)$  significa afirmar que quanto maior o valor de  $n$ , mais próximo a sequência está de  $a$ . Isto nos leva a seguinte definição:

**Definição 3.1.2** (Limite de Sequência). *Dizemos que o número real  $a$  é limite de uma sequência  $x_n$ , se para cada valor arbitrário de  $\varepsilon > 0$ , for possível obter um número  $n_0 \in \mathbb{N}$  tal que  $|x_n - a| < \varepsilon$ , para todo  $n > n_0$ . Denotaremos  $\lim x_n = a$ . Em símbolos matemáticos,*

$$\lim x_n = a \Leftrightarrow \forall \varepsilon > 0, \exists n_0 \in \mathbb{N}; n > n_0 \Rightarrow |x_n - a| < \varepsilon.$$

Uma sequência  $(x_n)$  limitada (ou seja,  $a \leq x_n \leq b$ ,  $\forall n$ ) pode não ter limite. Para resolver este problema, generaliza-se o limite, como vemos na Definição 3.1.3:

**Definição 3.1.3** (Limites Superior e Inferior). *Seja  $(x_n)$  uma sequência limitada em  $\mathbb{R}$ . O  $\limsup$  de  $(x_n)$  é definido por:*

$$\limsup x_n := \lim_{k \rightarrow \infty} \sup \{x_n \mid n \geq k\},$$

e o  $\liminf$  de  $(x_n)$  é definido por:

$$\liminf x_n := \lim_{k \rightarrow \infty} \inf \{x_n \mid n \geq k\}.$$

**Definição 3.1.4** (Séries). *Dada uma sequência  $(x_n)$  de números reais, chama-se série numérica a soma infinita:*

$$x_1 + x_2 + \dots, x_n + \dots = \sum_{k=1}^{\infty} x_n.$$

Para calcularmos, se possível, o valor da série  $\sum_{k=1}^{\infty} x_n$ , consideremos a seguinte sequência:

$$\begin{aligned} S_1 &= x_1; \\ S_2 &= x_1 + x_2; \\ S_3 &= x_1 + x_2 + x_3; \\ &\vdots \\ S_n &= x_1 + x_2 + \dots + x_n; \\ &\vdots \end{aligned}$$

em que cada  $S_i, \forall i$  é chamado de soma parcial de  $(x_n)$ . Com esta observação, a Definição 3.1.5 nos ajuda a verificar se a série tem ou não valor finito.

**Definição 3.1.5** (Convergência de Série). *Se existe um número real  $S$  tal que  $\lim_{n \rightarrow \infty} S_n = S$ , então dizemos que a série  $\sum_{k=1}^{\infty} x_k$  é convergente e  $\sum_{k=1}^{\infty} x_k = S$ .*

Uma condição necessária para a convergência de uma série é que seu termo geral tenda para zero [50], ou seja,

**Teorema 3.1.1.** *Se  $\sum_{k=1}^{\infty} x_k$  é uma série convergente, então  $\lim_{n \rightarrow \infty} x_n = 0$ .*

*Demonstração.* Como a série  $\sum_{k=1}^{\infty} x_k$  é convergente, pela Definição 3.1.5 temos que existe um valor  $S$  real tal que  $\lim_{n \rightarrow \infty} S_n = S$ .

Claramente,  $\lim_{n \rightarrow \infty} S_{n-1} = S$ .

Observe que  $S_n - S_{n-1} = x_1 + x_2 + \dots + x_n - x_1 - x_2 - \dots - x_{n-1} = x_n$ .

Desta forma,

$$0 = S - S = \lim_{n \rightarrow \infty} S_n - \lim_{n \rightarrow \infty} S_{n-1} = \lim_{n \rightarrow \infty} (S_n - S_{n-1}) = \lim_{n \rightarrow \infty} x_n,$$

ou seja,  $\lim_{n \rightarrow \infty} x_n = 0$ . □

Convexidade é uma noção muito importante na teoria de otimização, pois com hipóteses de convexidade, todo ponto estacionário torna-se uma solução global do problema. Em particular, qualquer minimizador local é global.

**Definição 3.1.6** (Conjunto convexo). *Seja  $D \subset \mathbb{R}^n$ . Dizemos que  $D$  é convexo, se*

$$\alpha x + (1 - \alpha)y \in D, \quad \forall x, y \in D, \quad \forall \alpha \in [0, 1].$$

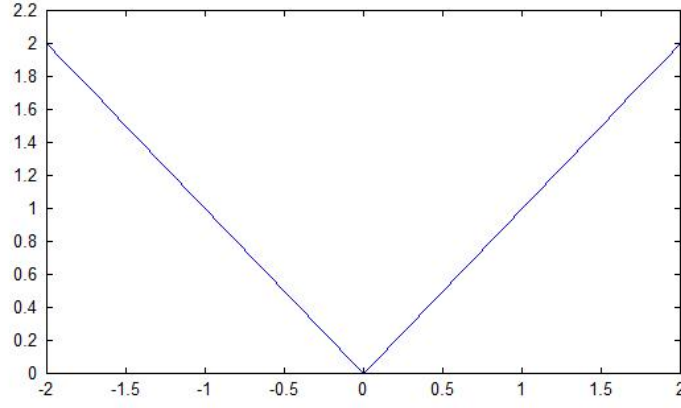
*O ponto  $\alpha x + (1 - \alpha)y$ , onde  $\alpha \in [0, 1]$ , chama-se combinação convexa de  $x$  e  $y$  (com parâmetro  $\alpha$ ).*

O conjunto vazio, o espaço  $\mathbb{R}^n$  e um conjunto que contém um ponto só são trivialmente convexos.

**Definição 3.1.7** (Função convexa). *Seja  $D \subset \mathbb{R}^n$  um conjunto convexo. A função  $f : D \rightarrow \mathbb{R}$  é dita convexa em  $D$  quando para quaisquer  $x, y \in D$ , e  $\alpha \in [0, 1]$ , tem-se:*

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

*Diremos que uma função  $f$  é côncava, se  $-f$  é convexa.*

Figura 3.1:  $f(x) = |x|$ .

**Exemplo 3.1.1.** A função  $f(x) = |x|$  é convexa em  $\mathbb{R}$ .

De fato, sejam  $x, y \in \mathbb{R}$ . Aplicando a definição 3.1.7, temos:

$$f(\alpha x + (1 - \alpha)y) = |\alpha x + (1 - \alpha)y| \leq \alpha|x| + (1 - \alpha)|y| = \alpha f(x) + (1 - \alpha)f(y).$$

Logo, a função  $f(x) = |x|$  é convexa em  $\mathbb{R}$ .

O gráfico desta função é exibido na Figura 3.1.1.

A seguir, estudaremos uma generalização do conceito de gradiente, chamada *subgradiente*, para o caso de funções convexas não diferenciáveis.

**Definição 3.1.8** (Subgradiente). Seja  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função convexa. Dizemos que  $y \in \mathbb{R}^n$  é um subgradiente de  $f$  no ponto  $x \in \mathbb{R}^n$  se

$$f(z) \geq f(x) + \langle y, z - x \rangle, \quad \forall z \in \mathbb{R}^n,$$

$$\text{em que } \langle a, b \rangle = \sum_{i=1}^n a_i b_i, \quad a, b \in \mathbb{R}^n.$$

O conjunto de todos os subgradientes de  $f$  em  $x$  se chama o subdiferencial de  $f$  em  $x$ , e o denotamos por  $\partial f(x)$ .

O subgradiente nada mais é do que uma aproximação linear de  $f$  cujo gráfico fica abaixo do gráfico de  $f$ , e cujo valor coincide com  $f$  no ponto  $x$ .

**Definição 3.1.9** (Direção de descida). Dizemos que  $d \in \mathbb{R}^n$  é uma direção de descida da função  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  no ponto  $x \in \mathbb{R}^n$ , se existe  $\varepsilon > 0$  tal que:

$$f(x + td) < f(x), \quad \forall t \in (0, \varepsilon].$$

Denotamos por  $\mathcal{D}_f(x)$  o conjunto de todas as direções de descida da função  $f$  no ponto  $x$ .

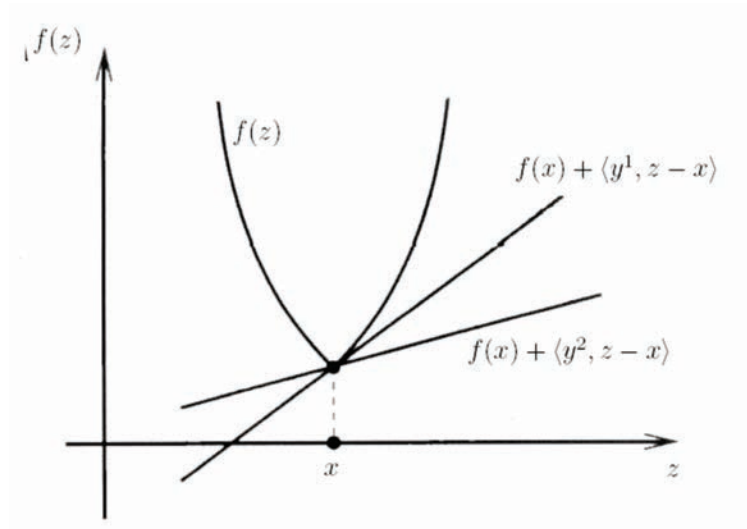


Figura 3.2: Os elementos  $y^1$  e  $y^2$  são subgradientes de  $f$  em  $x$ , isto é, para todo  $z$  tem-se que  $f(z) \geq f(x) + \langle y^i, z - x \rangle$ ,  $i = 1, 2$ .

Uma condição necessária e suficiente para que  $d \in \mathbb{R}^n$  seja uma direção de descida da função  $f$ , convexa em  $\mathbb{R}^n$ , no ponto  $x$  é:

$$d \in \mathcal{D}_f(x) \Leftrightarrow \langle y, d \rangle < 0, \forall y \in \partial f(x).$$

Se tomarmos um subgradiente qualquer, a direção de anti-subgradiente pode não ser de descida. Veja o exemplo abaixo:

**Exemplo 3.1.2.** *Seja  $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ ,  $f(x) = |x_1| + 2|x_2|$ . Veja o gráfico desta função na Figura 3.3.*

*Consideremos um ponto  $x = (x_1, 0)$ , tal que  $x_1 > 0$  é arbitrário.*

*É fácil verificar que*

$$y = (1, 2) \in \partial f(x),$$

*pois*

$$\begin{aligned} f(x) + \langle y, z - x \rangle &= f(x_1, 0) + \langle (1, 2), (z_1, z_2) - (x_1, 0) \rangle \\ &= |x_1| + \langle (1, 2), (z_1 - x_1, z_2) \rangle \\ &= |x_1| + z_1 - x_1 + 2z_2 \\ &\stackrel{x_1 > 0}{=} x_1 + z_1 - x_1 + 2z_2 \\ &= z_1 + 2z_2 \\ &\leq |z_1| + 2|z_2| \\ &= f(z), \forall z \in \mathbb{R}^2. \end{aligned}$$



No entanto, para todo  $t > 0$  suficientemente pequeno,

$$f(x - ty) = |x_1 - t| + 2|0 - 2t| = x_1 - t + 4t = x_1 + 3t > x_1 = f(x),$$

o que mostra que

$$-y \notin \mathcal{D}_f(x).$$

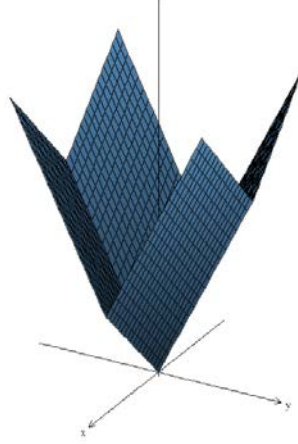


Figura 3.3: Gráfico de  $f(x) = |x_1| + 2|x_2|$ .

**Teorema 3.1.2.** *Sejam  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  e  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, p$ , funções convexas. Então,*

(a)  $\forall x \in \mathbb{R}^n$ ,  $\partial f(x)$  é convexo, compacto e não-vazio;

(b)  $f$  é diferenciável em  $x \Leftrightarrow \partial f(x) = \{f'(x)\}$ ;

(c)  $\bar{x}$  é minimizador de  $f$  no  $\mathbb{R}^n \Leftrightarrow 0 \in \partial f(\bar{x})$ ;

$$(d) \partial \left( \sum_{i=1}^p f_i(x) \right) = \sum_{i=1}^p \partial f_i(x).$$

*Demonstração.* Ver Teorema 1.3.9 em [42] e Proposição 3.4.8 em [41]. □

Na Definição 3.1.10 abaixo, apresentamos uma noção muito importante do ponto de vista computacional, a de subgradiente aproximado. Segundo Izmailov e Solodov [41], sua utilidade é analisar a convergência de vários métodos iterativos para otimização não diferenciável.

**Definição 3.1.10** ( $\varepsilon$  - subgradiente). *Sejam  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função convexa e  $\varepsilon \in \mathbb{R}_+$ . Dizemos que  $y \in \mathbb{R}^n$  é um  $\varepsilon$ -subgradiente de  $f$  no ponto  $x \in \mathbb{R}^n$  quando*

$$f(z) \geq f(x) + \langle y, z - x \rangle - \varepsilon, \quad \forall z \in \mathbb{R}^n.$$

O conjunto de todos os  $\varepsilon$ -subgradientes de  $f$  em  $x$ , denotado por  $\partial_\varepsilon f(x)$ , é chamado de  $\varepsilon$ -subdiferencial de  $f$  em  $x$ .

Um outro resultado importante que nos será útil nos capítulos seguintes é o Teorema 3.1.3.

**Teorema 3.1.3.** *Sejam  $D \subset \mathbb{R}^n$  um conjunto convexo e aberto, e  $f : D \rightarrow \mathbb{R}$  uma função convexa em  $D$ . Então,  $f$  é contínua em  $D$ .*

*Demonstração.* Ver Teorema 3.4.2 em [41]. □

Apresentaremos a seguir métodos para resolver o problema (3.1), escrito da forma geral

$$\min f(x) \text{ sujeito a } x \in X, \quad (3.1)$$

tal que o conjunto factível  $X \subset \mathbb{R}^n$  é convexo e a função objetivo  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  é convexa e não necessariamente diferenciável em todos os pontos.

Um dos métodos mais básicos para resolvermos o problema primal (3.1) é gerar uma sequência  $\{x^k\}$  tal que a função objetivo  $f(x^k)$  decresce em cada iteração, e as direções de descida são as caracterizadas na Definição 3.1.9. Um pseudocódigo para este método, Método de Descida, é exibido na Figura 3.4.

Método de Descida	
1.	Tome $x_0$ inicial e $k = 1$ .
2.	Se $0 \in \partial f(x^k)$ e $x^k \in X$ então PARE.
3.	Calcule uma direção de descida $d^k$ ;
4.	Defina um tamanho de passo $s_k$ onde $f(x^k + s_k d^k) < f(x^k)$ ;
5.	Atualize $x^{k+1} = x^k + s_k d^k$ ;
6.	Incremente $k = k + 1$ e volte ao Passo 2.

Figura 3.4: Pseudocódigo do Método de Descida.

O critério de parada  $0 \in \partial f(x^k)$  é justificado da seguinte maneira:

$$\begin{aligned} 0 \in \partial f(x^k) &\Leftrightarrow f(y) \geq f(x^k) - \langle 0, y - x^k \rangle, \quad \forall y \in \mathbb{R}^n, \\ &\Leftrightarrow f(y) \geq f(x^k), \quad \forall y \in \mathbb{R}^n, \end{aligned}$$

ou seja, se  $0$  é um subgradiente da função  $f$  em  $x^k$ , então  $x^k$  é a solução do problema (3.1), desde que  $x^k \in X$ .

Observamos que o teste de parada descrito no passo 2 é impraticável ou até mesmo impossível do ponto de vista computacional, pois não conhecemos todo o conjunto subdiferencial. Um bom exemplo desta situação é a relaxação Lagrangeana, onde a avaliação da função dual num dado ponto fornece apenas um subgradiente neste ponto. Assim, este método não é implementável. Nas próximas seções veremos algoritmos que tentam contornar essa questão.

## 3.2 Algoritmo do Subgradiente

Vimos na seção 3.1 que é impraticável conhecer todo o conjunto subdiferencial. Uma maneira de contornarmos isso é exigir o cálculo de apenas um subgradiente. Outro detalhe infeliz de algoritmos similares ao algoritmo exibido na Figura 3.4 é que nem sempre obtemos direções reais de descida, pois nem sempre a direção antissubgradiente é uma direção de descida, como vimos no Exemplo 3.1.2. Logo, para um algoritmo de resolução ser considerado eficiente ele tem que gerar e reconhecer candidatos reais de descida, ou seja, direções que realmente causem um decréscimo na função objetivo.

A ideia do Algoritmo de Subgradiente é baseada no método de Cauchy, onde uma direção de descida é a oposta ao vetor gradiente. No caso do método de Subgradientes, tomaremos o vetor oposto ao subgradiente.

Fukuda [24] diz que escolhas adequadas do tamanho do passo podem garantir a convergência do método, apesar de não gerar decréscimo a cada iteração na função objetivo. Na realidade, esta dificuldade em gerar decréscimos é contornada fixando inicialmente uma sequência de valores de comprimentos de passos. Entretanto, Solodov [42] diz que esta estratégia não é satisfatória na prática, apesar de resolver a dificuldade no âmbito teórico.

---

### Algoritmo de Subgradiente Básico

---

1. Escolha sequências de valores reais positivos  $\{\varepsilon_k\}$ ,  $\{s_k\}$ ;
  2. Tome  $x^0$  um vetor inicial e inicialize  $k = 0$ ;
  3. Calcule  $d^k \in \partial_{\varepsilon_k} f(x^k)$ ;
  4. Atualize  $x^{k+1} = x^k + s_k d^k$ ;
  5. Incremente  $k = k + 1$  e volte ao Passo 1.
- 

Figura 3.5: Pseudocódigo do Algoritmo de Subgradiente

O Algoritmo de Subgradiente depende então de três sequências:  $\{\varepsilon_k\}$ ,  $\{s_k\}$  e  $\{d^k\}$ . Claramente, não são quaisquer sequências que fazem o algoritmo convergir. Vejamos a seguir, quais condições tais sequências devem possuir para que o Algoritmo de Subgradiente convirja [42].

**Teorema 3.2.1.** *Seja  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função convexa no  $\mathbb{R}^n$ . Suponhamos que as sequências  $\{\varepsilon_k\}$ ,  $\{s_k\}$  e  $\{d^k\}$  escolhidas no Algoritmo de Subgradiente Básico, Figura 3.5, satisfaçam as condições:*

$$\sum_{k=0}^{+\infty} s_k = +\infty, \quad (3.2)$$

$$\lim_{k \rightarrow +\infty} \varepsilon_k = 0, \quad (3.3)$$

$$\lim_{k \rightarrow +\infty} s_k \|d_k\|^2 = 0. \quad (3.4)$$

Então, a sequência  $\{x^k\}$  gerada pelo Algoritmo de Subgradiente Básico converge para o ótimo de  $f$ , ou seja,

$$\liminf_{k \rightarrow +\infty} f(x^k) = \bar{v},$$

onde  $\bar{v} = \inf_{x \in \mathbb{R}^n} f(x)$ .

*Demonstração.* A ideia desta demonstração é usar o argumento da Redução ao Absurdo. Iremos supor que a sequência  $\{x^k\}$  não converge para  $\bar{v}$  e concluir que a série  $\sum_{k=0}^{+\infty} s_k$  diverge contradizendo as hipóteses do Teorema.

Sabemos da Definição 3.1.10, que:

$$\partial_\varepsilon f(x) = \{y \in \mathbb{R}^n \mid f(z) \geq f(x) + \langle y, z - x \rangle - \varepsilon, \forall z \in \mathbb{R}^n\},$$

e considere que a atualização da sequência  $\{x^k\}$  é dada por:

$$x^{k+1} = x^k + s_k d^k.$$

Assim, para qualquer  $x \in \mathbb{R}^n$ , temos que:

$$\begin{aligned} \|x^{k+1} - x\|^2 &= \|(x^k - x) + (x^{k+1} - x^k)\|^2 \\ &= \|x^k - x\|^2 + 2\langle x^k - x, x^{k+1} - x^k \rangle + \|x^{k+1} - x^k\|^2 \\ &= \|x^k - x\|^2 + 2\langle x^k - x, -s_k d^k \rangle + \|s_k d^k\|^2 \\ &= \|x^k - x\|^2 + 2s_k \langle d^k, x - x^k \rangle + s_k^2 \|d^k\|^2 \\ &\leq \|x^k - x\|^2 + 2s_k (f(x) - f(x^k) + \varepsilon_k) + s_k^2 \|d^k\|^2 \end{aligned} \quad (3.5)$$

Vamos supor, por absurdo, que  $\liminf_{k \rightarrow +\infty} f(x^k) > \bar{v}$ .

Assim, como  $\bar{v} = \inf_{x \in \mathbb{R}^n} f(x)$ , existem  $x \in \mathbb{R}^n$ ,  $\delta > 0$  e  $\bar{k} \in \mathbb{N}_+$ , tais que

$$f(x) < f(x^k) - \delta, \quad \forall k \geq \bar{k}. \quad (3.6)$$

Como temos (3.3) e (3.4) por hipótese, podemos tomar  $\bar{k}$  suficientemente grande de tal forma que  $2\varepsilon_k \leq \delta/2$  e  $s_k \|d^k\|^2 \leq \delta/2$ , ou seja,

$$s_k \|d^k\|^2 + 2\varepsilon_k \leq \delta, \quad \forall k \geq \bar{k}. \quad (3.7)$$

Substituindo as relações (3.6) e (3.7) em (3.5), temos que

$$\begin{aligned}
\|x^{k+1} - x\|^2 &\leq \|x^k - x\|^2 + 2s_k(f(x) - f(x^k) + \varepsilon_k) + s_k^2\|d^k\|^2 \\
&< \|x^k - x\|^2 + 2s_k(-\delta + \varepsilon_k) + s_k^2\|d^k\|^2 \\
&= \|x^k - x\|^2 - 2s_k\delta + 2s_k\varepsilon_k + s_k^2\|d^k\|^2 \\
&= \|x^k - x\|^2 - 2s_k\delta + s_k(2\varepsilon_k + s_k\|d^k\|^2) \\
&\leq \|x^k - x\|^2 - 2s_k\delta + s_k\delta \\
&\leq \|x^k - x\|^2 - s_k\delta, \quad \forall k \geq \bar{k}.
\end{aligned}$$

Assim, para qualquer  $k \in \mathbb{N}$ , com  $k \geq \bar{k}$ , temos que:

$$s_k\delta \leq \|x^k - x\|^2 - \|x^{k+1} - x\|^2.$$

Somando de  $\bar{k}$  a  $k$ , segue que

$$\begin{aligned}
\sum_{i=\bar{k}}^k s_i\delta &\leq \sum_{i=\bar{k}}^k \|x^i - x\|^2 - \|x^{i+1} - x\|^2 \\
&\leq \|x^{\bar{k}} - x\|^2 - \|x^{\bar{k}+1} - x\|^2 + \dots + \|x^k - x\|^2 - \|x^{k+1} - x\|^2 \\
&\leq \|x^{\bar{k}} - x\|^2 - \|x^{k+1} - x\|^2 \\
&< \|x^{\bar{k}} - x\|^2,
\end{aligned}$$

ou ainda,

$$\sum_{i=\bar{k}}^k s_i < \frac{1}{\delta} \|x^{\bar{k}} - x\|^2,$$

Daí, quando  $k \rightarrow +\infty$ , a série  $\sum_{i=\bar{k}}^{+\infty} s_i$  converge, contradizendo a hipótese (3.2).

