



O PROBLEMA DE RECOBRIMENTO DE SÓLIDOS POR ESFERAS DE DIFERENTES RAIOS

Renan Vicente Pinto

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: Nelson Maculan Filho

Rio de Janeiro
Março de 2015

O PROBLEMA DE RECOBRIMENTO DE SÓLIDOS POR ESFERAS DE
DIFERENTES RAIOS

Renan Vicente Pinto

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE)
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Nelson Maculan Filho, D.Sc.

Prof. Adilson Elias Xavier, D.Sc.

Prof. Marcia Helena Costa Fampa, D. Sc.

Prof. Rosiane de Freitas Rodrigues, D.Sc.

Prof. Luiz Satoru Ochi, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2015

Vicente Pinto, Renan

O Problema de Recobrimento de Sólidos por Esferas de Diferentes Raios/Renan Vicente Pinto. – Rio de Janeiro: UFRJ/COPPE, 2015.

XV, 75 p.: il.; 29, 7cm.

Orientador: Nelson Maculan Filho

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2015.

Referências Bibliográficas: p. 73 – 75.

1. Problema de Recobrimento por Esferas. 2.
 - Problema de Empacotamento de Esferas. 3.
 - Radiocirurgia Gamma Knife. 4. Clique de Peso Máximo.
- I. Maculan Filho, Nelson. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

À minha família.

Agradecimentos

Gostaria de agradecer a todos que contribuíram, de alguma forma, com esta obra. Não estarão aqui todos os nomes. Peço desculpas aos nomes ausentes.

Sou grato aos meus amigos de laboratório por toda a ajuda e paciência ao longo dos anos que estudamos juntos. Dentre eles, um agradecimento especial a Jesus Ossian e Daniela Lubke. Agradeço também aos mais recentes, Renan Spencer e Hugo Barbalho.

Quanto aos professores, agradeço de coração à ajuda e companheirismo do Professor Nelson Maculan e pela vontade de me orientar na minha formação a doutor. Agradeço ainda aos professores Philippe Michelin, Luidi Simonetti, Adilson Xavier, Sérgio Villas Boas, Abílio Lucena e Márcia Fampa por toda a ajuda.

Tenho a agradecer, ainda, aos amigos que hojem já não fazem mais parte da universidade, mas que também contribuíram para a finalização deste trabalho. Obrigado a Ana Flávia Uzeda, Virgínia Costa e Viviane Kohler.

Obrigado, ainda, pai, mãe, irmã e afilhados e todos da família que me mantiveram focados nos estudos e me apoiaram durante essa fase. Obrigado, Maria de Fátima Marques, pelo carinho dentro da universidade e ajuda com documentações. Obrigado, também, Josefina Solange e Sônia Regina.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

O PROBLEMA DE RECOBRIMENTO DE SÓLIDOS POR ESFERAS DE DIFERENTES RAIOS

Renan Vicente Pinto

Março/2015

Orientador: Nelson Maculan Filho

Programa: Engenharia de Sistemas e Computação

Neste trabalho, apresentamos um modelo de programação matemática para o problema de recobrimento de sólidos por esferas de diferentes raios. Dado um conjunto de esferas, possivelmente de diferentes diâmetros, e um sólido, deseja-se posicionar essas esferas de tal modo que a união delas forme uma cobertura para esse sólido, utilizando a menor quantidade possível de esferas desse conjunto. Esse problema tem aplicação no planejamento do tratamento por radiocirurgia conhecida como *Gamma Knife* e pode ser formulado como um problema de otimização não-convexa, com restrições quadráticas e função objetivo linear. Utilizamos técnicas de discretização a fim de trabalhar com um modelo linear. Propomos, ainda, uma heurística e uma reformulação baseada em uma estrutura de grafos, onde a clique de peso máximo é a solução ótima do modelo original, com a finalidade de encontrar boas soluções em tempos razoáveis.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

THE PROBLEM OF COVERING SOLIDS BY SPHERES OF DIFFERENT RADII

Renan Vicente Pinto

March/2015

Advisor: Nelson Maculan Filho

Department: Systems Engineering and Computer Science

In this work, we present a mathematical programming model for the problem of covering solids by spheres of different radii. Given a set of spheres, possibly with different diameters, and a solid, the goal is to locate the spheres in such a way their union forms a coverage for this solid, using the smallest possible number of spheres of this set. This problem has an application in the radiosurgical treatment planning known as Gamma Knife and can be formulated as a nonconvex optimization problem with quadratic constraints and a linear objective function. Linearizations techniques are also proposed to obtain a linear model. We also present an heuristic and reformulation based on a graph structure, where the maximum weight clique is the optimal solution the original model, in order to find good solutions in reasonable times.

Sumário

Lista de Figuras	x
Lista de Tabelas	xiii
1 Introdução	1
2 O Problema de Recobrimento	4
2.1 Introdução	4
2.2 Um Modelo para o Problema de Recobrimento	5
2.3 Um Problema Relacionado Bidimensional	7
3 O Problema de Empacotamento	11
3.1 Introdução	11
3.2 Um Modelo Para o Problema de Empacotamento	14
4 Uma Aplicação: o Gamma Knife	17
4.1 Introdução	17
4.2 Uma aplicação: o <i>Gamma Knife</i>	18
4.3 Ajustando o problema à aplicação	20
5 Modelo Proposto	22
5.1 O Modelo	22
5.2 A Região de Segurança	26
5.3 Melhorias no modelo	28
6 Alguns Resultados	30
6.1 COUENNE	33
6.2 BONMIN	35
6.3 Xpress-SLP	37
6.4 Análise dos resultados	41
7 Heurística Proposta	43
7.1 A Heurística	43

7.2	Resultados no IPOPT	44
8	Discretização	47
8.1	Primeira abordagem	47
8.2	Segunda abordagem	49
8.3	Incorporando ao modelo	50
8.4	Resultados	51
9	Abordagem por grafo	54
9.1	Introdução	54
9.2	Grafos e Cliques	54
9.3	Nova abordagem	56
9.4	Algoritmo de <i>Branch and Cut</i>	57
9.5	Resultados	59
	9.5.1 Comparação com resultados anteriores	66
	9.5.2 Caso Real	68
10	Conclusão	71
	Referências Bibliográficas	73

Lista de Figuras

2.1	Exemplo de recobrimento para T sendo um paralelepípedo. Como T está totalmente recoberto, não é visível em (a). Em (b), é utilizado o efeito de transparência para melhor visualização.	5
2.2	Exemplo onde não há recobrimento total.	5
2.3	Recobrimento dos mapas do (a) Brasil, com 5 círculos; e (b) do estado do Rio de Janeiro, com 9 círculos.	9
3.1	(a) Arranjo de círculos em formato quadricular e (b) arranjo de círculos em formato hexagonal.	12
3.2	As configurações <i>hexagonal close packing</i> , à esquerda, e a <i>face-centered cubic close packing</i> , à direita. As letras indicam quais camadas são as mesmas. Há duas camadas ‘A’ na configuração hexagonal, onde todas as esferas estão na mesma posição. Todas as camadas na configuração cúbica são diferentes. Esta última pode ser convertida na primeira se transladarmos a camada ‘C’ de forma que a esfera desenhada ocupe a região pontilhada.	12
3.3	Empacotamento de círculos em um quadrado.	13
3.4	Empacotamento de círculos em um círculo.	13
3.5	Empacotamento de círculos em um triângulo.	13
3.6	Empacotamento de círculos em um quadrante circular.	13
3.7	Empacotamento de círculos de raios diferentes em um quadrado.	13
3.8	Empacotamento de círculos de raios diferentes em um círculo.	13
3.9	Variável binária transformada em contínua.	15
4.1	(a) Duas esferas de raio r recobrimo T , em cinza. Em (b), alterou-se o raio de uma das esfera para R_{max} , mantendo tanto a quantidade de esferas quanto o recobrimento total.	17
4.2	(a) Paciente na unidade de tratamento. (b) Diferentes tamanhos de elmos, resultando em 4 diferentes tamanhos de <i>shots</i>	18
4.3	Um <i>shot</i> de radiação e a aproximação esférica da região afetada.	19