Portanto, o Algoritmo de Subgradiente Básico gera uma sequência que converge para o ótimo.  $\square$

O algoritmo de subgradiente mais empregado usa direções normalizadas e  $\varepsilon_k = 0$ ,  $\forall k$  [42]. Assim temos o seguinte teorema:

**Teorema 3.2.2.** *Seja  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  uma função convexa no  $\mathbb{R}^n$ . Supondo que o problema tenha uma solução e que utilizamos as sequências,*

$$\varepsilon_k = 0, \quad s_k = \frac{\beta_k}{\|d^k\|}, \quad \forall k \tag{3.8}$$

no Algoritmo de Subgradiente Básico, de tal forma que

$$\sum_{k=0}^{+\infty} \beta_k = +\infty, \quad \sum_{k=0}^{+\infty} \beta_k^2 < +\infty, \quad (3.9)$$

então, a sequência  $\{x^k\}$  gerada pelo Algoritmo de Subgradiente Básico converge para uma solução do problema.

*Demonstração.* Seja  $\bar{x}$  a solução do problema. Assim,  $f(\bar{x}) \leq f(x^k)$ ,  $\forall k$ , ou seja,

$$f(\bar{x}) - f(x^k) \leq 0, \quad \forall k. \quad (3.10)$$

Usando a hipótese de que  $\varepsilon_k = 0$  e  $\beta_k = s_k \|d^k\|$ ,  $\forall k$ , e tomando  $x = \bar{x}$  em (3.5), temos

$$\begin{aligned} \|x^{k+1} - \bar{x}\|^2 &\leq \|x^k - \bar{x}\|^2 + 2s_k(f(\bar{x}) - f(x^k) + \varepsilon_k) + s_k^2 \|d^k\|^2 \\ &\leq \|x^k - \bar{x}\|^2 + \beta_k^2. \end{aligned} \quad (3.11)$$

Seja  $k$  um valor fixo. Daí, para todo  $j \geq k+1$ , teremos

$$\begin{aligned} \|x^j - \bar{x}\|^2 &\leq \|x^{j-1} - \bar{x}\|^2 + \beta_{j-1}^2 \\ &\leq \|x^{j-2} - \bar{x}\|^2 + \beta_{j-1}^2 + \beta_{j-2}^2 \\ &\vdots \\ &\leq \|x^k - \bar{x}\|^2 + \sum_{i=k}^{j-1} \beta_i^2 \\ &\leq \|x^k - \bar{x}\|^2 + \sum_{i=0}^{+\infty} \beta_i^2 \\ &< +\infty, \end{aligned} \quad (3.12)$$

ou seja, a sequência  $\{x^k\}$  é limitada.

Observe que como  $\partial f(x^k)$  é compacto (Teorema 3.1.2), segue que a sequência  $\{d^k\}$  também é limitada, pois  $d^k \in \partial f(x)$ , para todo  $k$ .

Como  $\{\beta_k\} \subset \mathbb{R}_+$ , pois  $\{s_k\} \subset \mathbb{R}_+$  e temos (3.9), então  $\lim_{k \rightarrow \infty} \beta_k^2 = 0$ , e assim,  $\lim_{k \rightarrow \infty} \beta_k = 0$ .

Desta forma,

$$\lim_{k \rightarrow \infty} s_k \|d^k\|^2 = \lim_{k \rightarrow \infty} \frac{\beta_k}{\|d^k\|} \|d^k\|^2 = \lim_{k \rightarrow \infty} \beta_k \|d^k\| = 0,$$

pois  $\beta_k \rightarrow 0$  e  $\{d^k\}$  é limitada (Teorema 7, Capítulo VI, Lima [50]).

Ainda,  $\sum_{k=0}^{+\infty} s_k = +\infty$ , por (3.9) e  $\lim_{k \rightarrow +\infty} \varepsilon_k = 0$ , pois  $\varepsilon_k = 0$ ,  $\forall k$ . Portanto, estamos nas hipóteses do Teorema 3.2.2.

Logo,  $\liminf_{k \rightarrow +\infty} f(x^k) = f(\bar{x})$ .

Como  $f$  é contínua, pois é convexa (Teorema 3.1.3), e  $\{x^k\}$  é limitada, existe um ponto de acumulação  $\hat{x}$  de  $\{x^k\}$  onde  $f(\bar{x}) = f(\hat{x})$ . Isto é,  $\hat{x}$  é uma solução do problema.

Tomando  $\bar{x} = \hat{x}$  em (3.12), temos que

$$\|x^j - \hat{x}\|^2 \leq \|x^k - \hat{x}\|^2 + \sum_{i=k}^{+\infty} \beta_i^2, \quad (3.13)$$

e tendo (3.9) como hipótese, segue que

$$\lim_{k \rightarrow +\infty} \sum_{i=k}^{+\infty} \beta_i^2 = 0.$$

Por definição de limite, dado  $\delta > 0$ , existe  $k$  tal que

$$\sum_{i=k}^{+\infty} \beta_i^2 < \frac{\delta}{2}.$$

Ainda, por  $\hat{x}$  ser ponto de acumulação da sequência  $\{x^k\}$  podemos escolher um  $k$  de tal forma que

$$\|x^k - \hat{x}\|^2 < \frac{\delta}{2}.$$

Logo, por (3.13) segue que para todo  $\delta > 0$ , existe  $k$  tal que

$$\|x^j - \hat{x}\|^2 < \delta, \quad \forall j \geq k + 1,$$

ou seja, a sequência  $\{x^k\}$  gerada pelo Algoritmo de Subgradiente Básico converge para a solução  $\hat{x}$  do problema. □

A partir destes resultados, podemos derivar valores para o tamanho do passo  $s_k$  de tal forma que o Algoritmo de Subgradiente convirja.

Tome  $\bar{x}$  um minimizador de  $f$ . Então, de (3.5), com  $\varepsilon_k = 0$ ,  $\forall k$ , temos que

$$\|x^{k+1} - \bar{x}\|^2 \leq \|x^k - \bar{x}\|^2 - 2s_k(f(x^k) - f(\bar{x})) + s_k^2 \|d^k\|^2,$$

e concluímos que:

$$\|x^{k+1} - \bar{x}\|^2 < \|x^k - \bar{x}\|^2$$

se

$$0 < s_k < \frac{2(f(x^k) - f(\bar{x}))}{\|d^k\|^2}. \quad (3.14)$$

Entretanto, como não conhecemos o valor ótimo do problema, não há como escolhermos um comprimento de passo que garanta a propriedade (3.14). Naturalmente, outras condições de tamanho de passo podem garantir a convergência do método. De fato, podemos generalizar a condição (3.14) tomando, por exemplo, um limite inferior LB para  $f$ , ao invés de usar o valor  $f(\bar{x})$ . Temos assim um tamanho de passo que não depende de  $f(\bar{x})$ :

$$0 < s_k < \frac{2(f(x^k) - LB)}{\|d^k\|^2}. \quad (3.15)$$

Os métodos de subgradientes são muito atrativos pela sua extrema simplicidade, mas geralmente são pouco eficientes pois não possuem uma regra de parada razoável (normalmente é limitado pela quantidade de iterações) e não há garantia de decréscimo da função objetivo em toda iteração. Uma generalização deste método que contorna estes problemas é o Algoritmo de Volume proposto por Barahona e Anbil [21] e que apresentamos na seção 3.3.

### 3.3 Algoritmo de Volume

Apresentamos nesta seção o Algoritmo de Volume que é uma generalização do Algoritmo de Subgradientes. Esse algoritmo contorna os problemas que o algoritmo de subgradientes apresenta, e produz uma aproximação de solução primal enquanto se resolve o dual. A solução dual é obtida semelhantemente à do algoritmo de subgradientes, e a solução primal é obtida estimando-se o volume formado pelas faces ativas relativas às restrições dualizadas, o que justifica o nome do algoritmo.

Antes de apresentarmos o Algoritmo de Volume, faremos uma breve introdução ao método que fundamentou seu desenvolvimento (juntamente com o Método de Subgradiente), que é o Método de Feixes [42], [24].

Considere o seguinte problema de otimização irrestrito:

$$\min f(x), \quad x \in \mathbb{R}^n, \quad (3.16)$$

tal que  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  é uma função convexa e não diferenciável.

O método de feixes baseia-se em aproximar a função objetivo  $f(x)$  por funções lineares, ou seja, aproximação linear por partes, ao mesmo tempo agregando um artifício de estabilização



e uma regra de parada confiável. O método consiste em gerar duas sequências de pontos: a sequência  $\{x^k\}$ , que são os *candidatos*, ou seja, os valores iterados do algoritmo, e a sequência  $\{\hat{x}^k\}$ , que são os *centros de estabilização*, ou seja, os pontos que realmente decrescem o valor da função objetivo.

A cada iteração  $k$  do método, um problema linear por partes  $\psi_k(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  é construído de acordo com (3.17):

$$\psi_k(x) = \max_{i=0,1,\dots,k} \{f(x^i) + \langle v^i, x - x^i \rangle\}, \quad v^i \in \partial f(x^i), \quad \forall i. \quad (3.17)$$

Observe que a sequência  $\{\psi_k\}$  é uma sequência não-decrescente, pois  $\psi_k \leq \psi_{k+1}$ ,  $\forall k$  e, ainda,  $\psi_k \leq f$ . Desta forma, estamos aproximando a função  $f$  pela função  $\psi_k$  para cada  $k$ , inferiormente. Podemos ver essa propriedade na Figura 3.6.

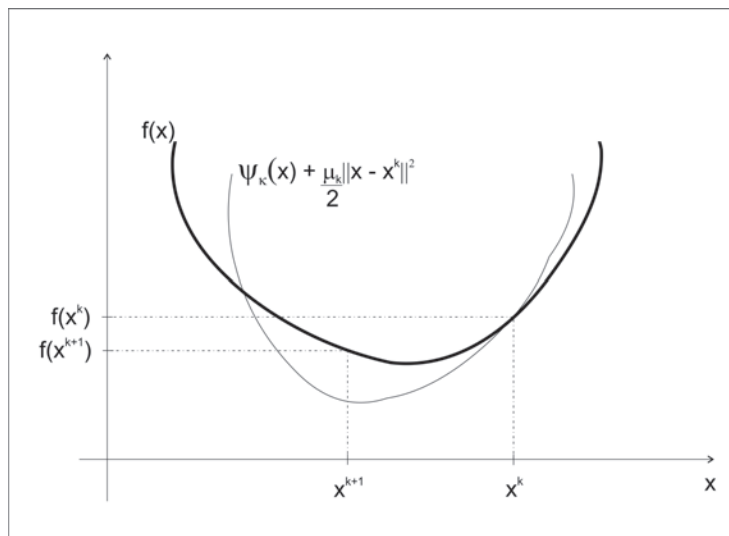


Figura 3.6: Um passo do método de feixes em que foi gerado um ponto  $x^{k+1}$  com valor de  $f$  menor de  $f(x^k)$ .

Precisaremos da noção de centro de estabilização, cuja definição está na Definição 3.3.1.

**Definição 3.3.1** (Centro de estabilização). *Um iterado  $x^{k+1}$  torna-se um centro de estabilização, isto é,  $\hat{x}^{k+1} = x^{k+1}$  se satisfizer uma condição do tipo Armijo, ou seja,*

$$f(\hat{x}^k) - f(x^{k+1}) \geq m\delta_{k+1}, \quad (3.18)$$

em que  $\delta_{k+1}$  é denominado decréscimo nominal e  $m$  é um parâmetro pertencente ao intervalo  $(0, 1)$ . Caso o iterado se torne um centro de estabilização, dizemos que foi realizado um passo sério, caso contrário, diremos que o passo foi nulo e  $\hat{x}^{k+1} = \hat{x}^k$ .

**Definição 3.3.2** (Decréscimo nominal). *O decréscimo nominal citado na Definição 3.3.1 é calculado da seguinte maneira:*

$$\delta_{k+1} = f(\hat{x}^k) - \left( \psi_k(x^{k+1}) + \frac{\mu_k}{2} \|x^{k+1} - \hat{x}^k\|^2 \right). \quad (3.19)$$

A sequência de centros de estabilização  $\{\hat{x}^1, \hat{x}^2, \dots, \hat{x}^k\}$  é uma subsequência de  $\{x^1, x^2, \dots, x^k\}$ .

A ideia do método de feixes é obter o próximo iterado  $x^{k+1}$  utilizando o problema (3.17).

O iterado  $x^{k+1}$  será a solução do seguinte problema, denominado problema estabilizado:

$$\min_{x \in \mathbb{R}^n} \psi_k(x) + \frac{\mu_k}{2} \|x - \hat{x}^k\|^2. \quad (3.20)$$

O termo quadrático da função objetivo do problema (3.20) tem o papel de não permitir um deslocamento muito grande do ponto  $x^{k+1}$  em relação ao centro  $\hat{x}^k$ . O centro de estabilização será mudado apenas quando o passo de  $\hat{x}^k$  a  $x^{k+1}$  resultar em um decréscimo suficiente da função objetivo  $f$ , ou seja, apenas quando o decréscimo nominal for verificado.

Com estas considerações, podemos desenvolver um pseudocódigo para o Método de Feixes apresentado na Figura 3.7.

Método de Feixes	
1.	Tome uma tolerância $\delta_{min}$ e $m \in (0, 1)$ ;
2.	Inicialize $x^1$ um ponto qualquer, $k = 1$ e $\delta_1 = \infty$ ;
3.	Calcule $v^1 \in \partial f(x^1)$ e $f(x^1)$ ;
4.	Construa o modelo $\psi_1(x) = f(x^1) + \langle v^1, x - x^1 \rangle$ e calcule $\delta_1$ ;
5.	Se $\delta_k \leq \delta_{min}$ então PARE;
6.	Senão,
7.	Resolva o problema (3.20) e tome $x^{k+1}$ o minimizador;
8.	Calcule o decréscimo nominal $\delta_{k+1}$ ;
9.	Se $f(\hat{x}^k) - f(x^{k+1}) \geq m\delta_{k+1}$
10.	Então $\hat{x}^{k+1} = x^{k+1}$ e atualize $\mu_{k+1}$ ;
11.	Senão,
12.	Faça $\hat{x}^{k+1} = x^k$ ;
13.	Construa o modelo $\psi_{k+1} = \max_{i=1,2,\dots,k} \{f(x^i) + \langle v^i, x - x^i \rangle\}$ ;
14.	Incremente $k = k + 1$ e volte ao Passo 5.

Figura 3.7: Pseudocódigo do Método de Feixes.

O parâmetro  $\mu_{k+1}$  é atualizado resolvendo a seguinte equação [24]:

$$\frac{1}{\mu_{k+1}} = \frac{1}{\mu_k} + \frac{\langle \hat{v}^{k+1} - \hat{v}^k, \hat{x}^{k+1} - \hat{x}^k \rangle}{\|\hat{v}^{k+1} - \hat{v}^k\|^2}, \quad (3.21)$$

em que  $\hat{v}_k \in \partial f(x^{k+1})$ .

Observe que no caso de um passo nulo, ou seja,  $\hat{x}^{k+1} = \hat{x}^k$ , a fórmula resulta em  $\mu_{k+1} = \mu_k$ .

Os métodos de feixe são métodos robustos, com regras de parada confiáveis, mas demandam um esforço computacional considerável pois tem-se que resolver vários problemas de otimização quadráticos.

Uma alternativa à este método é o denominado Algoritmo de Volume, que passaremos a apresentar. Suponhamos que o problema que temos interesse em resolver apresenta a seguinte estrutura:

$$\begin{aligned} \min \quad & cx \\ \text{s.a.} \quad & Ax \leq b, \\ & x \in \mathbb{Q}, \end{aligned} \tag{3.22}$$

em que  $\mathbb{Q}$  é um poliedro não vazio correspondente às restrições que não complicam a resolução do problema (3.22),  $c \in \mathbb{R}^n$ ,  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  e as restrições  $Ax \leq b$  são as que complicam o nosso problema.

Já vimos na subseção 2.2.2 que a função Lagrangeana associada às restrições  $Ax \leq b$  é dada por:

$$Z(\pi, x) = cx + \pi(Ax - b), \tag{3.23}$$

e o problema Lagrangeano é definido por

$$Z_{RL}(\pi) = \min\{Z(\pi, x) \mid x \in \mathbb{Q}\}. \tag{3.24}$$

**Proposição 3.3.1.** [24] Dado  $x^k \in \arg \min Z(\pi, x)$ , então  $v^k = Ax^k - b \in \partial Z_{RL}(\pi)$ .

*Demonstração.* Para todo  $\pi$ , temos que

$$\begin{aligned} Z_{RL}(\pi) &= \min Z(\pi, x) \\ &\leq Z(\pi, x^k) \\ &= cx^k + \pi(Ax^k - b) \\ &= cx^k + \pi(Ax^k - b) + \pi^k(Ax^k - b) - \pi^k(Ax^k - b) \\ &= cx^k + \pi^k(Ax^k - b) + (\pi - \pi^k)(Ax^k - b) \\ &= Z_{RL}(\pi^k) + (\pi - \pi^k)(Ax^k - b), \end{aligned}$$

ou seja,  $Z_{RL}(\pi) \leq Z_{RL}(\pi^k) + (\pi - \pi^k)(Ax^k - b)$ , e assim o resultado segue.  $\square$

Assim como vimos no Método de Feixes, o Algoritmo de Volume trabalha com duas sequências: a sequência  $\bar{\pi}_k$  de pontos candidatos, e a sequência  $\hat{\pi}_k$  de centros de estabilização. Além disso, as direções  $\hat{v}_k$  serão dadas por combinações convexas de subgradientes. A estrutura geral do Algoritmo de Volume é apresentada na Figura 3.8.

---



---

### Algoritmo de Volume

---

1. Sejam  $0 < \alpha < 1$  um parâmetro,  $\xi_{pd}$  tolerância de parada primal-dual
  2. e  $\xi_v$  tolerância de parada de viabilidade;
  3. Tome  $\bar{\pi}_0$  inicial e resolva o problema (3.24) obtendo  $\bar{x}_0$  e  $Z_{RL}(\bar{\pi}_0)$ ;
  4. Calcule  $\bar{v}_0 = A\bar{x}_0 - b$ ;
  5. Inicialize  $\hat{x}_1 = \bar{x}_0$ ,  $\hat{v}_1 = \bar{v}_0$ ,  $\hat{\pi}_1 = \bar{\pi}_0$  e  $k = 1$ ;
  6. Se  $\|\hat{v}_k\| < \xi_v$  e  $\frac{c\hat{x}_k - Z_{RL}(\hat{\pi}_k)}{Z_{RL}(\hat{\pi}_k)} < \xi_{pd}$  então PARE
  7. Calcule  $s_k = \mu \frac{UB - Z_{RL}(\bar{\pi}_k, \bar{x}_k)}{\|\hat{v}_k\|^2}$ ;
  8. Compute  $\bar{\pi}_k = \hat{\pi}_k + s_k \hat{v}_k$ ;
  9. Resolva o problema (3.24) obtendo  $\bar{x}_k$ ,  $Z_{RL}(\bar{\pi}_k)$  e  $\bar{v}_k = A\bar{x}_k - b$ ;
  10. Se  $Z_{RL}(\bar{\pi}_k) > Z_{RL}(\hat{\pi}_k)$  então  $\hat{\pi}_{k+1} = \bar{\pi}_k$ ;
  11. Senão,  $\hat{\pi}_{k+1} = \hat{\pi}_k$ ;
  11. Atualize  $\hat{x}_{k+1} = \alpha \bar{x}_k + (1 - \alpha)\hat{x}_k$  e  $\hat{v}_{k+1} = \alpha \bar{v}_k + (1 - \alpha)\hat{v}_k$ ;
  12. Incremente  $k = k + 1$  e volte ao Passo 5.
- 
- 

Figura 3.8: Pseudocódigo do Algoritmo de Volume

Um resultado muito importante é uma relação que temos entre os pontos  $\hat{x}_k$  e  $\hat{v}_k$  apresentada na forma da Proposição 3.3.2 a qual será substituída no Algoritmo de Volume.

**Proposição 3.3.2.** *Sejam  $\{\hat{x}_k\}$  e  $\{\hat{v}_k\}$  as sequências geradas pelo Algoritmo de Volume. Então,  $\hat{v}_k = A\hat{x}_k - b$ , em toda iteração  $k$ .*

*Demonstração.* [24] Nesta demonstração usaremos o Princípio de Indução Finita (PIF) para mostrar o resultado.

Quando  $k = 1$ , temos pelas definições dadas no Algoritmo de Volume:

$$\hat{v}_1 = \bar{v}_0 = A\bar{x}_0 - b = A\hat{x}_1 - b.$$

Suponhamos que o resultado é válido para uma iteração  $k \geq 1$ , ou seja, que  $\hat{v}_k = A\hat{x}_k - b$ , e mostremos que o resultado é válido para a iteração  $k + 1$  seguinte. De fato,

$$\begin{aligned}
\hat{v}_{k+1} &= \alpha \bar{v}_k + (1 - \alpha) \hat{v}_k \\
&= \alpha (A \bar{x}_k - b) + (1 - \alpha) (A \hat{x}_k - b) \\
&= \alpha A \hat{x}_k - \alpha b + A \hat{x}_k - b - \alpha A \hat{x}_k + b \alpha \\
&= A(\alpha \hat{x}_k + \hat{x}_k - \alpha \hat{x}_k) - b \\
&= A(\alpha \bar{x}_k + (1 - \alpha) \hat{x}_k) - b \\
&= A \hat{x}_{k+1} - b.
\end{aligned}$$

Logo, o resultado segue. □

O parâmetro  $\alpha$  usado no Algoritmo de Volume fica fixo por uma quantidade de iterações pré estabelecidas, e vai sendo atualizado de acordo com o algoritmo apresentado na Figura 3.9. A ideia básica é escolher  $\alpha$  de modo a atingir a viabilidade primal, ou seja, ter um valor para  $\alpha$  de modo que  $\hat{v}_k = A \hat{x}_k - b \approx 0$ .