5.1	Representação dos parâmetros α_{ij} , que correspondem à máxima interpenetração das esferas.	23
5.2	Representação da região de segurança. A região em cinza representa o volume a ser recoberto.	27
5.3	A região em cinza representa o volume a ser recoberto. Como as esferas devem estar contidas na região de segurança, o domínio da variável que representa o seu centro é restrito à região em laranja. . .	27
5.4	Soluções distintas que apresentam mesmos raios e centros.	28
5.5	Soluções distintas que apresentam mesmos raios e centros.	29
6.1	Exemplo bidimensional mostrando as regiões que correspondem às grandezas utilizadas na descrição de soluções. Em (a), a região a ser recoberta e as bolas presentes no recobrimento. As regiões hachuradas representam os parâmetros (b) cov, (c) miscov e (d) overlap.	31
6.2	Paralelepípedo de dimensões 14mm, 12mm e 10mm, usado como região a ser recoberta.	32
6.3	Imagens das soluções encontradas pelo COUENNE.	34
6.4	Imagens da solução encontrada pelo BONMIN.	36
6.5	Imagens das soluções encontradas pelo XPRESS-SLP.	40
7.1	Imagens da solução da heurística usando IPOPT.	45
8.1	Solução encontrada pelo XPRESS para $\Delta = 3$	52
8.2	Solução encontrada pelo XPRESS para $\Delta = 1$	52
9.1	Exemplo de grafo, onde $V = \{a,b,c,d,e,f\}$ e $A = \{(a,b), (b,c), (b,e), (b,f), (c,d), (c,f), (e,f)\}$	54
9.2	Exemplo de grafos complementares.	55
9.3	Exemplo de construção de vértices.	56
9.4	Grafo à esquerda e subgrafo à direita.	58
9.5	Imagens da solução para $\Delta = 3$	59
9.6	Imagens da solução encontrada pelo XPRESS para $\Delta = 1$	61
9.7	Imagens da solução encontrada pelo <i>Branch and Cut</i> para $\Delta = 1$. . .	61
9.8	Imagens da solução encontrada pelo <i>Branch and Cut</i> para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$	64
9.9	Imagens da solução encontrada pelo <i>Branch and Cut</i> para a instância bt0: (a) o elipsóide a ser recoberto; (b) as esferas no recobrimento; (c) efeito de transparência para melhor visualização; e (d) visualizando a parte não recoberta.	67
9.10	Dados reais cedidos pela empresa Elekta.	68

9.11	Imagens da solução encontrada pelo <i>Branch and Cut</i> para $\varepsilon = 1$, $c_r = r^3$	69
9.12	Imagens da solução encontrada pelo <i>Branch and Cut</i> para $\varepsilon = 1$, $c_4 = 20$ e $c_2 = 1$	70
9.13	Imagens da solução encontrada pelo <i>Branch and Cut</i> sem o uso da região de segurança, com efeito de transparência em (b).	70

Lista de Tabelas

6.1	Diferentes conjuntos de esferas disponibilizadas.	32
6.2	Soluções encontradas pelo COUENNE.	34
6.3	Resultado do COUENNE para o conjunto Dados1.	34
6.4	Resultado do COUENNE para o conjunto Dados2.	35
6.5	Soluções encontradas pelo BONMIN.	36
6.6	Resultado do BONMIN para o conjunto Dados1.	37
6.7	Soluções encontradas pelo XPRESS-SLP.	38
6.8	Resultado do XPRESS-SLP para o conjunto Dados1.	38
6.9	Resultado do XPRESS-SLP para o conjunto Dados2.	39
6.10	Resultado do XPRESS-SLP para o conjunto Dados3.	39
6.11	Comparação entre soluções.	41
7.1	Solução da heurística usando IPOPT.	45
7.2	Resultado da heurística usando IPOPT.	46
8.1	Resultados do XPRESS para os modelos discretizados.	51
8.2	Resultado do XPRESS para $\Delta = 3$	53
8.3	Resultado do XPRESS para $\Delta = 1$	53
9.1	Soluções para $\Delta = 3$	59
9.2	Resultado para $\Delta = 3$	60
9.3	Soluções para $\Delta = 1$	61
9.4	Resultado do XPRESS para $\Delta = 1$	62
9.5	Resultado do <i>Branch and Cut</i> para $\Delta = 1$	63
9.6	Solução do <i>B&C</i> para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$, em comparação com IPOPT.	64
9.7	Resultado do <i>Branch and Cut</i> para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$	65
9.8	Instâncias utilizadas em LIBERTI <i>et al.</i> [1].	66
9.9	Comparação entre os resultados de [1] e o algoritmo de <i>Branch and Cut</i> proposto.	66
9.10	Instância como em bt1, bt2 e bt3, com $\Delta = 1$	67
9.11	Resultado para a instância com $\Delta = 1$	67

9.12 Soluções do <i>B&C</i> para o teste com dados reais.	69
10.1 Comparação das soluções obtidas para uma mesma instância e diferentes métodos de resolução.	72

Capítulo 1

Introdução

Problemas de recobrimento aparecerem no cotidiano sob uma variedade de formas. Comumente, são problemas que envolvem determinar o posicionamento de certos objetos, visando recobrir uma área ou um outro objeto de maior dimensão.

Um exemplo prático, de fácil entendimento, surge na área de telecomunicações, quando há a necessidade de instalação de equipamentos. Deseja-se atender ou recobrir uma ou mais regiões predeterminadas, minimizando os custos. Supondo que cada equipamento instalado consiga atender a uma região circular centrada no mesmo, teríamos um problema de recobrimento de uma região bidimensional por círculos. Os custos podem diferir por diversas razões, como, por exemplo, de acordo com locais de instalação, em razão do tipo de terreno no local.

Matematicamente, esses problemas são formulados como modelos de programação matemática, que consistem em encontrar valores para um conjunto de variáveis, de forma a maximizar ou minimizar uma dada equação, chamada de função objetivo, satisfazendo um conjunto de restrições. Quando a função objetivo é linear e as restrições do problema formam um sistema de equações ou inequações lineares, dizemos que o modelo é linear (LP). Caso contrário, temos um modelo não-linear (NLP). Diremos, neste trabalho, que o problema é convexo quando o conjunto de restrições do modelo formar uma região convexa e a função a otimizar é também convexa. Quando é necessário que algumas variáveis do modelo, mas não todas, assumam valores no conjunto dos inteiros, temos um modelo de programação inteira mista (MILP ou MINLP). Se todas as variáveis devem ser um número inteiro, então temos um modelo de programação inteira (IP).