---

#### Algoritmo para a escolha de $\alpha$

---

1. Seja  $\alpha_{\max}$  um limite superior para  $\alpha$ .
  2. Compute  $\alpha_{\text{opt}} = \arg \min_{\xi \in \mathbb{R}} (\xi \hat{v}_{t+1} + (1 - \xi) \bar{v}_{t+1})^2$
  3. Se  $\alpha_{\text{opt}} < 0$ , então faça  $\alpha = \alpha_{\max}/10$ .
  4. Senão faça  $\alpha = \min\{\alpha_{\text{opt}}, \alpha_{\max}\}$ .
- 

Figura 3.9: Algoritmo para a escolha de  $\alpha$

# Capítulo 4

## Relaxações para o PIDS

Neste capítulo, iremos apresentar o algoritmos de Subgradiente e de Volume descritos nas seções 4.2.1 e 4.2.2, respectivamente, para resolver os problemas duais Lagrangeano e Lagrangeano/Surrogate descritos nas seções 2.2.2 e 2.2.3, respectivamente, particularizados para gerar limitantes para o problema integrado de dimensionamento de lotes e sequenciamento da produção descrito na seção 1.2. Iniciamos o capítulo com uma proposta de algoritmo guloso. A solução gulosa é útil não apenas para fornecer uma solução factível mas também nos Algoritmos de Subgradiente e de Volume fornecendo limitantes primais.

Considere as formulações para o PIDS apresentadas na seção 1.2. As relaxações que apresentamos nas seções 4.2 e 4.3 são baseadas nos modelos MT1S1MP e MCF1S1MP, os quais consideram de forma explícita a variável de preparo  $w_{jt}$ . A escolha destes modelos, como veremos no Capítulo 5, é por sua melhor performance em relação aos modelos MTZ1S1M e MCF1S1M, respetivamente.

### 4.1 Uma heurística gulosa para o PIDS

Uma maneira simples de obtermos limitantes superiores (no caso de problemas de minimização) são as heurísticas gulosas. Desenvolvemos uma heurística gulosa para o PIDS com a seguinte ideia:

- Fase I: Ordenação dos itens em ordem não-decrescente de custo de troca;
- Fase II: Resolvemos o problema de dimensionamento de lotes (PDL), descrito por (4.1)-(4.7):

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt} \quad (4.1)$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t \quad (4.2)$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t \quad (4.3)$$

$$x_{jt} \leq \frac{C_t}{p_j} \left( \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right), \quad \forall j, \forall t \quad (4.4)$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t \quad (4.5)$$

$$I_{j0}^+ = 0, \quad \forall j \quad (4.6)$$

$$I_{j0}^- = 0; \quad \forall j \quad (4.7)$$

- Por fim, verificamos se o sequenciamento construído na Fase I é factível. Se não for, factibilizamos a solução mantendo a ordenação obtida na Fase I e reduzindo a produção com a consequente introdução de atrasos.

O pseudocódigo para a heurística é apresentado na Figura 4.1.

## 4.2 Relaxação Lagrangeana para o PIDS

Nesta seção iremos apresentar o subproblema Lagrangeano para o problema PIDS e os métodos de resolução desenvolvidos para resolvê-lo.

Considere o problema PIDS apresentado na seção 1.2. Dualizando as restrições (1.3) com o multiplicador  $\pi^1 \in \mathbb{R}_+^T$  e (1.17) com o multiplicador  $\pi^2 \in \mathbb{R}_+^{JT}$ , temos o seguinte subproblema Lagrangeano (4.8)-(4.18):

$$\begin{aligned} \text{Min } L(x, I^+, I^-, z, \pi^1, \pi^2) = & \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt} \\ & + \sum_{t \in T} \pi_t^1 \left( \sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} - C_t \right) + \sum_{j \in J} \sum_{t \in T} \pi_{jt}^2 \left( w_{jt} - \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right) \end{aligned} \quad (4.8)$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t \quad (4.9)$$

$$x_{jt} \leq C_t \frac{w_{jt}}{p_j}, \quad \forall j, \forall t \quad (4.10)$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t \quad (4.11)$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t \quad (4.12)$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t \quad (4.13)$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, \forall j, \forall t \quad (4.14)$$

$$\text{Restrições de eliminação de subrotas (MTZ ou MCF)} \quad (4.15)$$

$$I_{j0}^+ = 0, \quad \forall j \quad (4.16)$$

$$I_{j0}^- = 0, \quad \forall j \quad (4.17)$$

$$w_{jt} = 0/1, \quad \forall j, \forall t. \quad (4.18)$$

O interessante na dualização das restrições (1.3) e (1.17) é que o subproblema resultante pode ser decomposto em dois: o problema de dimensionamento de lotes sem capacidade e o problema de sequenciamento. O subproblema Lagrangeano pode então ser resolvido com a resolução destes dois subproblemas independentes. Em nosso estudo utilizamos o *software* CPLEX para resolução dos subproblemas.



---

**Heurística Gulosa**


---

1. Fase 1 - Ordenação dos itens:
  2.     Passo 1: Tome  $i_0$  como item inicial e  $i_1$  o item subsequente tal
  3.             que  $s_{i_0 i_1} = \min\{s_{ij}, \forall i, j\}$ , e  $k = 2$ . Faça  $z_{i_0 i_1 t} = 1, \forall t$ ;
  4.     Passo 2: Enquanto  $k < J$  acrescente à sequência o item  $i_k$  tal
  5.             que  $s_{i_{k-1} i_k} = \min\{s_{ijk}, \forall j\}$  e  $i_k \neq i_0, i_1, \dots, i_{k-1}$ , e
  6.             faça  $z_{i_{k-1} i_k t} = 1, \forall t$ ;
  7. Fase 2 - Dimensionar e factibilizar:
  8.     Passo 1: Resolva o problema PDL: (4.1)-(4.7) ;
  9.     Passo 2: Seja  $z_{ijt}, \forall i, j, t$  a solução da Fase 1 e  $x_{jt}, I_{jt}^+$
  10.             e  $I_{jt}^-, \forall i, j, t$  a solução do dimensionamento de lotes
  11.             obtida no Passo 1;
  12.     Passo 3: Para cada período  $t$ , faça:
  13.             Tome  $a_t = \sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} - C_t$ .
  14.             Se  $a_t \leq 0$  então passe para o próximo período  $t$ .
  15.             Senão, seja  $j_1, j_2, \dots, j_J$  uma ordenação nos itens de
  16.             acordo com seus custos de atraso.
  17.             Tome  $k = 1$ .
  18.             Enquanto  $a_t \neq 0$ , faça:
  19.                 Se  $x_{j_k t} = 0$  então  $k = k + 1$ .
  20.             Senão,
  21.                 Se  $a_t - x_{j_k t} \geq 0$ , então
  22.                      $a_t = a_t - x_{j_k t}$ ,
  23.                      $x_{j_k t} = 0$ ,
  24.                      $z_{mj_k t} = z_{j_k n t} = 0$ , onde  $m$  é o item produzido
  25.                     imediatamente antes de  $j_k$  e  $n$  imediatamente depois,
  26.                      $I_{j_k t}^- = d_{j_k t} - x_{j_k t} + I_{j_k t-1}^-$ .
  27.             Senão,
  28.                      $x_{j_k t} = x_{j_k t} - a_t$ ,
  29.                      $I_{j_k t}^- = d_{j_k t} - x_{j_k t} + I_{j_k t-1}^-$ ,
  30.                      $a_t = 0$ .
  31.             Para todo  $i = 1, \dots, J$  se  $x_{it} - d_{it} = 0$  e  $I_{it-1}^- > 0$ ,
  32.             então  $I_{it}^- = I_{it-1}^-$ .
- 

Figura 4.1: Pseudocódigo para a heurística gulosa

### 4.2.1 Algoritmo de Subgradiente para resolução do Dual Lagrangeano

Apresentamos na Figura 4.2 o pseudocódigo do algoritmo de subgradiente proposto para resolver o problema dual Lagrangeano associado à formulação MTZ1S1MP. Para obter o algoritmo associado à formulação MCF1S1MP basta substituir o problema a ser resolvido no Passo 2 pelo problema (1.1) – (1.7), (1.11), (1.12) e (1.13) – (1.16), e utilizar as restrições MCF no problema (4.8) – (4.18). Denotaremos o multiplicador  $\pi_t^1$  na iteração  $k$  por  $\pi_t^{1,k}$  e o multiplicador  $\pi_{j,t}^2$  na iteração  $k$  por  $\pi_{j,t}^{2,k}$ .

---

#### Algoritmo de Subgradiente para o problema MTZ1S1MP

---

1. Faça  $k = 0$ ,  $\pi_t^{1,0} = 0$ ,  $\pi_{it}^{2,0} = 0, \forall i, t$ , e  $max\_it$  um limitante para a quantidade de iterações.
  2. Seja UB um limite superior para o problema (1.1) – (1.12).
  3. Resolva o subproblema (4.8) – (4.18) (MTZ) com  $\pi_t^1 = \pi_t^{1,k}$ ,  $\pi_{it}^2 = \pi_{it}^{2,k} \forall i, t$  e tome  $x_{jt}^k$ ,  $z_{ijt}^k$  e  $w_{jt}^k$  os minimizadores.
  4. Calcule  $v_t^{1,k} = \sum_{j \in J} p_j x_{jt}^k + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt}^k - C_t, \forall t$ ,
  5. 
$$v_{jt}^{2,k} = w_{jt} - \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt}^k, \forall j, t,$$
  6. 
$$s^k = \frac{\mu(UB - L(x^k, I^{+k}, I^{-k}, z^k, \pi^{1,k}, \pi^{2,k}))}{\|v^k\|^2}, \text{ onde } \mu \in (0, 2).$$
  7. Atualize  $\pi_t^{1,k+1} = \max(\pi_t^{1,k} + s^k v_t^{1,k}, 0), \forall t$ ,
  8.  $\pi_{it}^{2,k+1} = \pi_{it}^{2,k} + s^k v_{it}^{2,k}, \forall t$ ,
  9.  $k = k + 1$ .
  10. Se  $max\_it < k$  então PARE, senão volte ao Passo 3.
- 

Figura 4.2: Pseudocódigo do Algoritmo de Subgradiente para a formulação MTZ1S1MP - Dual Lagrangeano.

### 4.2.2 Algoritmo de Volume para resolução do Dual Lagrangeano

Apresentamos na Figura 4.3 o pseudocódigo do algoritmo de volume proposto para resolver o problema dual Lagrangeano associado à formulação MTZ1S1MP. Denotaremos o multiplicador  $\pi_t^1$  na iteração  $k$  por  $\pi_t^{1,k}$  e o multiplicador  $\pi_{j,t}^2$  na iteração  $k$  por  $\pi_{j,t}^{2,k}$ . Os centros de estabilizações

são denotados por  $\hat{\pi}^{1,k}$  e  $\hat{\pi}^{2,k}$ , onde  $k$  só é atualizado quando há uma melhoria no valor da função Lagrangeana.

---

### Algoritmo de Volume para o modelo MTZ1S1MP - Dual Lagrangeano

---

#### 1. Passo 0 - Início:

2. Sejam
3. -  $\bar{\pi}^{1,0}$  e  $\bar{\pi}^{2,0}$  vetores de multiplicadores de Lagrange;
4. -  $max\_iter = 2500$ ;
5. -  $\alpha = 0.5$ ;
6. -  $z_{LB}$  o limite inferior para o problema dual Lagrangeano (1.1) – (1.12);
7. -  $\xi_{pd} = 0.001$ ;
8. -  $\xi_v = 0.001$ .
9. - Resolva o subproblema Lagrangeano (4.8) – (4.18) (MTZ) com  $\pi^1 = \bar{\pi}^{1,0}$
10. e  $\pi^2 = \bar{\pi}^{2,0}$ , e tome  $\bar{I}^{+0}$ ,  $\bar{I}^{-0}$ ,  $\bar{x}^0$ ,  $\bar{w}^0$  e  $\bar{z}^0$  as soluções ótimas,
11. e  $\bar{\theta}_0 = L(\bar{I}^{+0}, \bar{I}^{-0}, \bar{x}^0, \bar{w}^0, \bar{z}^0)$  o valor ótimo.
12. - Calcule
13. 
$$\bar{v}_t^{1,0} = \sum_{j \in J} p_j \bar{x}_{jt}^0 + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} \bar{z}_{ijt}^0 - C_t, \quad t = 1, 2, \dots, T,$$
14. 
$$\bar{v}_{jt}^{2,0} = \bar{w}_{jt}^0 - \sum_{\substack{i=i_0 \\ i \neq j}}^J \bar{z}_{ijt}^0, \quad j = 1, 2, \dots, J, t = 1, 2, \dots, T,$$
15. - Inicialize  $z_{LB} = \bar{\theta}_0$ ,  $\widehat{I}^{+0} = \bar{I}^{+0}$ ,  $\widehat{I}^{-0} = \bar{I}^{-0}$ ,  $\widehat{x}^0 = \bar{x}^0$ ,  $\widehat{w}^0 = \bar{w}^0$
16. e  $\widehat{z}^0 = \bar{z}^0$ ,  $\widehat{v}^{1,0} = \bar{v}^{1,0}$ ,  $\widehat{v}^{2,0} = \bar{v}^{2,0}$ ,  $\widehat{\pi}^{1,0} = \bar{\pi}^{1,0}$ ,  $\widehat{\pi}^{2,0} = \bar{\pi}^{2,0}$ ,  $k = 0$  e  $\ell = 0$ .

#### 17. Passo 1:

18. - Calcule
19. 
$$s_\ell = \mu \frac{UB - \bar{\theta}_\ell}{\sum_{t=1}^T (\bar{v}_t^{1,0})^2 + \sum_{j=1}^J \sum_{t=1}^T (\bar{v}_{jt}^{2,0})^2},$$
20. onde  $\mu \in (0, 2)$  e  $UB$  é um limite superior para o problema (1.1) – (1.12).
21. - Tome
22. 
$$\bar{\pi}^{1,\ell+1} = \widehat{\pi}^{1,k} + s_\ell \widehat{v}^{1,\ell},$$
23. 
$$\bar{\pi}^{2,\ell+1} = \widehat{\pi}^{2,k} + s_\ell \widehat{v}^{2,\ell}.$$

#### 24. Passo 2:

25. - Resolva o subproblema Lagrangeano (4.8) – (4.18) (MTZ) com  $\pi^1 = \bar{\pi}^{1,\ell+1}$   
 26. e  $\pi^2 = \bar{\pi}^{2,\ell+1}$ , e tome  $\bar{I}^{+\ell+1}$ ,  $\bar{I}^{-\ell+1}$ ,  $\bar{x}^{\ell+1}$ ,  $\bar{w}^{\ell+1}$  e  $\bar{z}^{\ell+1}$  as soluções  
 27. ótimas, e  $\bar{\theta}_\ell = L(\bar{I}^{+\ell+1}, \bar{I}^{-\ell+1}, \bar{x}^{\ell+1}, \bar{w}^{\ell+1}, \bar{z}^{\ell+1})$  o valor ótimo.  
 28. **Passo 3:**

29. Atualize as variáveis primais:

$$\begin{aligned} 30. \quad & \hat{x}^{\ell+1} = \alpha \bar{x}^{\ell+1} + (1 - \alpha) \hat{x}^\ell, \\ 31. \quad & \hat{w}^{\ell+1} = \alpha \bar{w}^{\ell+1} + (1 - \alpha) \hat{w}^\ell, \\ 32. \quad & \hat{z}^{\ell+1} = \alpha \bar{z}^{\ell+1} + (1 - \alpha) \hat{z}^\ell, \\ 33. \quad & \widehat{I}^{+\ell+1} = \alpha \bar{I}^{+\ell+1} + (1 - \alpha) \widehat{I}^{+\ell}, \\ 34. \quad & \widehat{I}^{-\ell+1} = \alpha \bar{I}^{-\ell+1} + (1 - \alpha) \widehat{I}^{-\ell}, \end{aligned}$$

35. e o subgradiente

$$\begin{aligned} 36. \quad & \bar{v}_t^{1,\ell} = \sum_{j \in J} p_j \bar{x}_{jt}^\ell + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} \bar{z}_{ijt}^\ell - C_t, \quad t = 1, 2, \dots, T, \\ 37. \quad & \bar{v}_{jt}^{2,\ell} = \bar{w}_{jt}^\ell - \sum_{\substack{i=i_0 \\ i \neq j}}^J \bar{z}_{ijt}^\ell, \quad j = 1, 2, \dots, J, t = 1, 2, \dots, T, \end{aligned}$$

38. **Passo 4:**

39. - Verifique se  $\bar{\theta}_{\ell+1} > z_{LB}$ .  
 40. - Se for falso, declare uma iteração *vermelha*.  
 41. - Se for verdadeiro, calcule  $dir = \sum_{t=1}^T \hat{v}_t^{1,\ell+1} \bar{v}_t^{1,\ell+1} + \sum_{j=1}^J \sum_{t=1}^T \hat{v}_{jt}^{1,\ell+1} \bar{v}_{jt}^{1,\ell+1}$ .  
 42. Se  $dir < 0$  declare uma iteração *amarela*;  
 43. Se  $dir \geq 0$  declare uma iteração *verde* e atualize:  
 44. -  $z_{LB} = \bar{\theta}_{\ell+1}$ ,  
 45. -  $k = k + 1$ ,  
 46. -  $\hat{\pi}^{1,k} = \bar{\pi}^{1,\ell+1}$ ,  
 47. -  $\hat{\pi}^{2,k} = \bar{\pi}^{2,\ell+1}$ ,

48. **Passo 5 - Testes de parada:**

49. - 1. Se

$$\frac{\left| \sum_{j \in J} \sum_{t \in T} (h_j \widehat{I}_{jt}^{+\ell+1} + g_j \widehat{I}_{jt}^{-\ell+1}) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} \widehat{z}_{ijt}^{\ell+1} - z_{LB} \right|}{|z_{LB}|} < \xi_{pd},$$

51. e
52. 
$$\sqrt{\frac{\sum_{t=1}^T (\bar{v}_t^{1,\ell+1})^2 + \sum_{j=1}^J \sum_{t=1}^T (\bar{v}_{jt}^{2,\ell+1})^2}{T(J+1)}} < \xi_v,$$
53. então PARE.
54. - 2. Se
55. 
$$UB - z_{LB} < 1,$$
56. então PARE.
57. - 3. Se
58. 
$$\ell > max\_iter,$$
59. então PARE.
60. Senão, atualize  $\ell = \ell + 1$  e retorne ao **Passo 1**.
- 

Figura 4.3: Pseudocódigo do Algoritmo de Volume para a formulação MTZ1S1MP - Dual Lagrangeano.

Neste algoritmo, utilizamos uma atualização no fator de correção  $\mu$ , da seguinte maneira:

- $\mu = 2.1\mu$  a cada iteração verde;
- $\mu = 1.8\mu$  a cada 100 iterações amarelas;
- $\mu = 0.6\mu$  a cada 20 iterações vermelhas;

Os valores de  $max\_iter$  e  $\alpha$  foram dados empiricamente, assim como a quantidade de iterações verdes, amarelas e vermelhas.

Para obter o algoritmo associado à formulação MCF1S1MP basta substituir o problema a ser resolvido no Passo 0 para obtenção do  $z_{LB}$  pelo problema (1.1) – (1.7), (1.11), (1.12) e (1.13) – (1.16), e utilizar as restrições MCF no problema (4.8) – (4.18) (passos 0 e 2).

### 4.3 Relaxação Lagrangeana/Surrogate para o PIDS

Nesta seção iremos apresentar o subproblema Lagrangeano/Surrogate para o problema PIDS e o método de resolução desenvolvido para resolvê-lo.

Dualizando as restrições (1.17) com o multiplicador Surrogate  $\pi^2$  obtemos a restrição Surrogate

$$\sum_{j \in J} \sum_{t \in T} \pi_{jt}^2 \left( w_{jt} - \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right) = 0. \quad (4.19)$$

Dualizado a restrição Surrogate (4.19) com o multiplicador de Lagrange  $t$ , e as restrições (1.3) com o multiplicador  $\pi^1$ , obtemos o subproblema Lagrangeano/Surrogate (4.20)-(4.30):

$$\begin{aligned} \text{Min } \text{LS}(x, I^+, I^-, z, \pi^1, \pi^2) &= \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt} \\ &+ \sum_{t \in T} \pi_t^1 \left( \sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} - C_t \right) + \sum_{j \in J} \sum_{t \in T} t \pi_{jt}^2 \left( w_{jt} - \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right) \end{aligned} \quad (4.20)$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t \quad (4.21)$$

$$x_{jt} \leq C_t \frac{w_{jt}}{p_j}, \quad \forall j, \forall t \quad (4.22)$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t \quad (4.23)$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t \quad (4.24)$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t \quad (4.25)$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, \forall j, \forall t \quad (4.26)$$

$$\text{Restrições de eliminação de subrotas (MTZ ou MCF)} \quad (4.27)$$

$$I_{j0}^+ = 0, \quad \forall j \quad (4.28)$$

$$I_{j0}^- = 0, \quad \forall j \quad (4.29)$$

$$w_{jt} = 0/1, \quad \forall j, \forall t. \quad (4.30)$$

Da mesma forma que ocorre no subproblema Lagrangeano, o subproblema Lagrangeano/Surrogate também pode ser decomposto em dois: o problema de dimensionamento de lotes sem capacidade e o problema de sequenciamento.

Poderíamos ter optado em dualizar os dois conjuntos de restrições (1.3) e (1.17) via multiplicadores Surrogate e após, dualizar as restrições Surrogate via multiplicadores de Lagrange. Entretanto, testes preliminares indicaram que esta dualização fornece limitantes piores em relação à que propomos.

### 4.3.1 Algoritmo de Subgradiente para resolução do Dual Lagrangeano/Surrogate

Nesta subseção apresentamos na Figura 4.4 o pseudocódigo para resolver o problema dual Lagrangeano/Surrogate pelo Algoritmo de Subgradiente. Da mesma forma que no caso do dual Lagrangeano (subseção 4.2.1), denotamos o multiplicador  $\pi_t^1$  na iteração  $k$  por  $\pi_t^{1,k}$  e o multiplicador  $\pi_{j,t}^2$  na iteração  $k$  por  $\pi_{j,t}^{2,k}$ .