Variáveis inteiras que só podem assumir valor 0 ou 1 são chamadas de variáveis binárias. Com relação ao modelo, são também chamadas de variáveis de decisão, pois, normalmente, representam alguma escolha entre duas opções. Se todas as variáveis inteiras possuem um limite superior, então podemos transformar o problema inteiro em um problema contendo apenas variáveis binárias [2].

Um exemplo de modelo matemático para o problema geral de recobrimento, for-

mulado como um problema de programação linear inteira, na forma como é bastante encontrado na literatura, pode ser escrito como segue:

$$\min \quad \sum_{i=1}^n c_i x_i \quad (1.1)$$

s.a.

$$\sum_{i=1}^n a_{ij} x_i \geq b_j, \quad \forall j \in \{1, \dots, m\} \quad (1.2)$$

$$x_i \in \mathbb{N}, \quad \forall i \in \{1, \dots, n\} \quad (1.3)$$

Em (1.1)-(1.3), temos n tipos diferentes de objetos a serem usados no recobrimento. Cada um desses objetos de tipo i , com $i \in \{1, \dots, n\}$, tem um custo associado, c_i . As variáveis x_i representam quantos objetos do tipo i estão sendo usados. Se todas as restrições (1.2) estão sendo satisfeitas, é dito que $x = (x_1, \dots, x_n)$ é um recobrimento. Os parâmetros $\{a_{ij}\}_{i,j}$ armazenam informações sobre a estrutura a ser recoberta. Finalmente, o recobrimento ótimo é aquele que apresenta o menor custo total, ou o menor valor da função objetivo (1.1). Em muitos casos, as variáveis x_i são binárias, podendo somente assumir valor 0 ou 1. Um caso particular, quando todos os custos c_i são iguais, é o recobrimento que utiliza a menor quantidade possível de objetos.

No exemplo mencionado anteriormente, da área de telecomunicações, as variáveis de decisão $x_i \in \{0, 1\}$ representariam a instalação ou não do equipamento em um determinado local i . Para o uso desse modelo, deve-se previamente determinar um conjunto finito de possíveis locais de instalação. Os parâmetros a_{ij} assumiriam, para cada ponto j da região a ser recoberta, valor 1 caso um equipamento instalado no local i consiga atender a esse ponto j . Se for suficiente que cada ponto j da região seja atendido por apenas um equipamento, teremos os parâmetros b_j iguais a 1.

Os problemas de recobrimento são, naturalmente, problemas de minimização. Uma outra classe de problemas, bastante relacionada, é conhecida como problemas de empacotamento. Esses são formulados como problemas de maximização e são problemas duais dos problemas de recobrimento. Consistem em posicionar o máximo possível de objetos dentro de um objeto maior, normalmente chamado de contêiner, sem que haja sobreposição dos mesmos. Como exemplo simples, podemos citar o abastecimento de laranjas dentro de uma caixa.

Existem aplicações onde se pode utilizar tanto modelos de recobrimento quanto modelos de empacotamento como estratégias para encontrar soluções do problema associado. Podemos modelar problemas de empacotamento como segue:

$$\max \quad \sum_{i=1}^n c_i x_i \quad (1.4)$$

s.a.

$$\sum_{i=1}^n a_{ij} x_i \leq b_j, \quad \forall j \in \{1, \dots, m\} \quad (1.5)$$

$$x_i \in \mathbb{N}, \quad \forall i \in \{1, \dots, n\} \quad (1.6)$$

As diferenças entre os dois modelos são a direção de otimização e o sinal nas inequações.

Há bastantes estudos realizados sobre problemas de empacotamento de círculos. Em [3], há um resumo de melhores resultados para empacotamentos de círculos em diversos tipos de figuras geométricas, como retângulos, partes de círculos e triângulos, tendo esses círculos diâmetros iguais ou tamanhos diferentes. São apresentados, ainda, alguns resultados sobre empacotamentos de esferas de raios iguais em cubos.

Enquanto problemas de recobrimento e de empacotamento envolvendo objetos bidimensionais são, em geral, bem estudados e solucionados atualmente, os correspondentes desses problemas no \mathbb{R}^3 ainda apresentam grandes dificuldades na busca de soluções ótimas globais.

A seguir, apresentaremos, no Capítulo 2, o problema de recobrimento por esferas, tanto de figuras planas quanto de sólidos. Em seguida, no Capítulo 3, apresentaremos o problema de empacotamento de esferas. Uma aplicação dessa família de problemas será apresentada no Capítulo 4, referente ao planejamento do tratamento pela radiocirurgia conhecida como *Gamma Knife*. No Capítulo 5, proporemos um modelo de programação matemática para o problema em estudo. Resultados para esse modelo aplicado a alguns resolvedores comerciais serão apresentados no Capítulo 6. Nos Capítulos 7 e 8, apresentaremos uma heurística para tentar encontrar soluções de melhor qualidade e discutiremos técnicas de linearização para serem aplicadas ao modelo proposto, respectivamente. Finalmente, no Capítulo 9, apresentaremos uma nova abordagem para o problema, consistindo em utilizar informações do modelo proposto na geração de um grafo onde a clique de peso máximo é a solução ótima do modelo original.

Capítulo 2

O Problema de Recobrimento

2.1 Introdução

Neste capítulo, apresentaremos, com detalhes, o problema de recobrimento por esferas de diferentes raios. Apresentaremos, ainda, alguns modelos já presentes na literatura para esse tipo de problema e para problemas relacionados.

Podemos descrever matematicamente o problema da seguinte forma:

(P) Dados um conjunto compacto $T \subset \mathbb{R}^3$, um conjunto finito de raios $R \subset \mathbb{R}_+$, um conjunto N indexando as esferas e uma função $\rho : N \rightarrow R$, encontrar um conjunto de esferas $\{B(x(i), \rho(i)) \mid i \in N\}$ de cardinalidade mínima recobrindo cada ponto de T .

Em outras palavras, dado um volume T a ser recoberto, devemos posicionar a menor quantidade possível de esferas de forma a realizar o recobrimento de T . Por recobrimento, entende-se que cada ponto de T deve pertencer, também, a alguma esfera. O problema (P), como definido acima, pode ser formulado como um problema de programação matemática, com o objetivo de encontrar os centros de cada esfera e associar a cada uma delas um respectivo raio, através da função ρ , de forma que a união dessas esferas forme uma cobertura do conjunto T .

Nas Figuras 2.1 e 2.2, são exemplificados o caso em que as esferas recobrem totalmente o volume T e o caso em que não há recobrimento total, respectivamente.

Na Seção 2.2, apresentaremos um modelo para o problema geral de recobrimento de uma região tridimensional por esferas. Na Seção 2.3, mostraremos um problema relacionado, bidimensional, onde deseja-se recobrir regiões geográficas com círculos. Esse problema é aplicado, por exemplo, na área de telecomunicações.