---

#### Algoritmo de Subgradiente - Formulação MTZ1S1MP - Dual Lagrangeano/Surrogate

---

1. Faça  $k = 0$ ,  $\pi_t^{1,0} = 0$ ,  $\pi_{it}^{2,0} = 0$ ,  $\forall i, t$ , e  $max\_it$  um limitante para a quantidade de iterações.
  2. Seja UB um limite superior para o problema (1.1) – (1.12).
  3. Resolva o subproblema (4.20) – (4.30) (MTZ) com  $\pi_t^1 = \pi_t^{1,k}$ ,  $\pi_{it}^2 = \pi_{it}^{2,k} \forall i, t$  e tome  $x_{jt}^k$ ,  $z_{ijt}^k$  e  $w_{jt}^k$  os minimizadores.
  4. Calcule  $v_t^{1,k} = \sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} - C_t, \forall t$ ,
  5. 
$$v_{jt}^{2,k} = w_{jt} - \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt}, \forall j, t,$$
  6. 
$$s^k = \frac{\mu(UB - L(x^k, I^{+k}, I^{-k}, z^k, \pi^{1,k}, \pi^{2,k}))}{\|v^k\|^2}, \text{ onde } \mu \in (0, 2).$$
  7. Atualize  $\pi_t^{1,k+1} = \max(\pi_t^{1,k} + s^k v_t^{1,k}, 0), \forall t$ ,
  8.  $\pi_{it}^{2,k+1} = \pi_{it}^{2,k} + s^k v_{it}^{2,k}, \forall t$ ,
  9.  $k = k + 1$ .
  10. Se  $max\_it < k$  então PARE, senão volte ao Passo 3.
- 

Figura 4.4: Pseudocódigo do Algoritmo de Subgradiente para a formulação MTZ1S1MP - Dual Lagrangeano/Surrogate.

A cada 10 iterações deste algoritmo, rodamos o algoritmo *Busca Unidimensional* apresentado na Figura 4.5 com  $s = 0.1$ , sendo que na primeira iteração o valor de  $t$  é 0. Estes parâmetros foram gerados de forma empírica.

Para obter o algoritmo associado à formulação MCF1S1MP basta substituir o problema a ser resolvido no Passo 2, pelo problema (1.1) – (1.7), (1.11), (1.12) e (1.13) – (1.16), e utilizar as restrições MCF no problema (4.20) – (4.30).



Busca Unidimensional	
1.	Tome:
2.	- $s$ : tamanho do passo,
3.	- $k_{max}$ : quantidade máxima de iterações,
4.	- $t_1 = -\infty$ ,
5.	- $t_2 = +\infty$ ,
6.	- $t = s$ ,
7.	- $\delta$ : a tolerância de parada,
8.	- $v$ : o melhor limitante inferior corrente,
9.	- $k = 0$ .
10.	Enquanto $k < k_{max}$ e $(t_2 - t_1) < \delta$ faça
11.	Atualize $k = k + 1$ ;
12.	Resolva o subproblema Lagrangeano/Surrogate (2.21);
13.	Se $v(RLS(t, \mu^*)) > v$ então $v = v(RLS(t, \mu^*))$ ;
15.	Calcule $\omega = \mu(A^2x - b^2)$ ;
16.	Se $\omega > 0$ então
17.	$t_1 = t$ ;
18.	$s = 2s$ ;
19.	$t = t + s$ ;
20.	Se $t \geq t_2$ então
21.	$t = t - s$ ;
22.	$s = (t_2 - t_1)/2$ ;
23.	$t = t + s$ ;
24.	Senão
25.	$t_2 = t$ ;
26.	$t = t - s$ ;
27.	Se $t_1 \neq -\infty$ então
28.	$s = (t_2 - t_1)/2$ ;
29.	$t = t + s$ .

Figura 4.5: Algoritmo de busca unidimensional para o dual Lagrangeano/Surrogate.

# Capítulo 5

## Estudo computacional

Neste capítulo apresentamos os resultados dos estudos computacionais desenvolvidos. O primeiro estudo teve por objetivo avaliar o desempenho dos modelos MTZ1S1M e MTZ1S1MP, e os modelos MCF1S1M e MCF1S1MP apresentados no Capítulo 1 (Estudo 1). A partir dos resultados obtidos no Estudo 1, elaboramos métodos para gerar limitantes para as formulações MTZ1S1MP e MCF1S1MP que forneceram os melhores resultados. O objetivo do Estudo 2, é testar o desempenho dos algoritmos de subgradiente e do volume usados para resolver os problemas dual Lagrangeano e dual Lagrangeano/Surrogate, ou seja, testar a qualidade dos limites duais. Os algoritmos foram escritos na sintaxe do AMPL [23] e utilizamos o *solver* CPLEX 12.6 [39] para resolver os problemas de otimização envolvidos. Os códigos foram compilados e executados em um computador com processador Intel Core i7 com 3.40 GHz e com 12 GB de memória RAM, sob a plataforma Windows 7.

### 5.1 Geração das Instâncias

Para a realização dos testes, usamos instâncias baseadas em parâmetros propostos por Almada-Lobo *et al* [1] e geradas por Maldonado [56]. Esses parâmetros são apresentados na Tabela 5.1. Foram geradas 3 classes de instâncias, descritas na Tabela 5.2, variando o número total de itens ( $J \in \{15, 25\}$ ), número total de períodos ( $T \in \{5, 10, 15\}$ ), e fixando os parâmetros envolvidos nos custos de *setup* ( $\mu = 100$ ) e na capacidade da máquina em cada período ( $Cut = 0.6$ ). Para cada classe foram testadas 5 instâncias totalizando 15 exemplares. Todos os exemplares gerados satisfazem a desigualdade triangular quanto ao custo de troca.

As instâncias foram nomeados como *dadosa – b*, onde o parâmetro *a* corresponderá à classe, ou seja  $a \in \{14, 18, 22\}$  e o parâmetro *b* corresponde ao exemplar da classe específica, que varia no conjunto  $\{0, 1, 2, 3, 4\}$ . Desta forma, a instância *dados14 – 3*, por exemplo, corresponde ao exemplar 3 da classe 14.

Parâmetros	Valores
Números de itens - $J$	$\{15, 25\}$
Número de períodos - $T$	$\{5, 10, 15\}$
Custo de estoque - $h_j$	$U[2, 9]$
Custo de atraso - $g_j$	$U[20, 50]$
Tempo de produção - $p_j$	1
Tempo de troca - $b_{ij}$	$U[5, 10]$
Custo de troca - $s_{ij}$	$\mu * b_{ij}$ , onde $\mu \in \{50, 100\}$
Demanda - $d_{jt}$	$U[40, 59]$
Capacidade - $C_t$	$Cut * \sum_{j \in J} d_{jt} * p_j$ , onde $Cut \in \{0.6, 0.8\}$

Tabela 5.1: Parâmetros usados na geração das instâncias.

Classes	J	T	$\mu$	$Cut$
Classe 14	25	5	100	0.6
Classe 18	25	10	100	0.6
Classe 22	25	15	100	0.6

Tabela 5.2: Classes de instâncias [56].

## 5.2 Comparação entre os modelos

O objetivo deste estudo é verificar a influência de explicitar a variável de preparo  $w_{jt}$  nos modelos descritos no Capítulo 1. O tempo de CPU foi fixado em 3.600 segundos (1 hora). As instâncias foram resolvidas pelo método “*branch-and-cut*” com o ajuste padrão do CPLEX (“*default*”) e pelo método “*branch-and-bound*” puro, isto é, foram desligados o pré-processamento e a possibilidade de geração de planos de corte. Na Tabela 5.3 apresentamos os resultados gerais para os modelos MTZ1S1M e MTZ1S1MP, e na Tabela 5.6 os resultados gerais para os modelos MCF1S1M e MCF1S1MP. As tabelas mostram, para cada instância, a melhor solução encontrada (Melhor Solução), o valor do GAP (fornecido pelo *solver*), o número total de planos de corte (Cortes), o número de nós (Nós), e os tempos de processamentos (Tempo) em segundos.

Comparando os modelos baseados nas restrições MTZ para eliminação de subsequências (Tabela 5.3), temos que para a classe com 5 períodos, classe 14, o tempo foi reduzido significativamente para instâncias do modelo MTZ1S1MP quando comparada ao modelo MTZ1S1M, sendo que no exemplar dados14 – 2 o tempo foi reduzido em 80,73%. Nas outras duas classes, o tempo foi o mesmo para os dois modelos. Entretanto, o GAP no modelo MTZ1S1MP foi reduzido 56,15% em média para a classe 18 e em 45,25% para a classe 22, quando comparado com o modelo MTZ1S1M. O número de planos de cortes incluídos na solução das instâncias

do modelo MTZ1S1M foi menor do que no modelo MTZ1S1MP. Uma dificuldade constatada nestas formulações foi a prova de otimalidade pelo *solver* CPLEX. Por exemplo, o exemplar dados18 – 0 no modelo MTZ1S1MP utilizou 15,7 horas de CPU para provar a otimalidade da solução que foi obtida no nó 1.639.946, sendo que teve que examinar 3.783.507 nós no método *branch-and-cut*. Outro exemplo é o exemplar dados18 – 1 que precisou de 492.359 segundos (136,76 horas) para provar a otimalidade.

	MTZ1S1M					MTZ1S1MP				
	Melhor Solução	GAP	Cortes	Nós	Tempo	Melhor Solução	GAP	Cortes	Nós	Tempo
dados14 – 0	253.251,8	0,00%	415	90.307	127,75	253.251,8	0,00%	476	19.692	29,98
dados14 – 1	248.704,6	0,00%	269	22.558	29,53	248.704,6	0,00%	271	9.175	10,67
dados14 – 2	252.954,0	0,00%	547	1.055.204	1.016,10	252.954,0	0,00%	515	149.438	195,78
dados14 – 3	256.831,0	0,00%	336	36.074	51,12	256.831,0	0,00%	400	16.203	25,15
dados14 – 4	251.074,6	0,00%	440	113.652	194,42	251.074,6	0,00%	376	56.128	110,64
dados18 – 0	851.764,8	0,33%	855	366.004	3.600	851.231,4	0,15%	1.094	210.807	3.600
dados18 – 1	839.804,4	0,45%	640	588.537	3.600	838.599,8	0,21%	904	236.998	3.600
dados18 – 2	854.595,6	0,42%	771	352.097	3.600	852.910,8	0,12%	1.164	233.458	3.600
dados18 – 3	861.431,8	0,35%	717	381.548	3.600	860.761,4	0,15%	1.064	218.748	3.600
dados18 – 4	846.103,0	0,48%	718	412.583	3.600	845.168,2	0,26%	1.163	226.401	3.600
dados22 – 0	1.801.290,2	0,83%	929	118.962	3.600	1.794.379,2	0,34%	1.480	129.580	3.600
dados22 – 1	1.772.231,2	0,56%	915	161.506	3.600	1.772.435,8	0,49%	1.431	155.061	3.600
dados22 – 2	1.799.402,8	0,56%	1.067	166.098	3.600	1.796.366,4	0,33%	1.522	132.132	3.600
dados22 – 3	1.811.639,8	0,77%	913	172.836	3.600	1.806.691,8	0,39%	1.345	144.862	3.600
dados22 – 4	1.783.279,2	0,86%	905	138.317	3.600	1.777.578,0	0,41%	1.384	122.503	3.600

Tabela 5.3: Resultados para as formulações MTZ1S1M e MTZ1S1MP (estratégia CPLEX).

Diversas classes de planos de corte podem ser usadas na resolução das instâncias (ver subseção 2.2.1). Os cortes gerados pelo *solver* CPLEX foram:

1. GUB Cover Cuts - *Generalized Upper-Bound* (GUB);
2. Clique Cuts (Clique);
3. Cover Cuts (Cover);
4. Implied Bound Cuts (Implied);
5. Inequações de Cobertura de Fluxo (Flow Cuts);
6. Mixed Integer Rounding (Mixed);
7. Inequações de Cobertura de Caminho de Fluxo (Flow Path);
8. Zero-Half Cuts (Zero);
9. Inequações de Gomory (Gomory);
10. Lift and Project (Lift).

Na Tabela 5.4 mostramos o número de planos de corte de cada tipo que foram utilizados na resolução das instâncias para os modelos MTZ1S1M e MTZ1S1MP. Os resultados para os modelos MCF1S1M e MCF1S1MP são exibidos na Tabela 5.7.

Observando a Tabela 5.4 vemos que os planos de corte do tipo GUB não tiveram muita utilização, sendo usados apenas nas instâncias dados14 – 1 e dados14 – 3 para a formulação MTZ1S1M, e na instância dados18–3 para a formulação MTZ1S1MP. Outro detalhe importante foi o plano de corte do tipo *Flow Path* que apenas na instância dados22 – 0, considerando a formulação MTZ1S1M, foi utilizado.

Quanto as formulações MCF1S1M e MCF1S1MP, observando a Tabela 5.7, podemos ver que o plano de corte do tipo *Flow Path* não teve nenhum efeito na formulação MCF1S1M. Observamos também que os planos de corte do tipo GUB, Clique, Cover e Zero não tiveram efeitos nas formulações com as restrições MCF.

	GUB		Clique		Cover		Implied		Flow Cuts		Mixed		Flow Path		Zero		Gomory		Lift	
	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2	M1	M2
dados14 – 0	-	-	2	6	23	16	28	100	35	56	202	211	-	26	20	16	23	38	82	7
dados14 – 1	2	-	3	4	3	3	8	62	34	32	146	141	-	18	4	1	22	5	47	5
dados14 – 2	-	-	3	3	54	46	52	47	44	-	274	264	-	22	10	8	24	19	86	55
dados14 – 3	1	-	3	5	14	20	47	62	36	52	155	198	-	18	5	4	19	10	56	31
dados14 – 4	-	-	-	-	46	30	64	19	46	51	192	170	-	29	2	12	21	17	69	48
dados18 – 0	-	-	1	4	8	78	29	142	113	109	515	540	-	76	28	27	38	19	123	99
dados18 – 1	-	-	2	14	25	68	28	77	71	95	395	463	-	66	10	3	26	19	83	99
dados18 – 2	-	-	1	9	9	102	41	130	138	122	461	572	-	80	7	5	36	16	78	128
dados18 – 3	-	1	1	18	12	99	28	134	95	85	427	542	-	79	19	12	27	19	108	75
dados18 – 4	-	-	-	3	6	54	34	37	85	207	447	550	-	112	2	-	43	104	101	96
dados22 – 0	-	-	-	13	18	30	56	57	140	256	-	784	578	163	20	17	17	112	100	48
dados22 – 1	-	-	1	14	11	22	51	43	128	259	599	739	-	165	11	5	31	92	83	92
dados22 – 2	-	-	-	17	9	18	53	64	156	235	716	822	-	163	5	7	41	95	87	101
dados22 – 3	-	-	2	23	5	20	39	43	109	240	615	657	-	172	20	16	26	100	97	74
dados22 – 4	-	-	-	5	3	9	41	60	95	250	656	706	-	190	-	-	34	88	76	76

Tabela 5.4: Planos de cortes gerados para instâncias dos modelos MTZ1S1M e MTZ1S1MP, aqui renomeadas por M1 e M2, respectivamente.

A Tabela 5.5 mostra o limitante superior (LS) obtido antes da ramificação da árvore pelo *branch-and-cut*, o valor do GAP do CPLEX e o tempo (Tempo) em segundos, necessário para encontrar este LS para as instâncias dos modelos MTZ1S1M e MTZ1S1MP. O valor de LS é obtido após a aplicação de planos de cortes e heurísticas no nó raiz. Na Tabela 5.8 são exibidos os resultados para as instâncias dos modelos MCF1S1M e MCF1S1MP.

De acordo com os resultados exibidos na Tabela 5.5, podemos ver que em pouco tempo, médias de 2,07 ( $\pm 0,17$ ), 6,55 ( $\pm 0,63$ ) e 15,26 ( $\pm 2,63$ ) segundos, para as classes 14, 18 e 22, respectivamente, o modelo MTZ1S1MP encontra uma boa solução factível, com GAPs médios de 1,32% ( $\pm 0,78\%$ ), 4,19% ( $\pm 2,09\%$ ) e 5,17% ( $\pm 0,48\%$ ), para as classes 14, 18 e 22, respectivamente. Os dois modelos apresentam GAPs e tempos de resolução similares em relação as classes 14 e 18. Quanto a classe 22, apesar dos tempos ainda serem similares, os GAPs são bem distintos, com médias de 5,17% ( $\pm 0,48\%$ ) e 11,38% ( $\pm 6,75\%$ ) para os modelos MTZ1S1MP e MTZ1S1M, respectivamente, sendo que por exemplo, o exemplar dados22 – 1, onde o GAP no modelo MTZ1S1M era de 19,29% diminuiu para 5,50% no modelo MTZ1S1MP, ou seja, decréscimo de 71,49%.

	MTZ1S1M			MTZ1S1MP		
	LS	GAP	Tempo	LS	GAP	Tempo
dados14 – 0	255.182,2	1,16%	2,43	258.912,4	2,44%	2,18
dados14 – 1	252.717,6	1,88%	1,76	248.931,8	0,28%	2,25
dados14 – 2	258.387,0	2,58%	1,62	254.841,4	1,09%	2,07
dados14 – 3	257.342,0	0,53%	1,64	259.476,6	1,32%	2,08
dados14 – 4	253.923,8	1,64%	1,62	254.001,6	1,47%	1,81
dados18 – 0	868.662,2	2,56%	7,05	870.580,4	2,56%	7,10
dados18 – 1	871.693,2	4,38%	7,16	861.901,2	3,10%	7,30
dados18 – 2	870.981,4	2,61%	6,86	886.977,2	4,15%	6,27
dados18 – 3	876.653,6	2,41%	6,29	887.227,8	3,36%	5,80
dados18 – 4	868.471,0	3,34%	7,35	883.467,2	7,80%	6,26
dados22 – 0	1.870.487,8	4,71%	18,19	1.874.462,8	4,75%	13,14
dados22 – 1	2.176.741,0	19,29%	16,96	1.864.023,4	5,50%	13,59
dados22 – 2	1.983.683,4	10,08%	20,03	1.899.281,2	5,82%	15,09
dados22 – 3	1.894.694,0	5,34%	16,60	1.893.163,2	5,08%	14,74
dados22 – 4	2.136.264,0	17,49%	19,06	1.855.420,0	4,72%	19,75

Tabela 5.5: Limitante superior obtido antes da ramificação da árvore - Modelo MTZ1S1MP.

Diferente do que observamos para o modelo com as restrições MTZ, o número total de planos de cortes aplicados na solução das instâncias do modelo MCF1S1MP foi superior ao número aplicado nas instâncias do modelo MCF1S1M, apesar de que para a classe 14 os valores



são similares (ver Tabelas 5.6 e 5.7). A quantidade de nós examinados também foi superior. Isto indica que o modelo MCF1S1MP teve um maior escopo para o método *branch-and-cut* do que o modelo MTZ1S1M. Observamos também, que na classe considerada difícil, a classe 22, o CPLEX analisou apenas o nó raiz, com excessão do exemplar dados22 – 3. Como o tempo limite de 1 hora foi utilizado em sua totalidade, com excessão das instâncias dados14 – 0 e dados14 – 1, podemos ver que o GAP associado teve uma melhora em média de 37,54% e 39,72%, para as classes 18 e 22, respectivamente com relação ao modelo MCF1S1MP. Com relação a classe 14, excluindo as instâncias dados14 – 1 e dados14 – 2, o ganho foi de 72,06% para o modelo MCF1S1MP.

	MCF1S1M					MCF1S1MP				
	UB	GAP	Cortes	Nós	Tempo	UB	GAP	Cortes	Nós	Tempo
hline dados14 – 0	253.656,6	0,44%	198	3.461	3.600	253.251,8	0,00%	143	20.653	3.295,38
dados14 – 1	248.704,6	0,02%	153	25.007	3.600	248.704,6	0,00%	130	12.237	284,78
dados14 – 2	253.870,2	0,70%	215	3.584	3.600	253.063,0	0,24%	160	9.606	3.600
dados14 – 3	257.982,6	0,70%	178	3.453	3.600	256.831,0	0,03%	155	11.036	3.600
dados14 – 4	252.080,6	0,65%	142	3.718	3.600	251.377,0	0,29%	137	3.898	3.600
dados18 – 0	858.222,6	1,34%	333	169	3.600	855.109,4	0,79%	632	202	3.600
dados18 – 1	843.710,4	1,18%	320	45	3.600	842.054,0	0,79%	630	161	3.600
dados18 – 2	857.436,0	1,01%	350	42	3.600	857.525,2	0,86%	667	128	3.600
dados18 – 3	868.917,0	1,58%	330	320	3.600	863.740,8	0,72%	648	133	3.600
dados18 – 4	853.962,0	1,71%	337	58	3.600	850.682,8	1,10%	602	275	3.600
dados22 – 0	1.807.934,2	1,48%	442	0	3.600	1.807.873,2	1,23%	802	0	3.600
dados22 – 1	1.793.037,8	2,09%	452	0	3.600	1.778.727,8	0,96%	794	0	3.600
dados22 – 2	1.817.366,4	1,89%	510	0	3.600	1.806.968,8	1,03%	875	0	3.600
dados22 – 3	1.818.786,4	1,40%	478	51	3.600	1.817.972,4	1,14%	819	0	3.600
dados22 – 4	1.796.195,6	1,85%	462	0	3.600	1.783.658,2	0,89%	775	0	3.600

Tabela 5.6: Resultados para as formulações MCF1S1M e MCF1S1MP (estratégia CPLEX).

	Implied		Flow Cuts		Mixed		Flow Path		Gomory		Lift	
	MCF	MCFP	MCF	MCFP	MCF	MCFP	MCF	MCFP	MCF	MCFP	MCF	MCFP
dados14 – 0	1	-	37	32	108	52	-	28	10	23	42	8
dados14 – 1	2	-	23	30	102	69	-	15	12	16	14	-
dados14 – 2	4	-	38	45	103	64	-	17	19	14	51	20
dados14 – 3	12	-	50	30	105	63	-	20	11	19	-	23
dados14 – 4	-	-	42	53	78	50	-	22	15	8	7	4
dados18 – 0	21	16	93	190	201	256	-	70	18	100	-	-
dados18 – 1	12	14	66	188	224	260	-	72	17	94	1	2
dados18 – 2	23	31	129	199	183	263	-	79	14	95	1	-
dados18 – 3	17	10	89	193	210	260	-	79	13	105	1	1
dados18 – 4	21	19	75	173	216	224	-	91	23	93	2	2
dados22 – 0	-	4	136	230	300	341	-	129	6	98	-	-
dados22 – 1	-	5	117	221	316	325	-	150	18	90	1	3
dados22 – 2	-	5	144	218	348	394	-	163	18	94	-	1
dados22 – 3	27	4	114	232	315	341	-	150	19	92	3	-
dados22 – 4	-	2	80	199	362	351	-	143	20	80	-	-

Tabela 5.7: Planos de cortes gerados para instâncias dos modelos MCF1S1M e MCF1S1MP, aqui renomeadas por MCF e MCFP, respectivamente.

Na Tabela 5.8 podemos ver que o tempo necessário para encontrar o LS associado às instâncias dos modelos MCF1S1M e MCF1S1MP é bem superior ao das formulações com restrições MTZ (ver Tabela 5.5). Para a classe 22, o CPLEX analisou apenas o nó raiz no tempo limitado. O valor especificado na Tabela 5.8 para esta classe corresponde a melhor solução encontrada neste tempo usando as heurísticas incluídas no CPLEX. Os GAPs relativos à formulação MCF1S1MP são ligeiramente melhores que os do modelo MCF1S1M, apresentando GAPs médios de 1,09% ( $\pm 0,2\%$ ), 1,49% ( $\pm 0,35$ ) e 1,86% ( $\pm 0,23\%$ ) (para o modelo MCF1S1M), para as classes 14, 18 e 22 respectivamente; e 0,88% ( $\pm 0,49\%$ ), 0,97% ( $\pm 0,17\%$ ) e 1,05% ( $\pm 0,13\%$ ) (para o modelo MCF1S1MP), para as classes 14, 18 e 22 respectivamente.