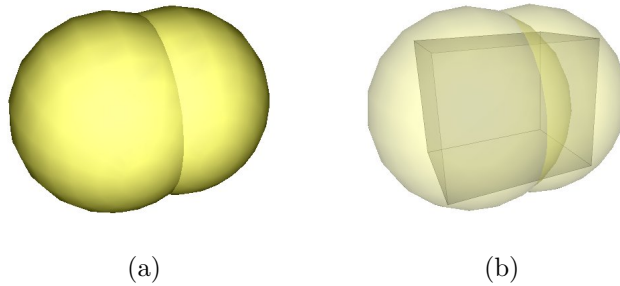


Figura 2.1: Exemplo de recobrimento para T sendo um paralelepípedo. Como T está totalmente recoberto, não é visível em (a). Em (b), é utilizado o efeito de transparência para melhor visualização.

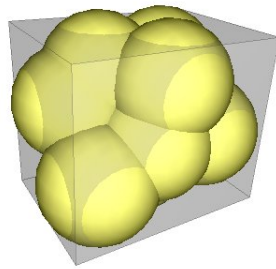


Figura 2.2: Exemplo onde não há recobrimento total.

2.2 Um Modelo para o Problema de Recobrimento

Apresentaremos, nesta seção, uma formulação matemática para o problema de recobrimento de uma região compacta do \mathbb{R}^3 utilizando esferas de diferentes raios, como definido em (P). Um modelo de programação infinita não-linear inteira mista foi desenvolvido por LIBERTI *et al.* [1], com um número infinito não-enumerável de variáveis e restrições, como segue.

Sejam N , R e T como descritos anteriormente em (P). Sejam, ainda, U o conjunto de índices para os elementos de R , uma constante $M \geq \text{diam}(T)$, suficientemente grande, e uma constante $\varepsilon > 0$, com $|\varepsilon| \ll 1$. Considere as seguintes variáveis:

- $x^i : N \rightarrow \mathbb{R}^3$, que representa o centro da esfera i ;
- $y_i : N \rightarrow \{0, 1\}$, que assume valor $\begin{cases} 1, & \text{se a esfera } i \text{ é usada na solução,} \\ 0, & \text{caso contrário;} \end{cases}$
- $u_i(p) : N \times T \rightarrow \{0, 1\}$, que assume valor $\begin{cases} 1, & \text{caso a esfera } i \text{ cubra o ponto } p \in T, \\ 0, & \text{caso contrário;} \end{cases}$
- $w_{ij} : N \times U \rightarrow \{0, 1\}$, que assume valor $\begin{cases} 1, & \text{caso a esfera } i \text{ tenha o raio } r_j \in R, \\ 0, & \text{caso contrário;} \end{cases}$

Com essas variáveis, os autores desenvolveram um modelo que não possui função objetivo. Assim, qualquer solução é um recobrimento para T , sendo, nesse momento, irrelevante a quantidade de esferas na solução. O problema é corretamente descrito pelas inequações (2.1)-(2.5):

$$\|x^i - p\|^2 \leq u_i(p) \sum_{j \in U} w_{ij} r_j^2 + (1 - u_i(p)) M^2, \quad \forall i \in N, \forall p \in T \quad (2.1)$$

$$\sum_{j \in U} w_{ij} = 1, \quad \forall i \in N \quad (2.2)$$

$$\sum_{i \in N} u_i(p) \geq 1, \quad \forall p \in T \quad (2.3)$$

$$\int_{p \in T} u_i(p) dp \geq \varepsilon y_i, \quad \forall i \in N \quad (2.4)$$

$$\int_{p \in T} u_i(p) dp \leq \text{Vol}(T) y_i, \quad \forall i \in N \quad (2.5)$$

De acordo com as restrições (2.1), se a esfera i cobre o ponto $p \in T$, situação representada por $u_i(p) = 1$, então a distância euclideana entre p e o centro x^i dessa esfera deve ser, no máximo, o raio dessa esfera. No caso contrário, utiliza-se uma constante M suficientemente grande, como o diâmetro do objeto a ser recoberto, por exemplo, para assegurar a validade da inequação. O fato de a cada esfera poder ser atribuído apenas um raio é representado pelas restrições (2.2). As restrições (2.3) asseguram que cada ponto $p \in T$ será recoberto por, pelo menos, uma esfera. Se a esfera i foi selecionada para a solução, onde teremos $y_i = 1$, as restrições (2.4) asseguram que ela recobre um volume não nulo de T . Caso contrário, as restrições (2.5) forçam as funções u_i a serem funções nulas.

Em (2.1) e em (2.3), temos uma restrição para cada ponto $p \in T$, resultando em um número infinito, não-enumerável, de restrições. Para contornar essa dificuldade, os autores aproximaram a região T por um conjunto finito de pontos, técnica conhecida como discretização, obtendo um modelo de programação não-linear, misto e não-convexo. Em seguida, a fim de eliminar a não-convexidade do modelo, causada pelos produtos entre variáveis binárias $u_i(p) w_{ij}$, foram introduzidos parâmetros $\rho : N \rightarrow \mathbb{R}_+$ que designam, previamente, a cada esfera, um raio pertencente a R . Tais parâmetros são incorporados ao problema em substituição às variáveis w_{ij} . O modelo resultante é, então, convexo, não-linear e misto:

$$\|x^i - p_v\|^2 \leq u_{iv}\rho_i^2 + (1 - u_{iv})M^2, \quad \forall i \in N, \forall v \in V \quad (2.6)$$

$$\sum_{i \in N} u_{iv} \geq 1, \quad \forall v \in V \quad (2.7)$$

$$\sum_{v \in V} u_{iv} \geq y_i, \quad \forall i \in N \quad (2.8)$$

$$\sum_{v \in V} u_{iv} \leq |V|y_i, \quad \forall i \in N \quad (2.9)$$

Com a nova notação, $\{p_v \mid v \in V\}$ é o conjunto dos pontos que representam a discretização de T , sendo V o conjunto de índices para esses pontos. Com a eliminação das variáveis binárias de atribuição de raio w_{ij} , houve um aumento do valor de N ou da quantidade de esferas, mas resultou em uma melhoria computacional.

Para encontrar a solução ótima do problema de recobrimento em estudo, definida como a que apresenta a menor quantidade de esferas, basta adicionar a seguinte função objetivo ao modelo:

$$\min \sum_{i \in N} y_i. \quad (2.10)$$

Por fim, os autores ainda utilizam técnicas de aproximação interna da função norma, a fim de transformar o modelo em um problema de programação linear inteira mista (MILP).

2.3 Um Problema Relacionado Bidimensional

Consideramos, nesta seção, uma versão bidimensional do problema de recobrimento, estudado por XAVIER e DE OLIVEIRA [4]. No problema que apresentaremos, a quantidade de círculos na solução é prefixada, ou seja, é um parâmetro prévio.

Busca-se encontrar, então, além do posicionamento de cada círculo, o menor raio possível, o mesmo para todos os círculos, de forma a se obter um recobrimento total.

Esse problema surge em um grande número de aplicações práticas, como, por exemplo, em localização e dimensionamento de centros de telecomunicação. O problema foi formulado pelos autores como um problema min-max-min, como explicado a seguir.

Seja n a quantidade de círculos com a qual deseja-se recobrir uma determinada região compacta $T \subset \mathbb{R}^2$. Seja $X = \{x^i \mid i \in N\}$ o conjunto dos centros desses círculos, com $N = \{1, \dots, n\}$. Dado um ponto $p \in T$, define-se a distância de p ao centro do círculo mais próximo por

$$d(p, X) = \min_{x^i \in X} \|p - x^i\|_2 . \quad (2.11)$$

A distância $d(p, X)$ é uma medida de recobrimento para um ponto p específico. De forma a qualificar o recobrimento de toda a região T pelos n círculos, os autores tomam o maior valor dentre as distâncias $d(p, X)$, o que corresponde ao recobrimento mais crítico de algum ponto $p \in T$. Esse valor é denotado como $D(X)$, e é escrito como

$$D(X) = \max_{p \in T} d(p, X) . \quad (2.12)$$

Essas distâncias são calculadas a partir de uma configuração fixa dos círculos. Ou seja, para o cálculo das distâncias, os círculos já devem ter as posições dos seus centros definidas. O posicionamento ótimo dos centros deve acarretar em um recobrimento de T de melhor qualidade. Para esse feito, minimiza-se o recobrimento mais crítico dos pontos. Denotando por X^* o posicionamento ótimo, temos:

$$X^* = \arg \min_{X \in \mathbb{R}^{2n}} D(X) . \quad (2.13)$$

Escrevendo de outra forma, temos:

$$X^* = \arg \min_{X \in \mathbb{R}^{2n}} \max_{p \in T} \min_{x^i \in X} \|p - x^i\|_2 . \quad (2.14)$$

Novamente, para resolver o problema numericamente, faz-se necessário discretizar o volume T . Após algumas reformulações, com prova de que a formulação final possui mesmo valor ótimo que 2.14, foi aplicada a técnica de suavização hiperbólica [5], para lidar com a não-diferenciabilidade da função empregada. Esta técnica consiste em solucionar uma sequência de problemas diferenciáveis que gradualmente se aproximam do problema original. Bons resultados numéricos foram obtidos. Alguns exemplos podem ser vistos na Figura 2.3, retiradas de [4].

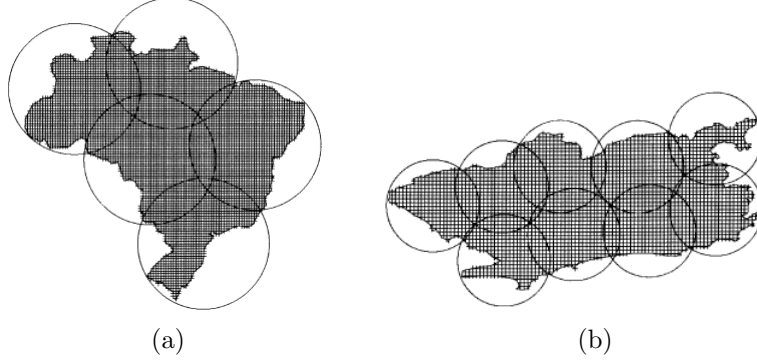


Figura 2.3: Recobrimento dos mapas do (a) Brasil, com 5 círculos; e (b) do estado do Rio de Janeiro, com 9 círculos.

Seja p_v um ponto da aproximação de T , com $v \in V$. Primeiramente, a partir de (2.11), cria-se um novo conjunto de variáveis z_v , com

$$z_v(x) = \min_{x^i \in X} \|p_v - x^i\|_2, \quad \forall v, \quad (2.15)$$

que devem, necessariamente, satisfazer o seguinte conjunto de desigualdades:

$$z_v(x) - \|p_v - x^i\|_2 \leq 0, \quad \forall i, \forall v. \quad (2.16)$$

Similarmente, a partir de (2.12), se criarmos a variável z ,

$$z(x) = \max_{v \in V} z_v(x), \quad (2.17)$$

então z deverá respeitar as restrições

$$z(x) \geq z_v(x), \quad \forall v. \quad (2.18)$$

Considere, então, o seguinte problema:

$$\min \quad z \quad (2.19)$$

$$\text{s.a.} \quad z_v - \|p_v - x^i\|_2 \leq 0, \quad \forall i, \forall v \quad (2.20)$$

$$z \geq z_v, \quad \forall v \quad (2.21)$$

Esse problema não é equivalente a (2.14), visto que as variáveis z_j não são limitadas inferiormente e, por consequência, z também não tem limite inferior. Para obter a equivalência, utilizaram a função $\varphi(y) = \max\{0, y\}$ e as restrições (2.20) para obter que

$$\sum_{i=1}^n \varphi(z_v - \|p_v - x^i\|_2) = 0, \quad \forall v \in V. \quad (2.22)$$

Porém, substituir (2.20) por (2.22) mantém a propriedade de z e z_j não possuírem limite inferior. O limite foi, então, obtido com uma leve perturbação na restrição, como segue:

$$\min \quad z \quad (2.23)$$

$$\text{s.a.} \quad \sum_{i=1}^n \varphi(z_v - \|p_v - x^i\|_2) \geq \varepsilon, \quad \forall v \in V \quad (2.24)$$

$$z \geq z_v, \quad \forall v \in V \quad (2.25)$$

para $\varepsilon > 0$. Como o conjunto viável de (2.19)-(2.21) é o limite do conjunto viável de (2.19)-(2.25), quando ε tende a zero, os autores resolveram um sequência de problemas como (2.19)-(2.25) para valores decrescentes de ε .

Os autores ainda diminuíram consideravelmente a dimensão do problema ao mostrar que sempre existe uma solução onde $z = z_v$, $\forall v \in V$, permitindo, assim, que o problema seja reescrito como

$$\min \quad z \quad (2.26)$$

$$\text{s.a.} \quad \sum_{i=1}^n \varphi(z - \|p_v - x^i\|_2) \geq \varepsilon, \quad \forall v \in V \quad (2.27)$$

Por fim, como a função $\varphi(y)$ é não-diferenciável no ponto $y = 0$, foi utilizada a técnica de suavização hiperbólica, que consiste em substituir a função φ por uma função aproximada $\phi : \mathbb{R} \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$:

$$\phi(y, \tau) = (y + \sqrt{y^2 + \tau^2})/2. \quad (2.28)$$

A função apresenta as seguintes propriedades:

- (a) $\phi(y, \tau) > \varphi(y)$, $\forall \tau > 0$;
- (b) $\lim_{\tau \rightarrow 0} \phi(y, \tau) = \varphi(y)$;
- (c) $\phi(\cdot, \tau)$ é uma função convexa, crescente e de classe C^∞ .

Um trabalho recente, que estende essa abordagem a recobrimentos de corpos tridimensionais por esferas, é a dissertação de mestrado da LUBKE [6]. Podemos citar também um outro trabalho, de GALIYEV [7], onde o recobrimento de regiões planas por um conjunto prefixado de elipses foi objeto de estudo.