	MCF1S1M			MCF1S1MP		
	LS	GAP	Tempo	LS	GAP	Tempo
dados14 – 0	254.956,4	1,06%	161,97	255.484,0	1,10%	192,24
dados14 – 1	250.067,6	0,76%	120,01	248.704,6	0,21%	57,11
dados14 – 2	254.828,4	1,20%	193,67	253.562,4	0,57%	95,83
dados14 – 3	259.106,4	1,26%	130,31	258.907,2	1,08%	97,58
dados14 – 4	253.114,8	1,20%	96,24	254.105,8	1,47%	116,50
dados18 – 0	858.716,2	1,41%	1.648,15	855.889,8	0,89%	1.343,34
dados18 – 1	844.586,6	1,29%	1.544,29	842.235,2	0,82%	1.222,35
dados18 – 2	858.312,0	1,12%	2.335,10	857.525,2	0,86%	1.182,53
dados18 – 3	869.356,8	1,63%	937,86	867.841,2	1,20%	750,43
dados18 – 4	856.696,8	2,03%	1.041,92	850.791,2	1,12%	890,83
dados22 – 0	1.807.934,2	1,48%	3.600	1.807.873,2	1,23%	3.600
dados22 – 1	1.793.037,8	2,09%	3.600	1.778.727,8	0,96%	3.600
dados22 – 2	1.817.366,4	1,89%	3.600	1.806.968,8	1,03%	3.600
dados22 – 3	1.829.757,2	2,00%	2.693,39	1.817.972,4	1,14%	3.600
dados22 – 4	1.796.195,6	1,85%	3.600	1.783.658,2	0,89%	3.600

Tabela 5.8: Limitante superior obtido no nó raiz antes da ramificação da árvore - Modelo MCF1S1MP.

Na Tabela 5.9 apresentamos os resultados obtidos resolvendo as formulações pelo CPLEX desligando todo o seu pré-processamento e os planos de corte, ou seja, forçando que o CPLEX utilize o método *branch-and-bound* puro para resolução das instâncias. A Tabela 5.9 mostra os valores dos limites primais (LS) fornecidos, os GAPs e a quantidade de nós examinados, fornecidos pelo CPLEX. Nos campos onde aparece o símbolo “-” o CPLEX não conseguiu encontrar uma solução inteira no tempo fixado de 1 hora. Observamos que o modelo MCF1S1MP foi o que apresentou melhor desempenho, mostrando que sua formulação é a mais forte do que as demais formulações estudadas.

	MTZ1S1M			MTZ1S1MP			MCF1S1M			MCF1S1MP		
	LS	GAP	Nós	LS	GAP	Nós	LS	GAP	Nós	LS	GAP	Nós
dados14 – 0	-	-	13.073.380	261.027,8	18,55%	9.709.308	417.122,6	52,33%	8.267	262.746,4	22,81%	21.678
dados14 – 1	-	-	12.783.043	256.364,0	18,39%	9.433.145	310.824,6	37,24%	11.878	257.984,4	22,39%	26.677
dados14 – 2	-	-	12.741.254	259.786,2	19,27%	10.119.288	331.616,6	40,62%	10.691	262.174,6	22,59%	17.091
dados14 – 3	-	-	12.889.046	264.720,4	17,81%	9.474.184	353.092,4	42,76%	9.417	266.928,4	22,16%	27.514
dados14 – 4	-	-	13.077.505	257.016,6	17,71%	9.241.277	388.207,8	49,72%	7.598	259.190,4	21,48%	34.267
dados18 – 0	-	-	6.388.237	-	-	4.356.210	-	-	7.248	880.384,8	17,31%	19.129
dados18 – 1	-	-	6.329.740	-	-	4.302.624	-	-	5.945	864.593,0	16,88%	18.859
dados18 – 2	-	-	6.363.118	-	-	4.453.140	-	-	6.466	885.885,0	17,60%	18.699
dados18 – 3	-	-	6.396.990	-	-	4.509.277	-	-	6.612	892.149,2	17,38%	19.665
dados18 – 4	-	-	6.392.555	-	-	4.292.441	-	-	5.755	870.022,6	16,83%	23.811
dados22 – 0	-	-	4.121.944	-	-	2.768.819	-	-	6.469	1.851.319,4	14,61%	14.673
dados22 – 1	-	-	4.177.625	-	-	2.806.209	-	-	6.622	1.817.666,0	14,20%	14.601
dados22 – 2	-	-	4.171.063	-	-	2.784.734	-	-	7.336	1.853.585,4	14,75%	12.869
dados22 – 3	-	-	4.120.811	-	-	2.775.828	-	-	6.759	1.866.197,0	14,65%	15.700
dados22 – 4	-	-	4.212.425	-	-	2.813.876	-	-	6.666	1.828.417,4	14,33%	15.444

Tabela 5.9: Limitantes superiores obtidos do CPLEX (*Branch-and-bound* - 3.600 segundos).

### 5.3 Comparação entre os limites duais

Nesta seção, iremos comparar os limitantes obtidos através dos algoritmos de subgradiente e volume usados para resolver os problemas dual Lagrangeano e dual Lagrangeano/Surrogate apresentados nas seções 2.2.2 e 2.2.3, respectivamente. Considerando os resultados apresentados na seção 5.2, foram estudados apenas as instâncias dos modelos MTZ1S1MP e MCF1S1MP.

A fim de compararmos os métodos, lançamos mão de algumas medidas ( $GAP_H^S$ ,  $GAP_H^{SU}$ ,  $GAP_H^V$ ,  $GAP_C^{SU}$  e  $GAP_H^{RL}$ ), calculadas pelas fórmulas:

$GAP_H^S$ : (solução heurística - limite inferior dado pela solução do dual Lagrangeano pelo Método de Subgradiente)/solução heurística,

$GAP_H^{SU}$ : (solução heurística - limite inferior dado pela solução do dual Lagrangeano/Surrogate pelo Método de Subgradiente)/solução heurística,

$GAP_H^V$ : (solução heurística - limite inferior dado pela solução do dual Lagrangeano pelo Algoritmo de Volume)/solução heurística,

$GAP_C^{SU}$ : (melhor solução CPLEX - limite inferior dado pela solução do dual Lagrangeano/Surrogate pelo Método de Subgradiente)/melhor solução CPLEX,

$GAP_H^{RL}$ : (melhor solução CPLEX - limite inferior dado pela relaxação linear)/melhor solução CPLEX.

Os valores do  $GAP_{RL}$  foram calculados sem distinção dos modelos MTZ1S1MP e MCF1S1MP, pois em todos eles o valor da relaxação linear foi o mesmo. Nas Tabelas 5.10 e 5.11 descrevemos os resultados do Estudo 2 para o modelo MTZ1S1MP, e nas Tabelas 5.12 e 5.13 para o modelo MCF1S1MP. São exibidos os limitantes inferiores obtidos por todos os métodos objeto do nosso estudo (ver seções 4.2.1, 4.3.1 e 4.2.2), assim como os valores dos GAPs de cada método e a quantidade de iterações realizadas por cada método no tempo fixado. Nas Tabelas 5.10 e 5.12 fixamos o tempo em 600 segundos (10 minutos), e nas Tabelas 5.11 e 5.13 os resultados correspondem ao tempo limite de 3.600 segundos (1 hora).

Para as instâncias do modelo MTZ podemos observar que os valores dos limitantes obtidos em 10 minutos (Tabela 5.10) para o problema dual Lagrangeano resolvido pelo método de subgradientes não melhoraram quando o tempo de execução aumenta (Tabela 5.11). Quanto às soluções do problema dual Lagrangeano/Surrogate resolvido pelo método de subgradiente, aumentar o tempo de CPU permite em média uma melhora de 0,46%, 8,38% e 22,28%, para as classes 14, 18 e 22, respectivamente. Para as soluções do problema dual Lagrangeano resolvido pelo Algoritmo de Volume, tivemos uma melhora em média de 9,96%, 5,41% e 4,74% para as classes 14, 18 e 22, respectivamente.

A quantidade de iterações realizadas pelos algoritmo de subgradiente e algoritmo de volume para solucionar o dual Lagrangeano são similares, tanto utilizando 10 minutos de CPU quanto 1 hora. Já o dual Lagrangeano/Surrogate resolvido pelo método de subgradiente, a quantidade

de iterações é significativamente menor em relação aos outros dois métodos. Entretanto, cabe ressaltar que a cada 10 iterações do método do subgradiente rodamos 5 iterações do método de busca (ver Figura 4.5). Isto indica que o problema dual Lagrangeano/Surrogate é de maior dificuldade de resolução. No geral, considerando o tempo de 1 hora, o problema dual Lagrangeano/Surrogate fornece um melhor limitante inferior para o problema estudado, considerando o modelo MTZ1S1MP.

	Dual Lagrangeano - Subgradiente				Dual Lagrangeano - Volume				Dual Lag/Surrogate - Subgradiente		
	LI	$GAP_H^S$	Iterações		LI	$GAP_H^V$	Iterações		LI	$GAP_H^{SU}$	Iterações
dados14 - 0	198.299	31,35%	2.354		216.802	24,94%	2.362		250.830	13,16%	795
dados14 - 1	194.429	30,54%	2.300		213.021	23,89%	2.376		247.136	11,70%	901
dados14 - 2	196.302	28,65%	2.368		215.653	21,61%	2.430		251.017	8,76%	992
dados14 - 3	201.493	29,91%	2.305		220.485	23,31%	2.398		254.745	11,39%	794
dados14 - 4	194.934	30,31%	2.296		211.349	24,44%	2.392		249.851	10,68%	943
dados18 - 0	722.510	27,33%	1.460		736.858	25,88%	1.515		808.467	18,68%	649
dados18 - 1	710.904	28,51%	1.451		726.140	26,98%	1.493		652.239	34,41%	715
dados18 - 2	720.798	27,01%	1.430		738.413	25,23%	1.516		763.531	22,69%	720
dados18 - 3	730.610	27,60%	1.441		745.142	26,16%	1.505		829.541	17,80%	702
dados18 - 4	714.713	28,31%	1.435		724.274	27,35%	1.524		826.457	17,10%	741
dados22 - 0	1.576.420	28,30%	1.184		1.583.260	27,99%	1.312		1.163.710	47,07%	515
dados22 - 1	1.553.850	27,38%	1.158		1.538.190	28,11%	1.323		1.236.550	42,21%	577
dados22 - 2	1.572.800	25,74%	1.177		1.542.840	27,15%	1.309		1.660.900	21,58%	551
dados22 - 3	1.585.550	26,72%	1.167		1.541.010	28,77%	1.319		1.272.610	41,18%	579
dados22 - 4	1.557.350	27,35%	1.173		1.559.750	27,24%	1.336		1.572.450	26,65%	607

Tabela 5.10: Comparações entre os Métodos - Modelo MTZ1SIMP - 600 segundos.



	Dual Lagrangeano - Subgradiente				Dual Lagrangeano - Volume				Dual Lag/Surrogate - Subgradiente		
	LI	$GAP_H^S$	Iterações		LI	$GAP_H^V$	Iterações		LI	$GAP_H^{SU}$	Iterações
dados14 - 0	198.299	31,35%	14.228		242.101	16,19%	11.191		252.673	12,53%	4.645
dados14 - 1	194.429	30,54%	13.789		237.373	15,19%	11.263		248.188	11,33%	4.604
dados14 - 2	196.302	28,65%	14.243		236.361	14,08%	11.301		252.120	8,36%	5.138
dados14 - 3	201.493	29,91%	13.955		241.331	16,06%	11.290		256.103	10,92%	4.783
dados14 - 4	194.934	30,31%	13.679		239.273	14,46%	11.313		250.344	10,50%	4.965
dados18 - 0	722.510	27,33%	8.618		779.461	21,60%	7.566		848.573	14,64%	2.926
dados18 - 1	710.904	28,51%	8.538		766.422	22,93%	7.462		835.176	16,02%	3.024
dados18 - 2	720.798	27,01%	8.557		775.737	21,45%	7.565		850.375	13,89%	3.649
dados18 - 3	730.610	27,60%	8.482		789.006	21,82%	7.472		857.665	15,01%	3.374
dados18 - 4	714.713	28,31%	8.452		770.244	22,74%	7.553		841.517	15,59%	3.097
dados22 - 0	1.576.420	28,30%	7.133		1.642.490	25,30%	5.155		1.775.410	19,25%	2.181
dados22 - 1	1.553.850	27,38%	6.873		1.614.140	24,56%	5.022		1.759.510	17,77%	2.085
dados22 - 2	1.572.800	25,74%	7.007		1.630.410	23,02%	5.030		1.787.710	15,59%	2.140
dados22 - 3	1.585.550	26,72%	6.958		1.652.080	23,64%	5.009		1.796.240	16,98%	2.125
dados22 - 4	1.557.350	27,35%	6.948		1.612.940	24,76%	5.019		1.766.310	17,60%	2.179

Tabela 5.11: Comparações entre os Métodos - Modelo MTZ1S1MP - 3.600 segundos.

Podemos observar nas Tabelas 5.12 e 5.13 que os valores dos limitantes tem uma melhora mais gradativa quando comparado com o modelo MTZ1S1MP. Para o modelo MCF1S1MP, o problema dual Lagrangeano resolvido pelo Método de Subgradiente não estagnou como no modelo MTZ1S1MP. Ele apresentou em média uma melhora de 0,08%, 2,75% e 10,85% para as classes 14, 18 e 22, respectivamente, quando comparadas as soluções obtidas em 600 e em 3.600 segundos. Já o problema dual Lagrangeano/Surrogate resolvido pelo Método de Subgradiente apresentou em média melhora de 9,97%, 31,60% e 26,76% para as classes 14, 18 e 22, respectivamente, e o problema dual Lagrangeano resolvido pelo Algoritmo de Volume conseguiu melhorar as soluções em média em 36,93%, 37,21% e 27,04% para as classes 14, 18 e 22, respectivamente. Ambas as comparações foram feitas com soluções obtidas em 600 e 3.600 segundos.

Assim como no caso do modelo MTZ1S1MP, a quantidade de iterações são similares tanto para 10 minutos quanto para 1 hora em relação a solução do dual Lagrangeano resolvido pelo método do subgradiente e pelo algoritmo do volume. Já no caso do dual Lagrangeano/Surrogate resolvido pelo método de subgradiente, a quantidade de iterações continuou significativamente menor em relação aos outros dois métodos.

Observamos que em geral, para o caso do modelo MCF1S1MP, o problema dual Lagrangeano resolvido pelo Método de Subgradiente apresentou melhor desempenho que os demais métodos, perdendo apenas na classe considerada fácil, classe 14, onde o problema dual Lagrangeano/Surrogate resolvido pelo Método de Subgradiente forneceu melhores limitantes atingindo um  $GAP_H^{SU}$  médio de 12,56%.

No Apêndice B, apresentamos os gráficos de evolução dos iterados de um exemplar de cada uma das classes para cada um dos métodos objeto deste estudo no tempo fixado em 1 hora. Observamos que o Algoritmo de Subgradiente apresentou um comportamento desejado, de acordo com a literatura, ou seja, sua evolução é em “zigzag”. Já o Algoritmo de Volume apresentou uma evolução não-decrescente, como esperado, pois seu desenvolvimento foi feito considerando centros de estabilizações (seção 3.3).

Logo, podemos concluir que o problema dual Lagrangeano/Surrogate obtido a partir do modelo MTZ1S1MP resolvido pelo Algoritmo de Subgradiente foi o que obteve melhor desempenho, ou seja, gerou limitantes duais de melhor qualidade.

	Dual Lagrangeano - Subgradiente			Dual Lagrangeano - Volume			Dual Lag/Surrogate - Subgradiente		
	LI	$GAP_H^S$	Iterações	LI	$GAP_H^V$	Iterações	LI	$GAP_H^{SU}$	Iterações
dados14 – 0	198.085	31,42%	373	135.436	53,11%	404	217.526	24,69%	73
dados14 – 1	194.357	30,56%	373	126.960	54,64%	403	213.647	23,67%	100
dados14 – 2	196.036	28,74%	372	139.462	49,31%	416	233.451	15,14%	73
dados14 – 3	201.378	29,95%	372	134.532	53,21%	419	214.771	25,30%	79
dados14 – 4	194.764	30,37%	372	127.829	54,30%	403	230.983	17,42%	104
dados18 – 0	707.623	28,82%	152	378.624	61,92%	186	469.047	52,82%	50
dados18 – 1	703.764	29,23%	152	375.140	62,28%	186	445.784	55,17%	46
dados18 – 2	696.320	29,49%	152	369.930	62,54%	186	402.181	59,28%	37
dados18 – 3	704.211	30,22%	151	389.097	61,44%	186	454.266	54,99%	37
dados18 – 4	688.481	30,94%	151	372.156	62,67%	186	427.811	57,09%	48
dados22 – 0	1.361.570	38,07%	95	810.938	63,12%	126	476.883	78,31%	37
dados22 – 1	1.345.160	37,14%	95	790.376	63,06%	126	466.725	78,19%	46
dados22 – 2	1.408.540	33,49%	96	783.821	62,99%	126	685.949	67,61%	39
dados22 – 3	1.512.870	30,07%	95	807.452	62,68%	126	1.052.740	51,34%	34
dados22 – 4	1.366.120	36,27%	95	782.640	63,49%	126	703.903	67,16%	36

Tabela 5.12: Comparações entre os Métodos - Modelo MCF1S1MP- 600 segundos.

	Dual Lagrangeano - Subgradiente				Dual Lagrangeano - Volume			Dual Lag/Surrogate - Subgradiente		
	LI	$GAP_H^S$	Iterações		LI	$GAP_H^V$	Iterações	LI	$GAP_H^{SU}$	Iterações
dados14 - 0	198.299	31,35%	2.216		213.185	26,20%	1.979	249.257	13,71%	477
dados14 - 1	194.429	30,54%	2.229		208.654	25,45%	1.879	236.690	15,44%	458
dados14 - 2	196.302	28,65%	2.219		208.829	24,09%	1.742	250.704	8,87%	478
dados14 - 3	201.493	29,91%	2.222		215.027	25,21%	1.883	251.506	12,52%	451
dados14 - 4	194.934	30,31%	2.228		207.365	25,87%	1.918	245.442	12,25%	523
dados18 - 0	722.508	27,33%	914		645.847	35,04%	965	653.198	34,30%	227
dados18 - 1	710.901	28,52%	912		542.873	45,41%	961	586.525	41,02%	226
dados18 - 2	720.794	27,01%	911		605.613	38,68%	942	657.742	33,40%	189
dados18 - 3	730.605	27,60%	911		648.338	35,76%	966	658.288	34,77%	203
dados18 - 4	714.708	28,31%	910		571.468	42,67%	960	668.087	32,98%	199
dados22 - 0	1.575.832	28,33%	572		1.136.370	48,32%	613	881.931	59,89%	134
dados22 - 1	1.553.720	27,39%	571		1.122.440	47,54%	611	721.714	66,27%	172
dados22 - 2	1.572.600	25,75%	571		1.146.290	45,88%	610	979.157	53,77%	67
dados22 - 3	1.585.033	26,74%	571		1.128.010	47,86%	611	1.052.740	51,34%	72
dados22 - 4	1.556.568	27,39%	572		942.206	56,05%	541	909.185	57,59%	120

Tabela 5.13: Comparações entre os Métodos - Modelo MCF1S1MP - 3.600 segundos.

## 5.4 Considerações finais

No decorrer do nosso estudo observamos que a formulação MCF1S1MP é a formulação mais forte em relação as demais estudadas (MTZ1S1M, MTZ1S1MP e MCF1S1M), e que as formulações com restrições MTZ são as mais afetadas (influenciadas) pelos planos de cortes e pré-processamento do CPLEX (Tabela 5.9).

Em relação às relaxações estudadas (Relaxação Lagrangeana e Relaxação Lagrangeana/Surrogate) e os métodos propostos para solução dos problemas duais (Algoritmo de Subgradiente e Algoritmo de Volume), o Método mais eficaz foi o que forneceu limitantes duais através da resolução do problema dual Lagrangeano/Surrogate pelo Algoritmo de Subgradiente. Na Tabela 5.14 apresentamos os valores dos  $GAP_C^{SU}$  para as instâncias das classes 14, 18 e 22, para analisarmos o comportamento do método proposto com o método *branch-and-cut* incluído no solver *CPLEX*. Observamos que o dual Lagrangeano/Surrogate forneceu limitantes muito próximos aos fornecidos pelo CPLEX, todos no tempo limite de 3.600 segundos. Na Tabela 5.15 podemos ver os valores médios dos  $GAP_C^{SU}$  para cada classe com seus respectivos desvios padrões, mostrando que os valores dos  $GAP_C^{SU}$  não distanciam da média amostral.

Também podemos observar na Tabela 5.16, que os melhores limitantes duais obtidos são melhores que os obtidos pelo CPLEX através da relaxação linear.

	$GAP_C^{SU}$
dados14 – 0	0,23%
dados14 – 1	0,21%
dados14 – 2	0,33%
dados14 – 3	0,28%
dados14 – 4	0,29%
dados18 – 0	0,30%
dados18 – 1	0,41%
dados18 – 2	0,30%
dados18 – 3	0,36%
dados18 – 4	0,43%
dados22 – 0	1,06%
dados22 – 1	0,73%
dados22 – 2	0,48%
dados22 – 3	0,58%
dados22 – 4	0,63%

Tabela 5.14: Valores dos  $GAP_C^{SU}$  - Modelo MTZ1S1MP - 3.600 segundos.

	$GAP_C^{SU}$ médio
Classe 14	0,27% ( $\pm$ 0,049%)
Classe 18	0,36% ( $\pm$ 0,059%)
Classe 22	0,70% ( $\pm$ 0,221%)

Tabela 5.15: Valores médios dos  $GAP_C^{SU}$  e seus respectivos desvios padrões.

	$GAP_H^{SU}$	$GAP_H^{RL}$
dados14 – 0	12,53%	31,50%
dados14 – 1	11,33%	30,69%
dados14 – 2	8,36%	28,82%
dados14 – 3	10,92%	30,09%
dados14 – 4	10,50%	30,46%
dados18 – 0	14,64%	27,42%
dados18 – 1	16,02%	28,60%
dados18 – 2	13,89%	27,12%
dados18 – 3	15,01%	27,71%
dados18 – 4	15,59%	28,40%
dados22 – 0	19,25%	28,37%
dados22 – 1	17,77%	27,45%
dados22 – 2	15,59%	25,80%
dados22 – 3	16,98%	26,79%
dados22 – 4	17,60%	27,42%

Tabela 5.16: Comparações entre os limitantes inferiores obtidos pelo dual lagrangeano/surrogate e a relaxação linear.