Capítulo 3

O Problema de Empacotamento

3.1 Introdução

Os problemas de empacotamento são uma classe de problemas de otimização combinatória que envolvem empacotar certos objetos dentro de algum contêiner, da forma mais densa possível, sem sobreposição dos objetos. Por densidade, entende-se a proporção do volume da região a ser recoberta que está sendo preenchida pelos objetos. Matematicamente, temos:

$$\text{densidade} = \frac{\sum_i \text{volume}(\text{objeto}_i)}{\text{volume}(\text{contêiner})}. \quad (3.1)$$

Estamos interessados no caso em que os objetos são esferas n -dimensionais e almejamos empacotá-las em regiões compactas contidas em \mathbb{R}^n . Em regiões não-limitadas, soluções desses tipos de problemas envolvem posicionar esferas em arranjos reticulados, ou arranjos regulares, onde os centros das esferas formam um padrão simétrico, embora existam outros tipos de arranjos.

Em duas dimensões, para círculos de mesmo raio, dois arranjos que surgem naturalmente são o arranjo em formato quadricular e o arranjo em formato hexagonal, como podem ser vistos na Figura 3.1.

O empacotamento de esferas com o arranjo hexagonal é sabido ser o mais denso no plano, provado pelo matemático László Fejes Tóth [8, 9], em 1940. Essa maneira de dispor as esferas resulta em uma densidade de

$$\frac{1}{6}\pi\sqrt{3} \approx 0.9069. \quad (3.2)$$

Para o caso tridimensional, uma disposição aleatória das esferas apresenta um resultado experimental com densidade de, aproximadamente, 0.65. A melhor densidade que pode ser obtida em três dimensões, segundo a conjectura de Kepler,

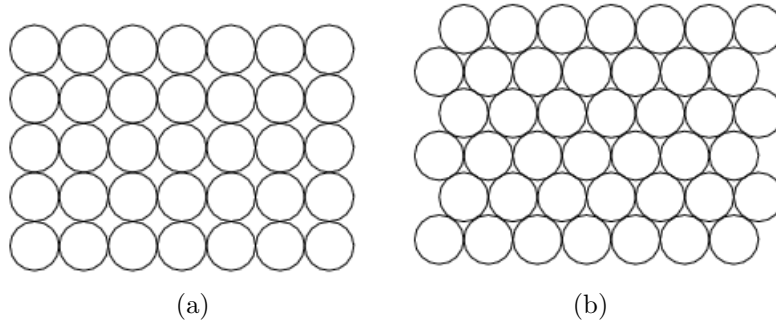


Figura 3.1: (a) Arranjo de círculos em formato quadricular e (b) arranjo de círculos em formato hexagonal.

provada em HALES [10], é a resultante dos arranjos conhecidos como *face-centered cubic close packing* e *hexagonal close packing*. Cada uma delas pode ser vista na Figura 3.2.

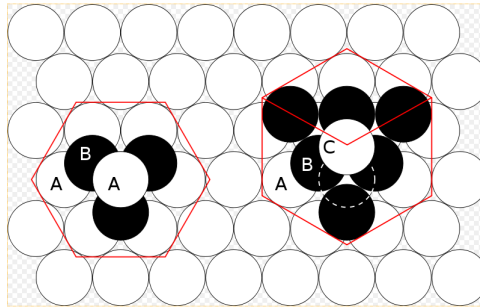


Figura 3.2: As configurações *hexagonal close packing*, à esquerda, e a *face-centered cubic close packing*, à direita. As letras indicam quais camadas são as mesmas. Há duas camadas ‘A’ na configuração hexagonal, onde todas as esferas estão na mesma posição. Todas as camadas na configuração cúbica são diferentes. Esta última pode ser convertida na primeira se transladarmos a camada ‘C’ de forma que a esfera desenhada ocupe a região pontilhada.

Ambas as configurações apresentam a mesma densidade média, com valor

$$\frac{\pi\sqrt{18}}{18} \approx 0.74048. \quad (3.3)$$

Quando há um contêiner, ou seja, quando queremos empacotar os objetos em um subconjunto compacto do espaço, há na literatura resultados bem interessantes. Há estudos bem abrangentes quando o assunto é empacotamento de círculos [3, 11]. Encontra-se, empacotamento desses objetos em círculos maiores, em retângulos, em formatos especiais de triângulos e até em cilindros [12]. Alguns exemplos podem ser vistos nas Figuras 3.3 a 3.8, todas retiradas de [3].

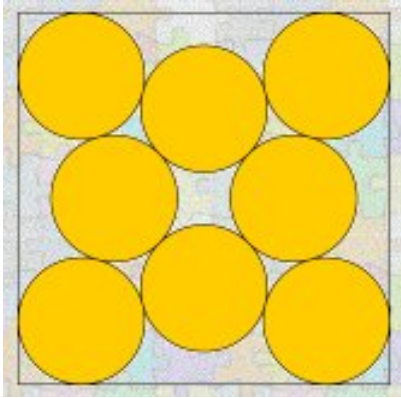


Figura 3.3: Empacotamento de círculos em um quadrado.

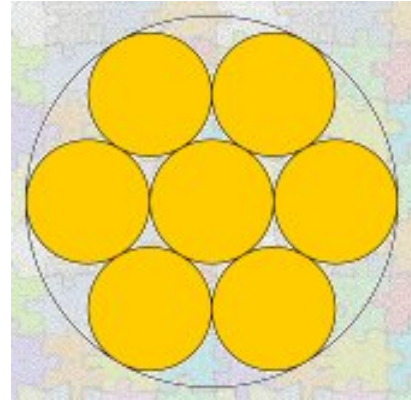


Figura 3.4: Empacotamento de círculos em um círculo.

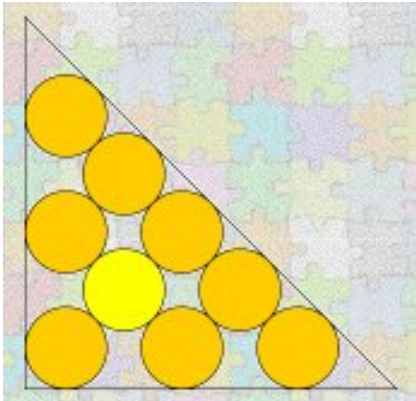


Figura 3.5: Empacotamento de círculos em um triângulo.

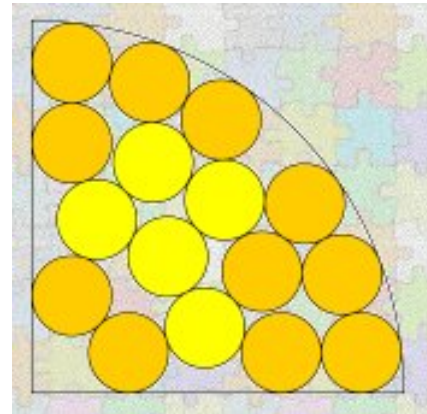


Figura 3.6: Empacotamento de círculos em um quadrante circular.

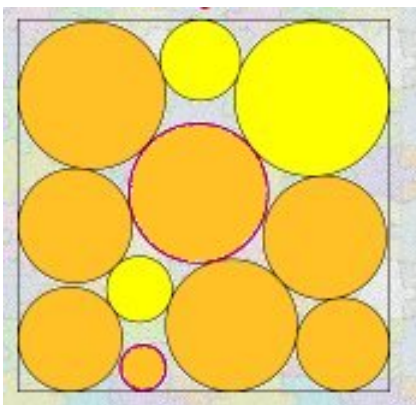


Figura 3.7: Empacotamento de círculos de raios diferentes em um quadrado.