# Capítulo 6

## Conclusões e perspectivas futuras

Neste trabalho, propomos três métodos, problema dual Lagrangeano resolvido pelo Algoritmo de Subgradiente, dual Lagrangeano resolvido pelo Algoritmo de Volume e dual Lagrangeano/Surrogate resolvido pelo Algoritmo de Subgradiente, para obter limites duais e uma heurística gulosa para derivar um limite primal para o problema integrado de dimensionamento de lotes e sequenciamento da produção (PIDS). Fizemos um estudo teórico dos métodos propostos e uma breve revisão bibliográfica do problema PIDS.

Um estudo preliminar foi feito para saber quais restrições seriam dualizadas e aquelas que apresentaram um melhor resultado foram as restrições de capacidade e a restrição de acoplamento dentre preparo e a troca dos itens. Essa dualização nos permitiu obter um subproblema que pode ser decomposto em dois, sendo um de dimensionamento de lotes e o outro de sequenciamento da produção. A partir desta dualização foram propostos os algoritmos de subgradiente e de volume para resolução dos duais Lagrangeanos e Lagrangeanos/Surrogate. Propomos estes métodos como alternativa ao CPLEX, num contexto onde não podemos utilizá-lo.

Para pesquisas futuras queremos entender o comportamento dos métodos propostos quando desligamos todo o pré-processamento e utilização de planos de cortes do CPLEX durante a resolução dos subproblemas, já que constatamos que todo esse ferramental do CPLEX influencia nos modelos. Também queremos analisar qual o impacto de utilizar os modelos desacoplados (pois quando dualizadas as restrições de capacidade e de ligação entre o preparo e a troca, conseguimos dois subproblemas) durante o evolução dos métodos propostos.

Outros temas interessantes são:

1. Investir em uma heurística Lagrangeana baseada na heurística de Trigeiro [76], rodar mais testes com instâncias maiores;
2. Analisar a eficiência dos métodos propostos, assim como melhoramentos destes, como o algoritmo de subgradiente projetado, em que toda solução dos problemas dual Lagrangeano e dual Lagrangeano/Surrogate são factíveis para o problema original [42], e o algoritmo

de volume revisado [8];

3. Considerando que o dual Lagrangeano/Surrogate foi o que forneceu melhores limitantes duais, seria interessante propor um método de resolução exato para resolvê-lo [47].
4. Continuar explorando a linha de problemas integrados, utilizando os métodos propostos para obter limitantes, como o problema integrado de dimensionamento de lotes e corte de estoque [81], [72]; o problema integrado de localização e roteamento [45], [48]; outras formulações para o problema integrado de dimensionamento e sequenciamento da produção [37], [56], [65], [70].
5. Considerar várias máquinas trabalhando em série e /ou paralelo no PIDS [17], [71];
6. Explorar outra abordagem para resolver os modelos propostos utilizando a ideia apresentada por Munari *et al* [32], ou seja, utilizar a técnica de geração de colunas com o método de pontos interiores, já que esta abordagem apresentou bom resultados para os problemas de corte de estoque e dimensionamento de lotes [83].



# Referências Bibliográficas

- [1] Almada-Lobo, B., Klabjan, D., Oliveira, J.F., Carravilla, M.A., *Single machine multi-product capacitated lot sizing with sequence-dependent setups*. International Journal of Production Research. v.45, n.20, p.4873 – 4894, 2007.
- [2] Araujo, S. A., *Métodos de Decomposição*, Notas de Aula, 2014.
- [3] Araujo, S. A., Arenales, M. N., Clark, A. R., *Joint rolling-horizon scheduling of materials processing and lot-sizing with sequence-dependent setups*, Journal of Heuristics, 13, 337 – 358, 2007.
- [4] Araujo, S. A., Arenales, M. N., Clark, A. R., *Lot sizing and furnace scheduling in small foundries*, Computers & Operations Research, 35, 916 – 932, 2008.
- [5] Araujo, S.A., Rangel, S., *Matemática Aplicada ao Planejamento da Produção e Logística*, Notas em Matemática Aplicada, v. 76, 2014.
- [6] Arenales, M., Armentano, V., Morabito, R. e Yanasse, H. Pesquisa Operacional, Ed. Campus, 2007.
- [7] Atamturk, A., Nemhauser, G.L., Savelsbergh, M.W.P., *Conflict Graphs in Integer Programming*, European Journal of Operations Research. v.121, p. 40 – 55, 2000.
- [8] Bahiense, L. O algoritmo do volume: convergência e resolução do problema de Steiner em grafos, UFRJ, 2000.
- [9] Balas, E., Ceria, S., Cornuejols, G., Natraj, N., *Gomory Cuts Revisited*, Operations Research Letters.v. 19, p. 1 – 10, 1996.
- [10] Bazaraa, M. S., Sherali, H. D., e Shetty, C. M., *Nonlinear Programming - Theory and Algorithms*, John Wiley & Sons, 3ª Edição, 2006.
- [11] Clark, A., Almada-Lobo, B., Almeder, C., *Lot sizing and scheduling: industrial extensions and research opportunities*, International Journal of Production Research, v.49, n.9, p. 2457 – 2461, 2011. ISSN 0020 – 7543.

- [12] Clark, A., Mahdieh, M., Rangel, S., *Production lot sizing and scheduling with non-triangular sequence-dependent setup times*, International Journal of Production Economics, ISSN 0925 – 5273, 2014.
- [13] Claus, A., *A new formulation for the travelling salesman problem*. SIAM Journal on Algebraic and Discrete Methods, SIAM, v. 5, n. 1, p. 21 – 25, 1984.
- [14] Dauzère-Pérès, S., Lasserre, B., *Integration of lot sizing and scheduling decisions in a job-shop*, European Journal of Operational Research, 75(2), 413 – 426, 1994.
- [15] Dauzère-Pérès, S., Lasserre, B., *On the importance of sequencing decisions in production planning and scheduling*, International Transactions in Operational Research, 9(6), 779 – 793, 2002.
- [16] Defalque, C. M., *Estratégias para incorporação das decisões de sequenciamento em um problema integrado de produção de bebidas*, Dissertação de Mestrado, UNESP/IBILCE, 2010.
- [17] Deleplanque, S., Kedad-Sidhoum, S., Quilliot, A., *Lagrangian Heuristic for a Multi-Plant Lot-Sizing Problem with Transfer and Storage Capacities*, RAIRO Operations Research, hal-00920396, version 1, 2013.
- [18] Drexel, A., Kimms, A., *Lot-sizing and scheduling - Survey and extensions*. European Journal Of Operational Research, 99(2), 221 – 235, 1997.
- [19] Fandel, G., Stammen-Hegene, C., *Simultaneous lot sizing and scheduling for multi-product multi-level production*. International Journal of Production Economics, 104(2), 308 – 316, 2006.
- [20] Ferreira, D., Morabito, R., Rangel, S., *Solution approaches for the soft drink integrated production lot sizing and scheduling problem*, European Journal of Operational Research, 196, 697 – 706, 2009.
- [21] F. Barahona e R. Anbil The volume algorithm: producing primal solutions with a sub-gradient method, Mathematical Programming, 87(3) : 385 – 389, 2000.
- [22] Fleischmann, B., Meyr, H., *The general lot-sizing and scheduling problem*. OR Spektrum, 19(1), 11 – 21, 1997.
- [23] Fourer, R., Gay, D. M., Kernighan, B. W., *AMPL - A Modeling Language for Mathematical Programming*, Second Edition, Cengage Learning, 2002.
- [24] Fukuda, E. H., Algoritmo do volume e otimização não diferenciável, IME-USP, 2007.

- [25] Galvão, R. D., Espejo, L. G., Boffey A. B., *A comparison of Lagrangean and Surrogate relaxations for the maximal covering location problem*, European Journal of Operational Research 124, (2000), 377 – 389.
- [26] Geoffrion, A. M., *Lagrangean relaxation for integer programming*, Mathematical Programming Study 2, 1974, 82 – 114.
- [27] Ginsberg, M. L., *Essentials of Artificial Intelligence*, São Francisco: Morgan Kaufmann, 1993.
- [28] Glover, F. *A multiphase dual algorithm for the zero-one integer programming problem*. Operations Research, 13, 879 – 919, 1965.
- [29] Glover, F. *Surrogate constraints*. Operations Research, 16(4) : 741 – 749, 1968.
- [30] Glover, F. *Surrogate Constraints Duality in Mathematical Programming*. Operations Research, 23, pp 434 – 451, 1975.
- [31] Gomory, R.E., *An Algorithm for the Mixed Integer Problem*, Technical Report RM- 2597, The RAND Corporation, 1960.
- [32] Gondzio, J., González-Brevis, P., Munari, P., *New developments in the primal-dual column generation technique*, European Journal of Operational Research, v. 224 (1), p. 41-51, 2013.
- [33] Greenberg, H. J., Pierskalla, W. P. *Surrogate Mathematical Programming*. Operations Research, 18, 924 – 939, 1970.
- [34] Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.P., *Lifted Cover Inequalities for 0-1 Integer Programs*, INFORMS Journal on Computing. v. 10, p. 417 – 426, 1998.
- [35] Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.P., *Lifted Flow Covers for Mixed 0-1 Integer Programs*, Mathematical Programming. v. 85, p. 439 – 467, 1999.
- [36] Hoffman, K, Padberg, M.W., *Improving Representations of Zero-One Linear Programs for Branchand- Cut*, ORSA Journal of Computing. v.3, p.121 – 134, 1991.
- [37] Hoto, R., Borssoi, A., Maculan, N., Fenato, A., *Reducing the Setup of a Tubettes Machine*, IEEE Latin America Transactions, v.8, n.1, março 2010.
- [38] *IBM Knowledge Center - Cuts*, [http://www-01.ibm.com/support/knowledgecenter/SSSA5P\\_12.6.0/ilog.odms.cplex.help/CPLEX/UsrMan/topics/discr\\_optim/mip/cuts/26\\_cuts\\_title\\_synopsis.html](http://www-01.ibm.com/support/knowledgecenter/SSSA5P_12.6.0/ilog.odms.cplex.help/CPLEX/UsrMan/topics/discr_optim/mip/cuts/26_cuts_title_synopsis.html). (última visita em 24/06/2015).
- [39] ILOG Using the CPLEX Callable Library, Copyright, ILOG, 2006.

- [40] Jans, R., Degraeve, Z., *Modeling Industrial Lot Sizing Problems: A Review*, International Journal of Production Research, 46(6), 1619 – 1643, 2008.
- [41] Izmailov, A.F., Solodov, M.V., *Otimização - Vol. 1. Condições de Otimizabilidade, Elementos de Análise Convexa e de Dualidade*, IMPA, Rio de Janeiro, 2005.
- [42] Izmailov, A.F., Solodov, M.V., *Otimização - Vol. 2. Métodos Computacionais*, IMPA, Rio de Janeiro, 2005.
- [43] James, R., Almada-Logo, B., *Single and parallel machine capacitated lot sizing and scheduling: New iterative MIP-based neighborhood search heuristics*. Computers and Operations Research, 38(12), 1816 – 1825, 2011.
- [44] Jans, R., Degraeve, Z., *Meta-Heuristics for Dynamic Lot Sizing: A Review and Comparison of Solution Approaches*, European Journal of Operational Research, 177, 1855–1875, 2007.
- [45] Jarboui, B., Derbel, H., Hanafi, S., Mladenovic, N., *Variable neighborhood search for location routing*. Computers and Operations Research, 40, 47-57, 2013.
- [46] Karwan, M. H., Rardin, R. L. *Some relationships between lagrangian and Surrogate duality in integer programming*. Mathematical Programming, 17, 320 – 334, 1979.
- [47] Kim, S.-L, Kim, S., Communicated by Luenberger, D.G., *Exact Algorithm for the Surrogate Dual of an Integer Programming Problem: Subgradient Method Approach*, Journal of Optimization Theory and Applications, Vol. 96(2), pp. 363 – 375, 1998.
- [48] Laporte, G., Norbert, Y., *An exact algorithm for minimizing routing and operating costs in depot location*, European Journal of Operational Research, 6, 224-226, 1981.
- [49] Larsen, J., *Set Partition and Applications*, Department of Management Engineering, Technical University of Denmark, Notas de Aula da Matéria Advanced Topics in Operations Research, 2010.
- [50] Lima, E. L., *Curso de Análise vol. 1*, Projeto Euclides, 2009.
- [51] Lorena, L. A. N., Plateau, G and Freville, A. *New subgradiente algorithms for the multiknapsack Lagrangean and Surrogate duals*. Pre-publication # 90. LIPN-Universite Paris Nord, 1990.
- [52] Lorena, L. A. N., Lopes, F. B. *A Surrogate heuristic for set covering problems*. European Journal of Operational Research. 79(1), 138 – 150, 1994.
- [53] Lorena, L. A. N., Narciso, M. G. *Relaxation heuristics for a generalized assignment problem*. European Journal of Operational Research, 91(1), 600 – 610, 1996.

- [54] Lorena, L.A.N., Senne E.L.F., *Stabilizing column generation using Lagrangean/Surrogate relaxation: an application to p-median location problems*. Submetido ao European Journal of Operational Research, 2001.
- [55] Maculan, F. N., Campello, R. E. *Algoritmos e Heurísticas: Desenvolvimento e Avaliação de Performance*. Niterói: Editora da UFF, 1994. v.1.
- [56] Maldonado, M. *Reformulações e Métodos de Solução para o Problema Integrado de Dimensionamento de Lotes e Sequenciamento da Produção*, Tese de Doutorado, UNESP/IBILCE, 2015.
- [57] Maldonado, M. ; Rangel, S. ; Clark, A., *Reformulation-linearization technique for modelling the scheduling decisions*, XVII Latin-Iberian-American Conference on Operations Research (CLAIO), 2014, Monterrey. Anais do CLAI/O/CSMIO 2014, v. unico. p. 1 – 1.
- [58] Maldonado, M.; Rangel, S., *Restrições do tipo Multi-Commodity Flow Aplicadas ao Problema Integrado de Dimensionamento e Sequenciamento da Produção*, XVI ONPCE - Oficina Nacional de Problemas de Corte, Empacotamento, Dimensionamento de Lotes e Programação da Produção, São Carlos, 2014.
- [59] Marchand, H, Wolsey, L.A., *Aggregation and Mixed Integer Rounding to Solve MIPs*, Operations Research. v.49, n. 3, p. 363 – 371, 2001.
- [60] Marchand, H., Martin, A., Weismantel, R., Wolsey, L., *Cutting planes in integer and mixed integer programming*, Discrete Applied Mathematics. v.123, p. 397 – 446, 2002.
- [61] Mateus, G. R., Ravetti, M. G., de Souza, M. C., Valeriano, T. M., *Capacitated lot sizing and sequence dependent setup scheduling: an iterative approach for integrtrion*, Journal of Scheduling, v. 13, 3, 245 – 259, 2010.
- [62] M. Guignard, *Lagrangean Relaxation*, Top, Volume 11, Number 2, 151 – 228, December 2003.
- [63] Miller, C. E., Tucker, A. W., Zemlin, R. A., *Integer programming formulation of traveling salesman problems*. J. ACM, ACM, New York, NY, USA, v. 7, n. 4, p. 326 – 329, out. 1960. ISSN 0004 – 5411.
- [64] Narciso, M. G., *A relaxação Lagrangeana/Surrogate e algumas aplicações em otimização combinatória*. Tese de Doutorado em Computação Aplicada, INPE, 1998.
- [65] Noon, C. E., Bean, J. C., *A lagrangian based approach to the asymmetric generalized traveling salesman problem*, Technical Report, 88-17, 1988.

- [66] Oncan, T., Altinel, K. and Laporte, G., *A comparative analysis of several asymmetric traveling salesman problem formulations*, Computers and Operations Research 36(3) : 637 – 654, 2009.
- [67] Padberg, M.W., Van Roy, T.J., Wolsey, L.A., *Valid Inequalities for Fixed Charge Problems*, Operations Research v. 33, p.842 – 861, 1985.
- [68] Parker, R. G., Rardin, L. R. *Discrete Optimization*. Academic Press, INC, London, 1988.
- [69] Pochet, Y., Wolsey, L.A., *Production planning by mixed integer programming*, New York: Springer Verlag, 2006. (Series in Operations Research and Financial Engineering).
- [70] Pureza, V., Morabito, R., Luna, H.P., *Modeling and Solving the Traveling Salesman Problem with Priority Prizes*, Relatório Técnico UFSCar, 2014 (submetido para publicação).
- [71] Sambasian, M., Yahya, S., *A Lagrangean-based heuristic for multi-plant, multi-item, multi-period capacitated lot-sizing problems with inter-plant transfers*, Computers and Operations Research, 32, 537-555, 2005.
- [72] Silva, E., Alvelos, F., Valério de Carvalho, J. M., *Integrating two-dimensional cutting stock and lot-sizing problems*, Journal of Operational Research Society, 65, 108-123, 2014.
- [73] Silver, E. A., Meal, H. A., *A heuristic for selecting lot size requirements for the case of a deterministic time varying demand rate and discrete opportunities for replenishment*, Production and Inventory Management Journal, 14(2), 64 – 74, 1973.
- [74] Sikora, R., Chhajed, D., Shaw, M., *Integrating the lot-sizing and sequencing decisions for scheduling a capacitated flow line*, Computers & Industrial Engineering, 30(4), 659 – 679, 1996.
- [75] Takahashi, R. H. C., *Notas de aula. Otimização Escalar e Vetorial. Volume 2: Otimização Escalar*, UFMG, 2007.
- [76] Trigeiro, W.W.; Thomas, L.J., McClain, J.O., *Capacitated Lot Sizing With Setup Times*, Management Science, 35(3), 353 – 366, 1989.
- [77] Toledo, C. F. M., França, P. M., Morabito, R., Kimms, A., *Multi-population genetic algorithm to solve the synchronized and integrated two-level lot sizing and scheduling problem*, International Journal of Production Research, 47, 3097 – 3119, 2009.
- [78] Toso, E. A. V., Morabito, R., Clark, A. R., *Lot sizing and sequencing optimisation at an animal-feed plant*, Computers & Industrial Engineering, 57(3), 813 – 821, 2009.

- [79] Valério de Carvalho, J. M., *Exact solution of bin-packing problems using column generation and branch-and-bound*, Annals of Operations Research, 86, 629-659, 1999.
- [80] Valério de Carvalho, J. M., *LP models for bin packing and cutting stock problems*, European Journal of Operational Research, 144, 253-273, 2002.
- [81] Vanzela, M., Rangel, S., Araujo, S.A., *The Integrated Lot Sizing and Cutting Stock Problem in a Furniture Factory*, In: 11th IFAC Workshop on Intelligent Manufacturing Systems, 2013, 2013, São Paulo. Intelligent Manufacturing Systems, 2013. v. 11. p. 390-395.
- [82] Van Roy, T.J., Wolsey, L.A., *Valid Inequalities and Separation for Uncapacitated Fixed Charge Networks*, Operations Research Letters, v.4, p. 105 – 112, 1985.
- [83] Wright, S.J., *Primal-Dual Interior-Point Methods*, SIAM, ISBN: 978-0-89871-382-4, 1997.
- [84] Wolosewicz, C., Dautère-Pérès, S., Aggoune, R., *A Lagrangian heuristic for an integrated lot-sizing and fixed scheduling problem*, European Journal of Operational Research 000(2015)1 – 10.
- [85] Wolsey, L. A., *Integer Programming*, John Wiley & Sons, 1998.
- [86] Wolsey, L.A., *Strong formulations for mixed integer programs: valid inequalities and extended formulations*, Mathematical programming, Series B. v. 423 – 447, 2003.
- [87] Zhu, X., Wilhelm, W. E., *Scheduling and lot sizing with sequence-dependent setup: A literature review*, IIE Transactions, 38, 987 – 1007, 2006.

# Apêndice A

## Modelos para o PIDS

### A.1 Modelo MTZ1S1M (ver descrição na seção 1.2)

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt}$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t$$

$$x_{jt} \leq \frac{C_t}{p_j} \left( \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right), \quad \forall j, \forall t$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t$$

$$u_{jt} \geq u_{it} + 1 - (J)(1 - z_{ijt}), \quad \forall i, \forall j, i \neq j, \forall t$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t$$

$$1 \leq u_{jt} \leq J \quad \forall j, t$$

$$I_{j0}^+ = 0, \quad \forall j$$

$$I_{j0}^- = 0, \quad \forall j$$



## A.2 Modelo MTZ1S1MP (ver descrição na seção 1.2)

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt}$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t$$

$$p_j x_{jt} - C_t w_{jt} \leq 0, \quad \forall j, \forall t$$

$$w_{jt} = \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t$$

$$u_{jt} \geq u_{it} + 1 - (J)(1 - z_{ijt}), \quad \forall i, \forall j, i \neq j, \forall t$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t$$

$$1 \leq u_{jt} \leq J \quad \forall j, t$$

$$w_{jt} = 0/1, \quad \forall j, \forall t,$$

$$I_{j0}^+ = 0, \quad \forall j$$

$$I_{j0}^- = 0, \quad \forall j$$

### A.3 Modelo MCF1S1M (ver descrição na seção 1.2)

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt}$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t$$

$$x_{jt} \leq \frac{C_t}{p_j} \left( \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt} \right), \quad \forall j, \forall t$$

$$\sum_{j \in J} z_{i_0 j t} \geq \sum_{i=i_0}^J z_{i k t}, \quad \forall k \in J; k \neq i, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{i k t} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{k j t}, \quad \forall k \in J, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{i j t} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t$$

$$\sum_{j \in J} m_{r i_0 j t} - \sum_{j \in J} m_{r j i_0 t} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{j r t}, \quad \forall r, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq r}}^J m_{r j r t} - \sum_{\substack{j=i_0 \\ j \neq r}}^J m_{r r j t} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{j r t}, \quad \forall r, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq j}}^J m_{r i j t} = \sum_{\substack{i=i_0 \\ i \neq j}}^J m_{r j i t}, \quad \forall r, \forall t; j \neq r, \forall t,$$

$$0 \leq m_{r i j t} \leq z_{i j t}, \quad \forall i, j = i_0, 1, \dots, J, \forall r, \forall t,$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t$$

$$m_{r i j t} = 0/1, \quad \forall r, \forall i, \forall j, \forall t,$$

$$I_{j0}^+ = 0, \quad \forall j$$

$$I_{j0}^- = 0, \quad \forall j$$

## A.4 Modelo MCF1S1MP (ver descrição na seção 1.2)

$$\text{Min } Z = \sum_{j \in J} \sum_{t \in T} (h_j I_{jt}^+ + g_j I_{jt}^-) + \sum_{t \in T} \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} s_{ij} z_{ijt}$$

sujeito a:

$$I_{j(t-1)}^+ + I_{jt}^- + x_{jt} - I_{jt}^+ - I_{j(t-1)}^- = d_{jt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} p_j x_{jt} + \sum_{i \in J} \sum_{\substack{j \in J \\ j \neq i}} b_{ij} z_{ijt} \leq C_t, \quad \forall t$$

$$p_j x_{jt} - C_t w_{jt} \leq 0, \quad \forall j, \forall t$$

$$w_{jt} = \sum_{\substack{i=i_0 \\ i \neq j}}^J z_{ijt}, \quad \forall j, \forall t$$

$$\sum_{j \in J} z_{i_0 jt} \geq \sum_{i=i_0}^J z_{ikt}, \quad \forall k \in J; k \neq i, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq k}}^J z_{ikt} = \sum_{\substack{j=i_0 \\ j \neq k}}^J z_{kjt}, \quad \forall k \in J, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq i}}^J z_{ijt} \leq 1, \quad \forall i = i_0, 1, \dots, J, \forall t$$

$$\sum_{j \in J} m_{ri_0 jt} - \sum_{j \in J} m_{rji_0 t} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{jrt}, \quad \forall r, \forall t$$

$$\sum_{\substack{j=i_0 \\ j \neq r}}^J m_{rjrt} - \sum_{\substack{j=i_0 \\ j \neq r}}^J m_{rrjt} = \sum_{\substack{j=i_0 \\ j \neq r}}^J z_{jrt}, \quad \forall r, \forall t$$

$$\sum_{\substack{i=i_0 \\ i \neq j}}^J m_{rijt} = \sum_{\substack{i=i_0 \\ i \neq j}}^J m_{rjit}, \quad \forall r, \forall t; j \neq r, \forall t,$$

$$0 \leq m_{rijt} \leq z_{ijt}, \quad \forall i, j = i_0, 1, \dots, J, \forall r, \forall t,$$

$$x_{jt}, I_{jt}^+, I_{jt}^- \geq 0, z_{ijt} = 0/1, \quad \forall i, j, \forall t$$

$$m_{rijt} = 0/1, \quad \forall r, \forall i, \forall j, \forall t,$$

$$w_{jt} = 0/1, \quad \forall j, \forall t,$$

$$I_{j0}^+ = 0, \quad \forall j$$

$$I_{j0}^- = 0, \quad \forall j$$

# Apêndice B

## Gráficos - instâncias dados14 – 0

Nas Figuras B.1 a B.6 apresentamos os gráficos de evolução dos iterados da instância dados14–0 considerando cada um dos métodos objeto deste estudo no tempo fixado em 1 hora.

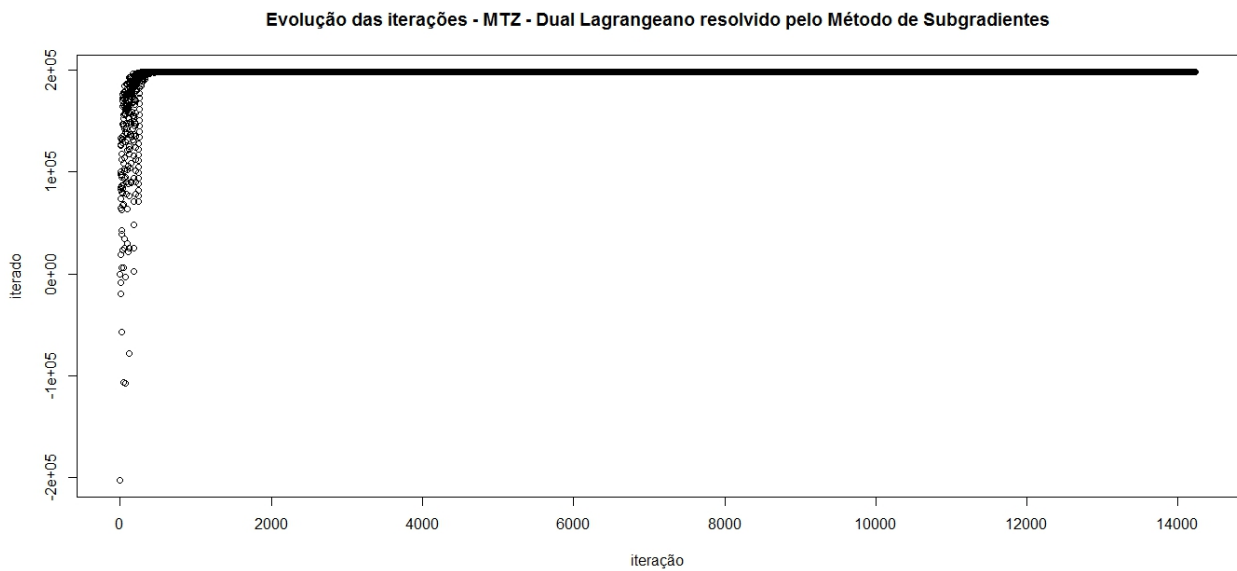


Figura B.1: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MTZ1S1MP.

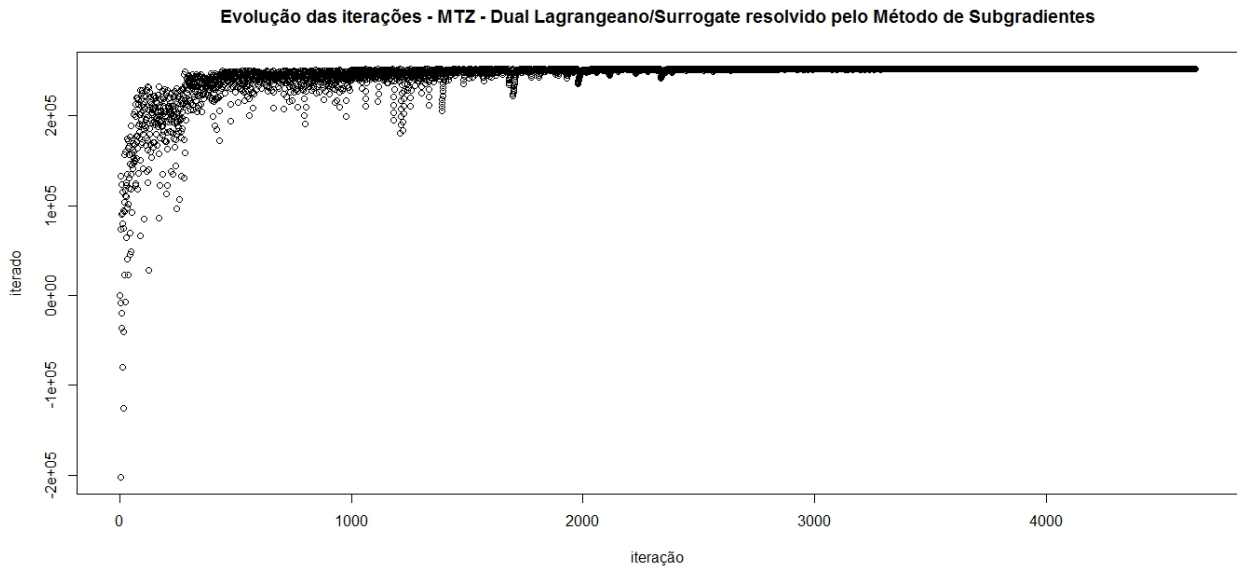


Figura B.2: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MTZ1S1MP.

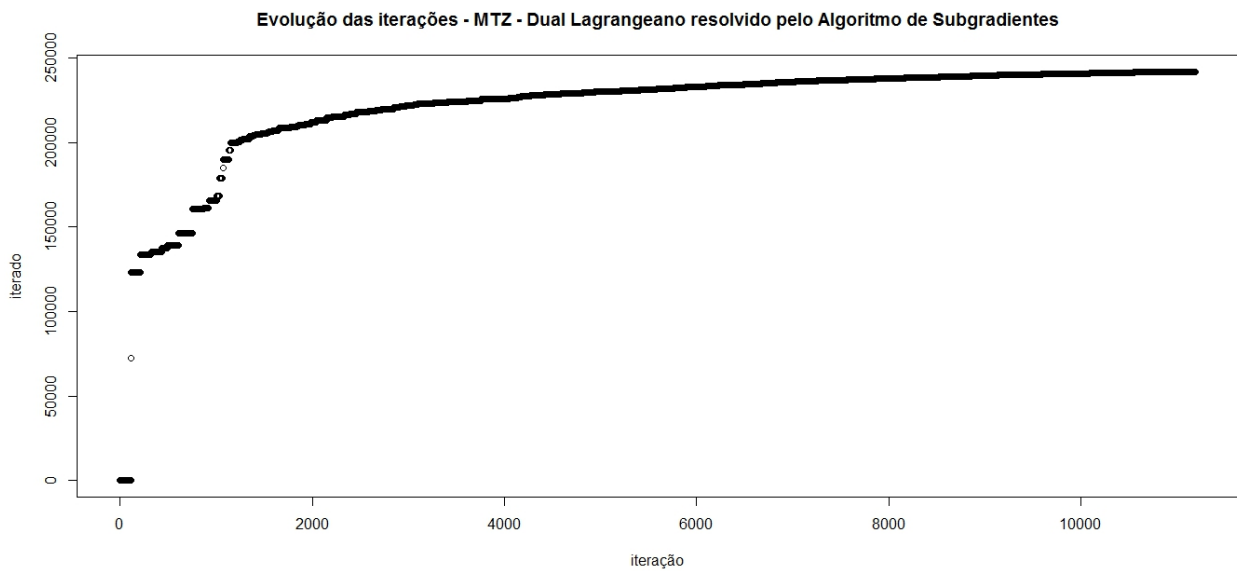


Figura B.3: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados14-0 - Modelo MTZ1S1MP.

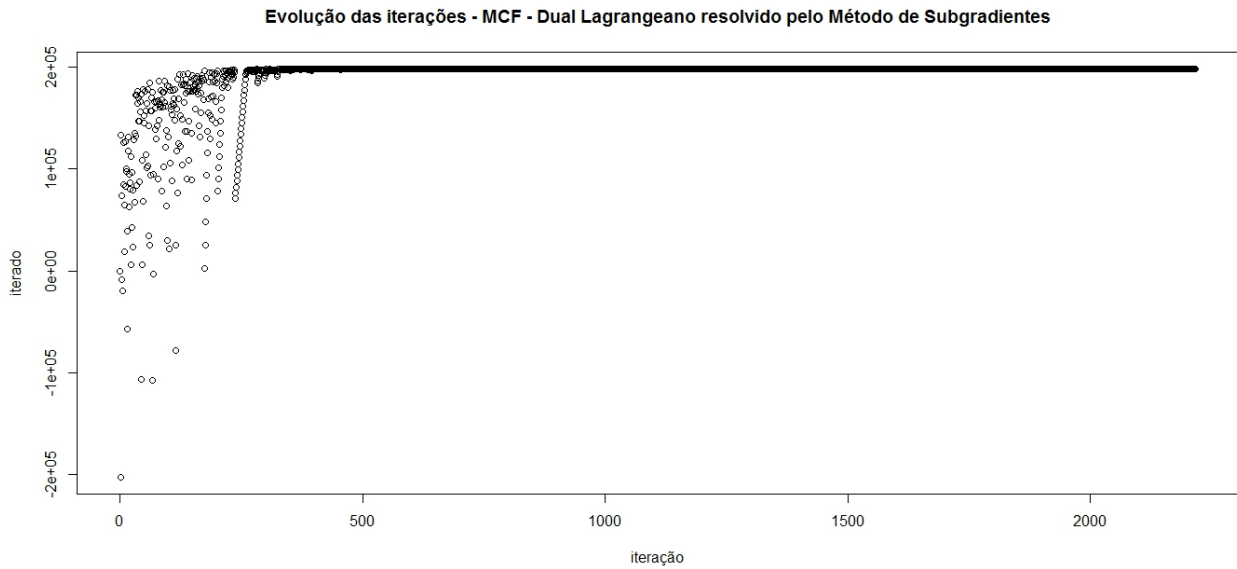


Figura B.4: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MCF1S1MP.

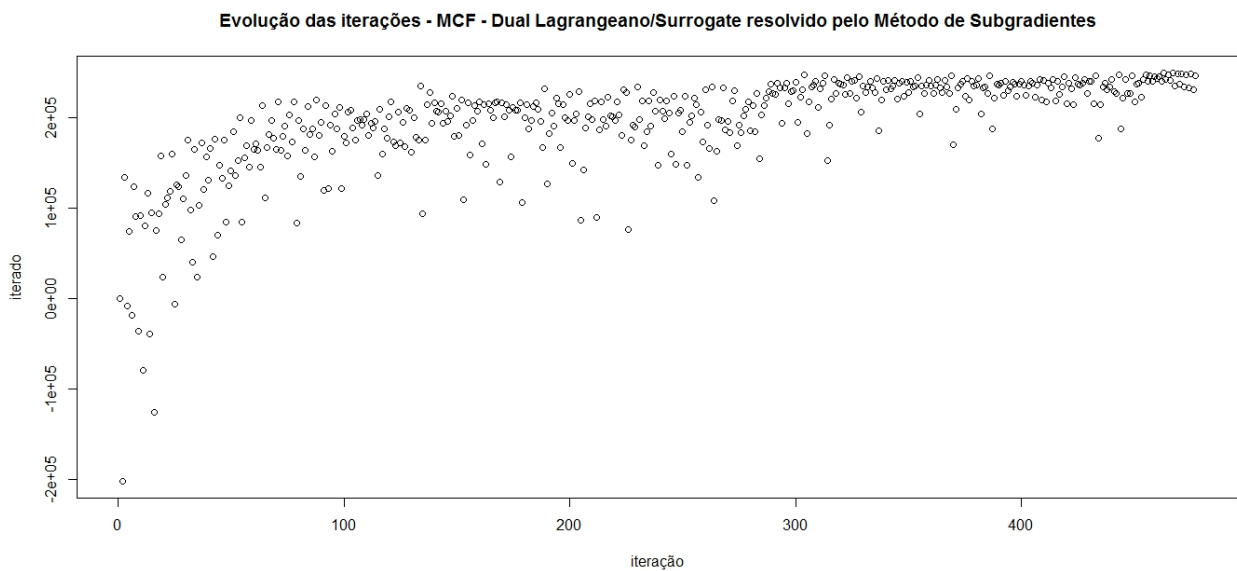


Figura B.5: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados14-0 - Modelo MCF1S1MP.

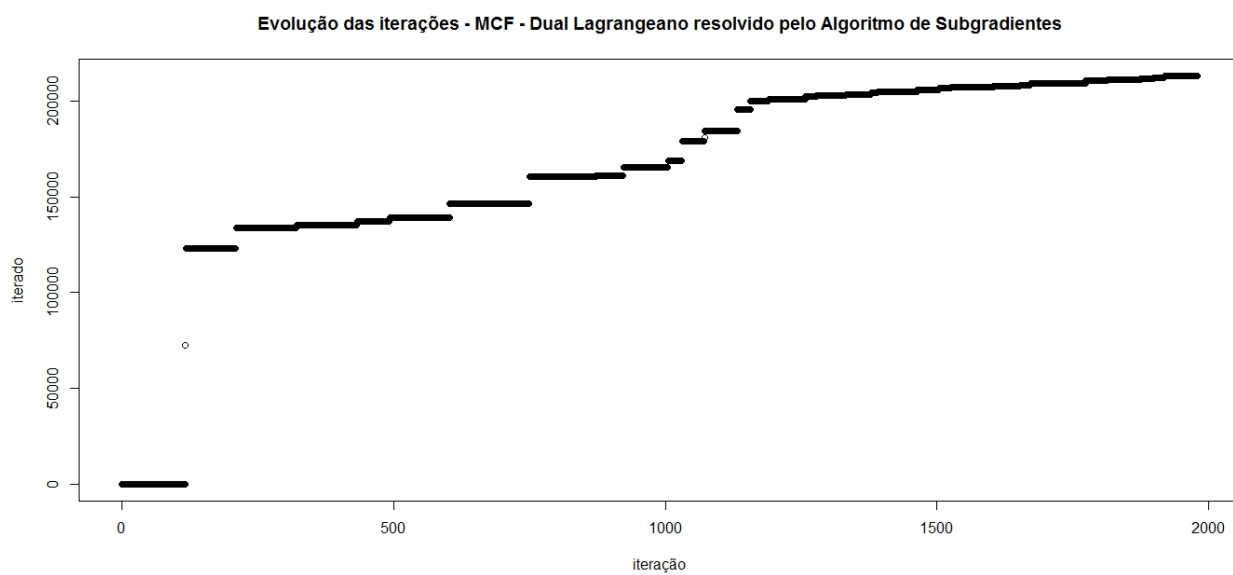


Figura B.6: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados14-0 - Modelo MCF1S1MP.

# Apêndice C

## Gráficos - instâncias dados18 – 0

Nas Figuras C.1 a C.6 apresentamos os gráficos de evolução dos iterados da instância dados18–0 considerando cada um dos métodos objeto deste estudo no tempo fixado em 1 hora.

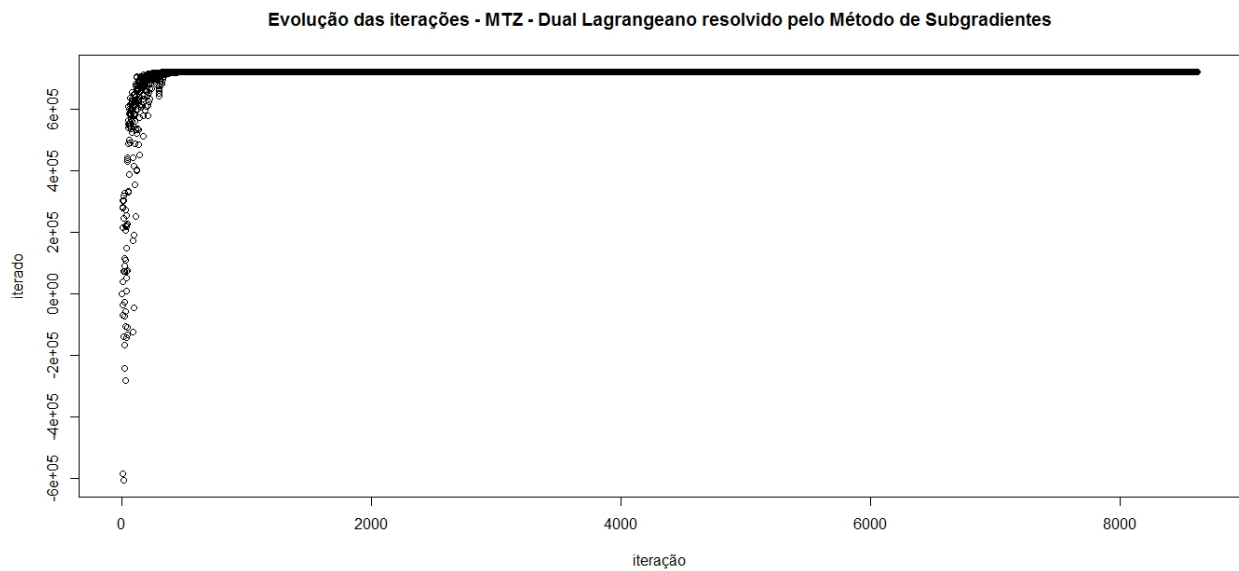


Figura C.1: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MTZ1S1MP.



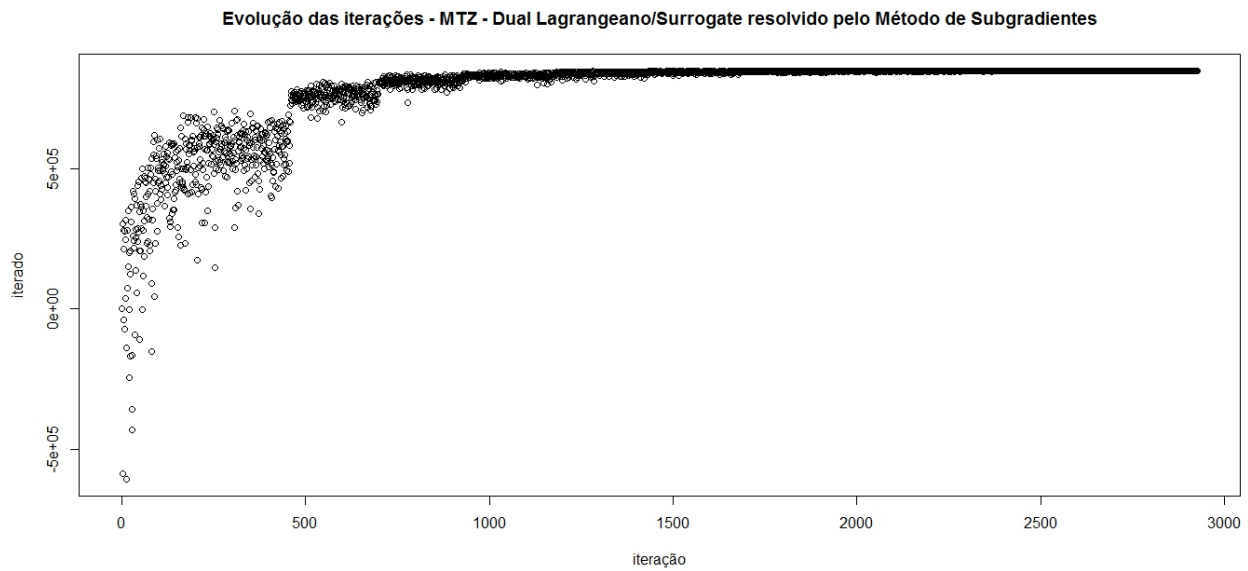


Figura C.2: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MTZ1S1MP.

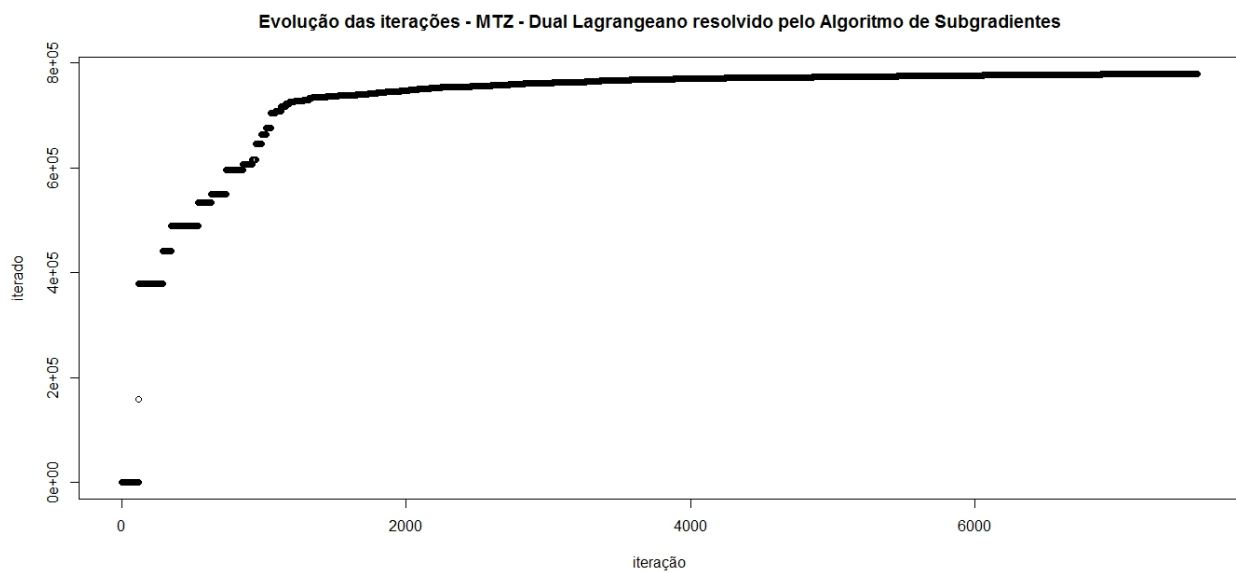


Figura C.3: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados18-0 - Modelo MTZ1S1MP.