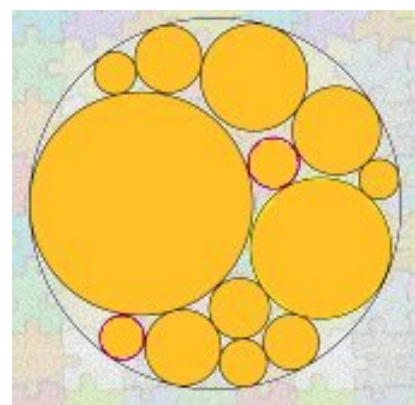


Figura 3.8: Empacotamento de círculos de raios diferentes em um círculo.

3.2 Um Modelo Para o Problema de Empacotamento

Como estamos interessados, neste trabalho, em regiões tridimensionais, apresentaremos, a seguir, um modelo matemático para o problema de empacotamento de esferas em um politopo em \mathbb{R}^3 , estudado por SUTOU e DAI [13]. Um politopo é a interseção de semi-espacos, definidos por hiperplanos. Considere os seguintes parâmetros:

- r_k , com $k = 1, \dots, K$, os K diferentes raios das esferas;
- q_k , a quantidade de esferas disponíveis para cada raio r_k ; e
- o politopo $P = \{(x, y, z) \in \mathbb{R}^3 \mid a_m x + b_m y + c_m z \geq d_m, \quad m = 1, \dots, M\}$, para algum $M > 0$.

Em [13], os autores apresentam um modelo de programação matemática cuja solução é um empacotamento de, no máximo, L esferas, em um politopo P , como definido acima. O problema é formulado como um problema não-convexo, com restrições quadráticas e função objetivo linear. São utilizadas as seguintes variáveis no modelo:

- (x_i, y_i, z_i) , $i = 1, \dots, L$, representando o centro da esfera i ; e
- $t_{ik} \in \{0, 1\}$, que assume valor $\begin{cases} 1, & \text{caso a esfera } i \text{ tenha raio } r_k, \\ 0, & \text{caso contrário;} \end{cases}$

Dessa forma, o problema de otimização foi formulado como em (3.4)-(3.10):

$$\max \quad \frac{4}{3}\pi \sum_{i=1}^L \sum_{k=1}^K r_k^3 t_{ik} \quad (3.4)$$

$$\text{s. a} \quad \|(x_i, y_i, z_i) - (x_j, y_j, z_j)\|^2 \geq \left(\sum_{k=1}^K r_k t_{ik} + \sum_{k=1}^K r_k t_{jk} \right)^2, \quad \forall i \leq j, i \neq j \quad (3.5)$$

$$\frac{|a_m x_i + b_m y_i + c_m z_i - d_m|}{\sqrt{(a_m^2 + b_m^2 + c_m^2)}} \geq \sum_{k=1}^K r_k t_{ik}, \quad \forall i, \forall m \quad (3.6)$$

$$a_m x_i + b_m y_i + c_m z_i - d_m \geq 0, \quad \forall i, \forall m \quad (3.7)$$

$$\sum_{k=1}^K t_{ik} \leq 1, \quad \forall i \quad (3.8)$$

$$\sum_{i=1}^L t_{ik} \leq q_k, \quad \forall k \quad (3.9)$$

$$t_{ik} \in \{0, 1\}, \quad \forall i, \forall k \quad (3.10)$$

As restrições (3.5) asseguram que as esferas não se interceptam e as restrições (3.7) forçam os seus centros a pertencerem ao politopo. As restrições (3.8) e (3.10) garantem que, no máximo, um raio será escolhido para cada esfera. Isto é, se $t_{ik} = 1$, a esfera i , com raio r_k , é empacotada na posição (x_i, y_i, z_i) . As restrições (3.9) e (3.10) limitam em q_k a quantidade de esferas com raio r_k . Junto com as restrições (3.7) e (3.10), as restrições (3.6) exprimem que a distância entre o centro de uma esfera e a fronteira do politopo é de, pelo menos, o raio dessa esfera. Ou seja, a esfera deve estar totalmente contida no politopo.

A equação (3.4) define o objetivo do problema, a maximização da densidade do empacotamento. O custo de cada esfera na função objetivo é o seu volume, de acordo com (3.1). Logo, esferas de diâmetros maiores serão priorizadas no empacotamento.

Os autores ainda reformularam o problema baseado nas seguintes idéias:

- Seja $e_m = \sqrt{a_m^2 + b_m^2 + c_m^2}$. Por (3.7), as restrições (3.6) podem ser reescritas como:

$$a_m x_i + b_m y_i + c_m z_i - d_m \geq e_m \sum_{k=1}^K r_k t_{ik}, \quad \forall i, \forall m. \quad (3.11)$$

Como o lado direito é não-negativo, a equação acima implica em (3.7), não havendo a necessidade de manter estas restrições no modelo.

- As variáveis binárias t_{ik} podem ser substituídas pelas seguintes desigualdades:

$$t_{ik}(t_{ik} - 1) \geq 0 \quad \text{e} \quad 0 \leq t_{ik} \leq 1, \quad (3.12)$$

como pode ser visto na Figura 3.9. Mas note que $t_{ik} \leq 1$ é implicado pelas restrições (3.8).

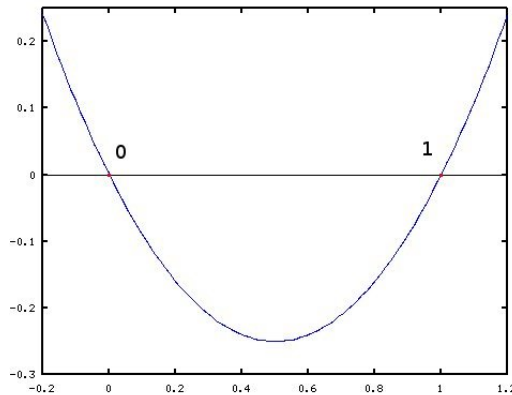


Figura 3.9: Variável binária transformada em contínua.

Com isso, temos a reformulação como em (3.13) e (3.18):

$$\max \sum_{i=1}^L \sum_{k=1}^K r_k^3 t_{ik} \quad (3.13)$$

$$\text{s. a } \|(x_i, y_i, z_i) - (x_j, y_j, z_j)\|^2 \geq \left(\sum_{k=1}^K r_k t_{ik} + \sum_{k=1}^K r_k t_{jk} \right)^2, \quad \forall i \leq j, i \neq j \quad (3.14)$$

$$a_m x_i + b_m y_i + c_m z_i - d_m \geq e_m \sum_{k=1}^K r_k t_{ik}, \quad \forall i, \forall m \quad (3.15)$$

$$t_{ik}(t_{ik} - 1) \geq 0, \quad \forall i, \forall k \quad (3.16)$$

$$\sum_{k=1}^K t_{ik} \leq 1, \quad \forall i \quad (3.17)$$

$$\sum_{i=1}^L t_{ik} \leq q_k, \quad \forall k \quad (3.18)$$

$$t_{ik} \geq 0, \quad \forall i, \forall k \quad (3.19)$$

Note que a constante $(4/3)\pi$ foi omitida na função objetivo e que as restrições (3.14) e (3.16) não formam conjuntos convexos. As esferas que não fazem parte da solução terão suas variáveis associadas $t_{ik} = 0$, para todo k .