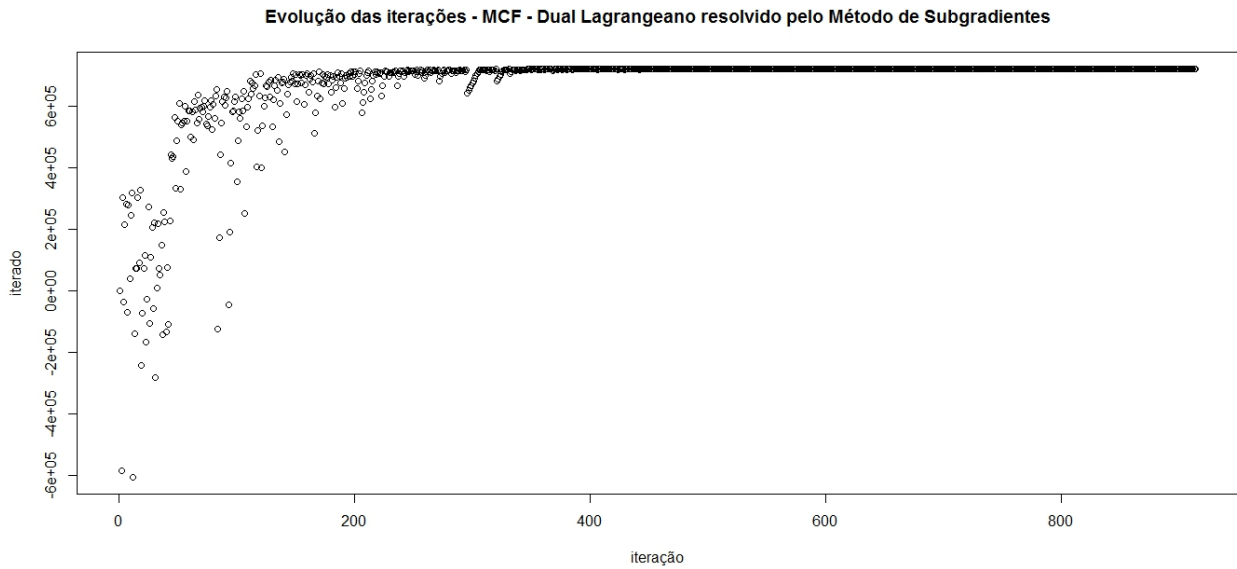


Figura C.4: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MCF1S1MP.

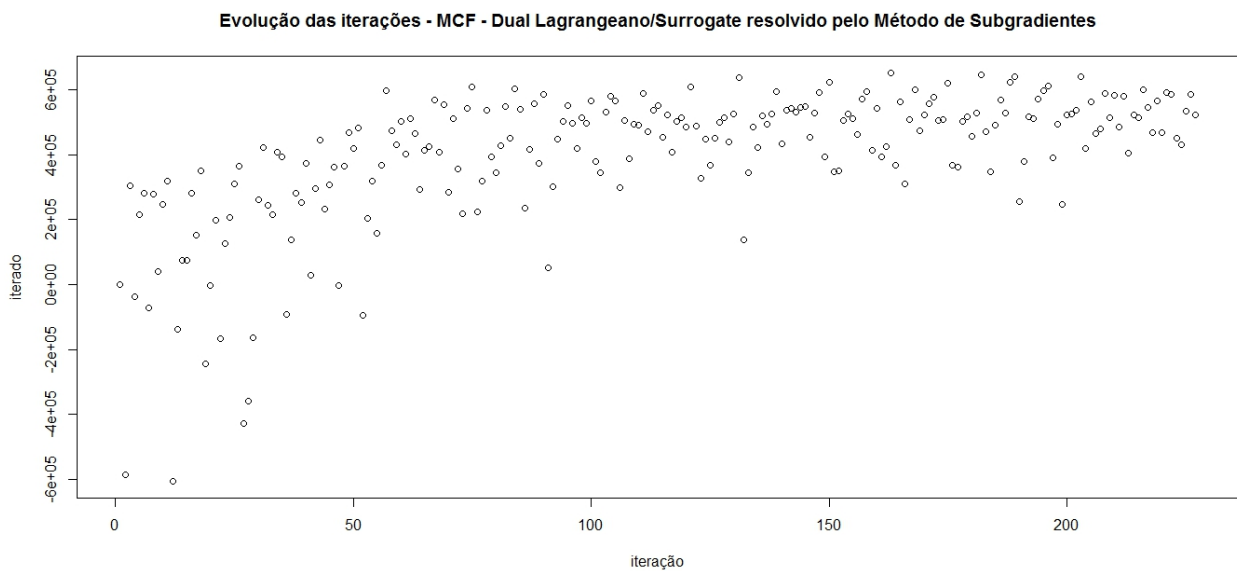


Figura C.5: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados18-0 - Modelo MCF1S1MP.

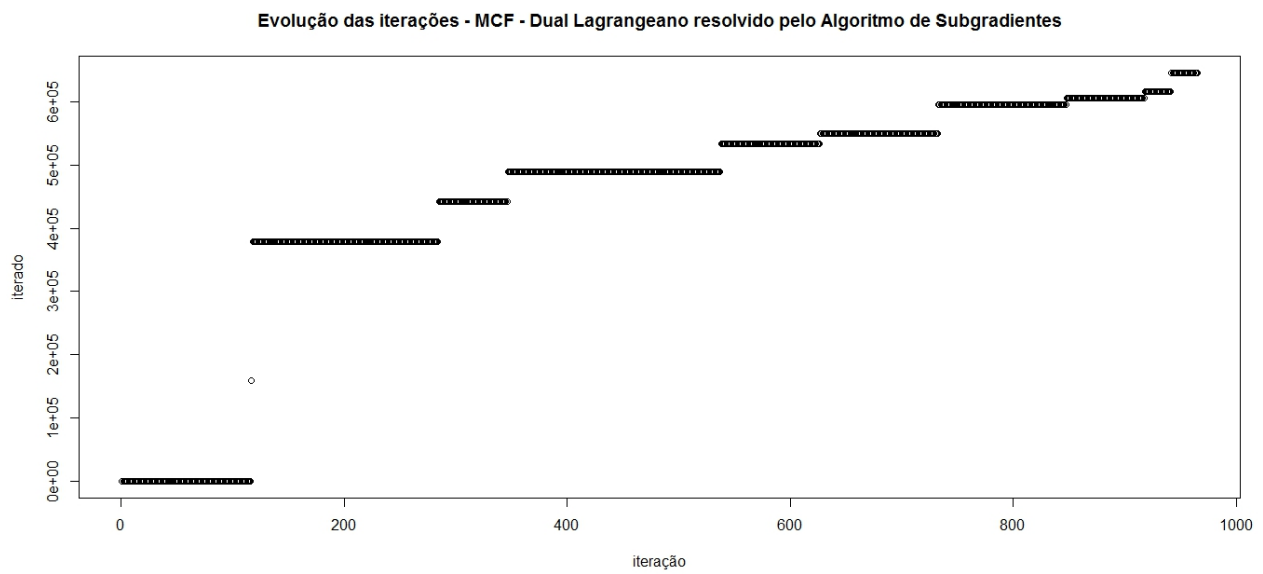


Figura C.6: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados18-0 - Modelo MCF1S1MP.

# Apêndice D

## Gráficos - instâncias dados22 – 0

Nas Figuras D.1 a D.6 apresentamos os gráficos de evolução dos iterados da instância dados22–0 considerando cada um dos métodos objeto deste estudo no tempo fixado em 1 hora.

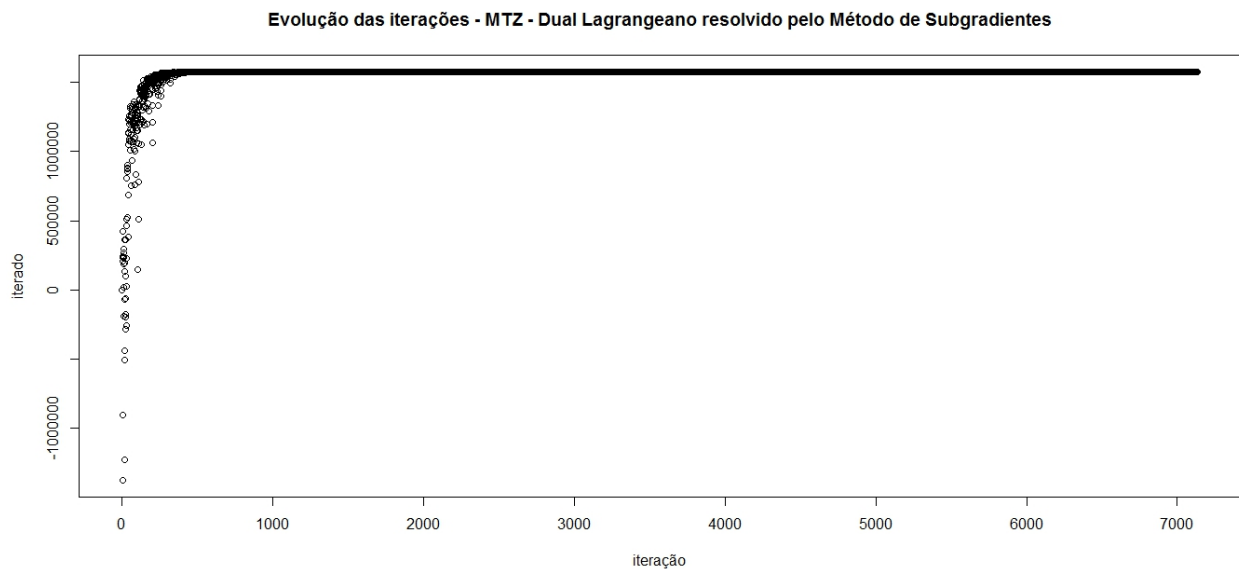


Figura D.1: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MTZ1S1MP.

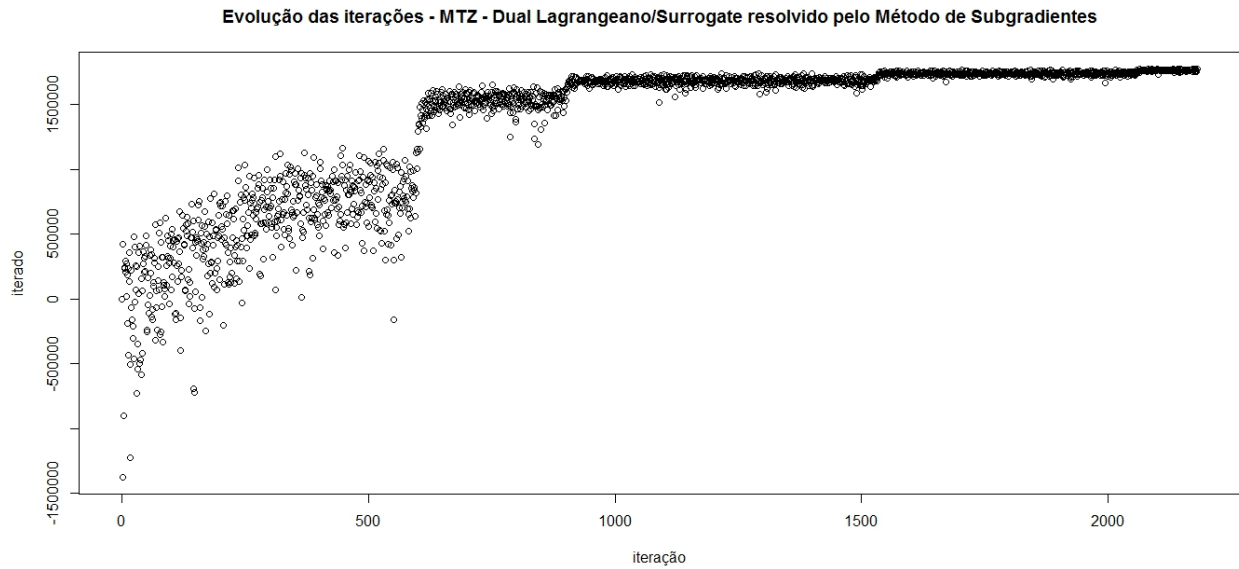


Figura D.2: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MTZ1S1MP.

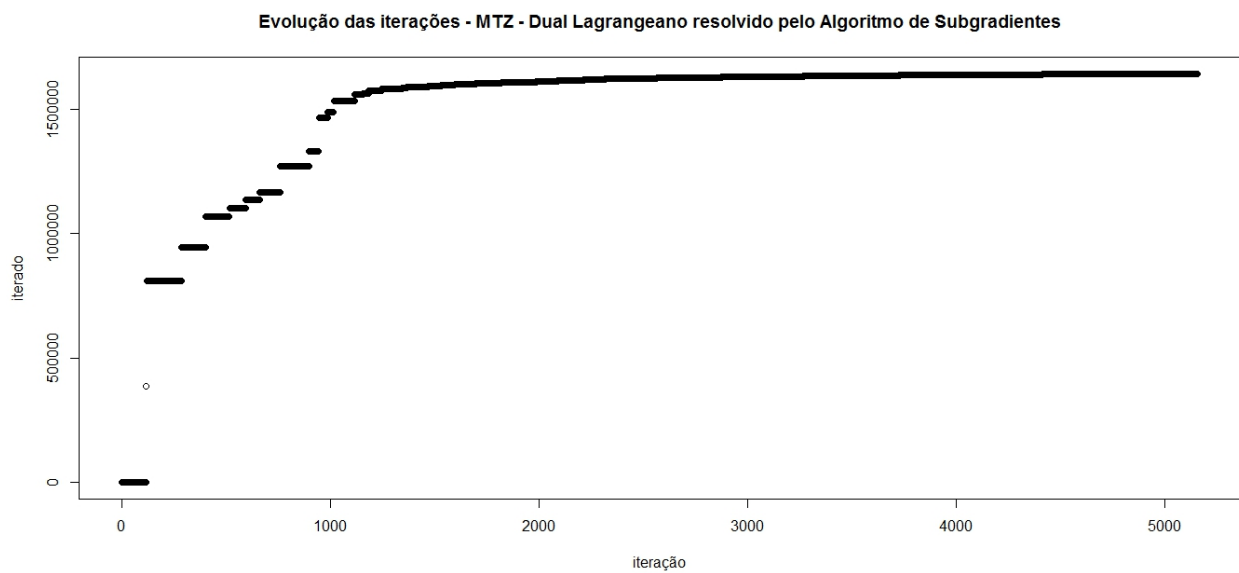


Figura D.3: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados22-0 - Modelo MTZ1S1MP.

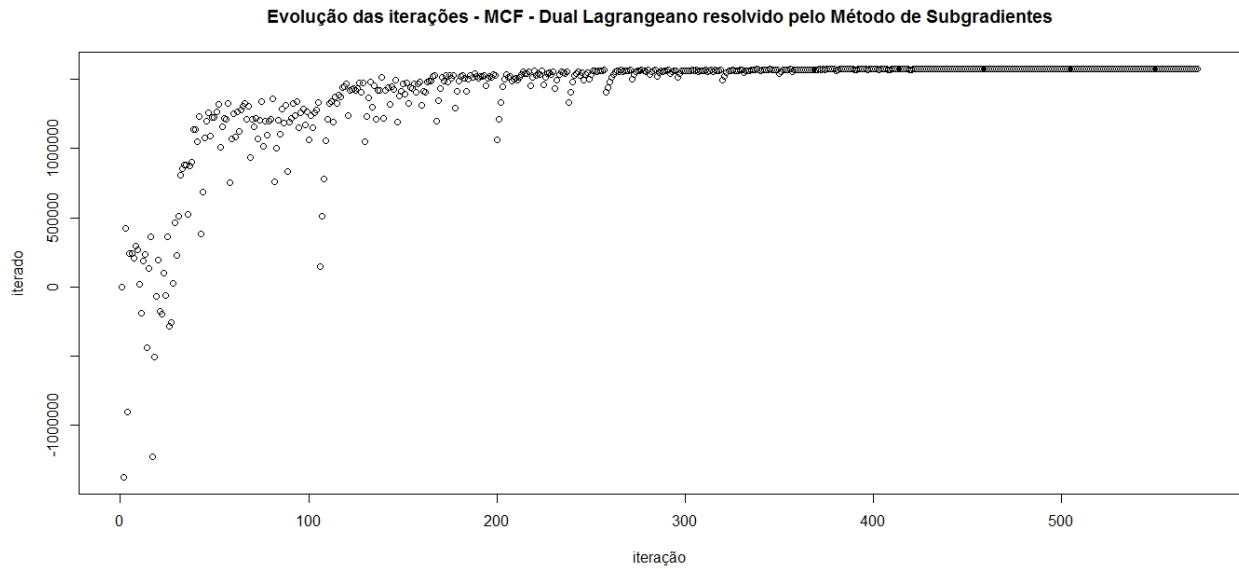


Figura D.4: Evolução dos iterados do dual Lagrangeano resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MCF1S1MP.

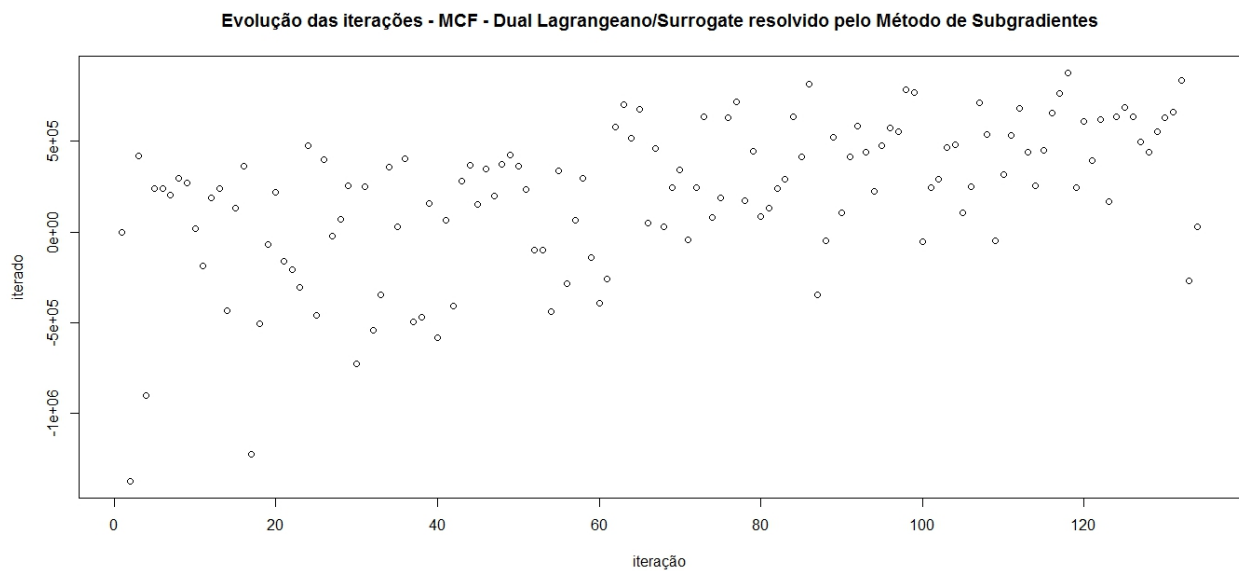


Figura D.5: Evolução dos iterados do dual Lagrangeano/Surrogate resolvido pelo Método de Subgradientes - Dados22-0 - Modelo MCF1S1MP.

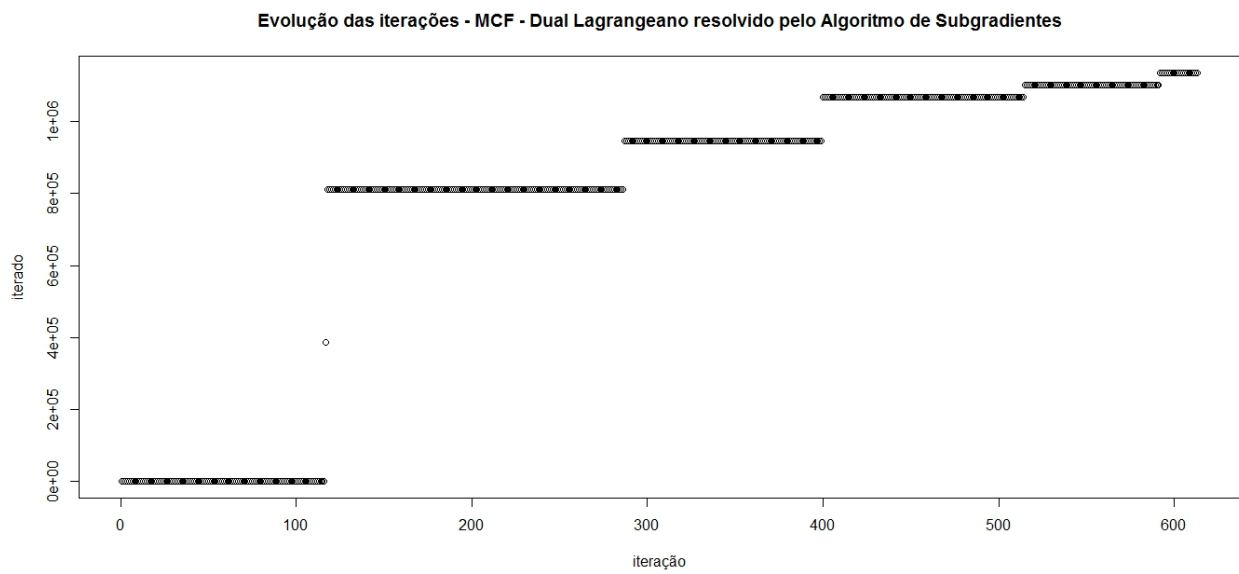


Figura D.6: Evolução dos iterados do dual Lagrangeano resolvido pelo Algoritmo de Volume - Dados22-0 - Modelo MCF1S1MP.

Autorizo a reprodução xerográfica para fins de pesquisa.

São José do Rio Preto, 13 de Agosto de 2015.

---

Tiago Tiburcio da Silva