Capítulo 4

Uma Aplicação: o Gamma Knife

4.1 Introdução

Nesta seção, discutiremos uma característica do problema (P) e mostraremos que não está, ainda, bem definido. Primeiro, observe que a única condição ao recobrimento é que se utilize a menor quantidade possível de esferas. Dessa forma, haverá sempre uma solução ótima contendo apenas esferas de maior raio disponível, ou seja, o maior raio pertencente ao conjunto R .

De fato, seja R_{max} o maior raio do conjunto R . Suponha que exista uma solução ótima para (P) contendo alguma esfera de raio $r \leq R_{max}$. Se substituirmos o raio dessa esfera por R_{max} , mantendo-a na mesma posição, não alteraremos a quantidade de esferas na solução e manteremos o recobrimento total de T , visto que essa nova esfera recobre a esfera anterior e, portanto, todos os pontos por ela recobertos. Assim, essa nova solução também será ótima. Isso pode ser visto na Figura 4.1.

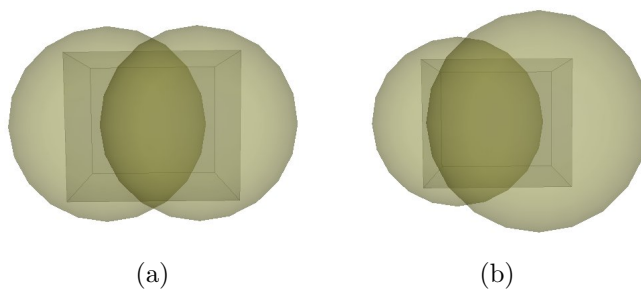


Figura 4.1: (a) Duas esferas de raio r recobrimo T , em cinza. Em (b), alterou-se o raio de uma das esfera para R_{max} , mantendo tanto a quantidade de esferas quanto o recobrimento total.

Da forma como o problema foi definido, dada uma solução ótima, se substituirmos o raio de qualquer esfera por um valor maior, a solução continuará sendo ótima. A questão natural, nesse momento, é identificar qual a melhor dentre

essas soluções. Ou, dada uma aplicação a esse problema, qual dentre essas soluções é a mais plausível. Portanto, apresentaremos, na seção 4.2, uma aplicação bastante encontrada na literatura para esse tipo de problema.

4.2 Uma aplicação: o *Gamma Knife*

Nesta seção, apresentaremos uma aplicação ao problema de recobrimento por esferas de diferentes tamanhos, a fim de melhor definir o conjunto de soluções ótimas.

Esse problema tem importantes aplicações em planejamentos de tratamentos por radiocirurgias estereotáxicas, sendo uma das modalidades mais efetivas dessa classe de cirurgias a radiocirurgia conhecida como *Gamma Knife* [14]. O *Gamma Knife* consiste em uma unidade de tratamento altamente especializada, direcionada a tratamentos de tumores cerebrais, malformações vasculares e distúrbios de dor na cabeça.

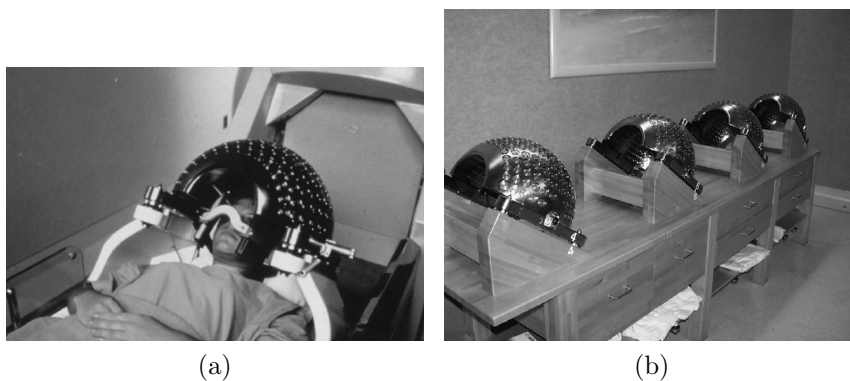


Figura 4.2: (a) Paciente na unidade de tratamento. (b) Diferentes tamanhos de elmos, resultando em 4 diferentes tamanhos de *shots*.

No interior da unidade de tratamento (ver Figura 4.2a), feixes de 201 fontes radioativas de Cobalto-60 são irradiados e focados em um único ponto no espaço. Esse procedimento é conhecido como um *shot* de radiação. Um único feixe não é suficiente para queimar consideravelmente o tecido. Porém, os 201 feixes se concentram ao redor do ponto focado, queimando um região do tecido ao seu redor. Logo, podemos considerar que a região afetada pelo *shot* é, aproximadamente, uma esfera centrada no foco de irradiação, como mostrado na Figura 4.3.

Um tratamento típico consiste de aplicações de *shots*, eventualmente de diferentes tamanhos (ver Figura 4.2b), posicionados em diferentes pontos no tumor, cujo objetivo é recobrir o volume de tratamento com uma certa dose de radiação, minimizando o efeito nos tecidos saudáveis ao redor.

Em alguns casos, o processo de planejamento do tratamento é relativamente fácil. Por exemplo, algumas lesões pequenas podem ser recobertas com apenas um

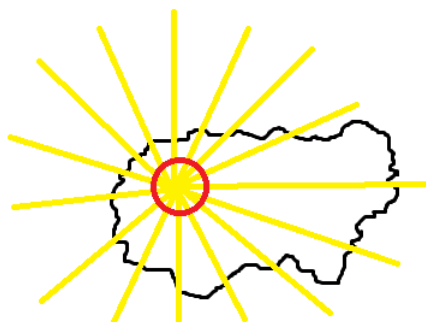


Figura 4.3: Um *shot* de radiação e a aproximação esférica da região afetada.

shot de radiação e o planejamento estará completo em minutos. Infelizmente, esse processo pode se tornar muito mais complexo quando o volume do tumor é grande ou de formato irregular. Estes casos requerem o uso de mais *shots* de radiação.

No equipamento afixado à cabeça do paciente, como mostrado na Figura 4.2a, podem ser anexados 4 capacetes que se diferenciam no tamanho, como pode ser visto na Figura 4.2b. Esses capacetes são responsáveis pelos tamanhos dos *shots*. Portanto, quatro tamanhos diferentes de *shots* estão disponíveis, afetando regiões aproximadamente esféricas com raios de 2mm, 4mm, 7mm e 9mm. Combinando múltiplos *shots* de radiação, um plano de tratamento pode ser realizado, de forma a tratar lesões de diferentes tamanhos e formas. Contudo, esse procedimento geralmente resulta em um recobrimento não-homogêneo, devido à sobreposição dos diferentes *shots*. Na prática, muitos casos são tratados com o uso de 1 a 15 *shots* de radiação [15, 16], embora um número maior que 10 seja considerado impraticável pelas rotinas clínicas de tratamento.

Toda vez que há uma mudança do tamanho do *shot*, deve-se retirar o capacete em uso e substituí-lo com o capacete de tamanho correto. Esse é um processo que despence bastante tempo, devido ao alto peso dos capacetes, devendo ser manipulados com o auxílio de máquinas. Além disso, após cada *shot*, deve-se ajustar a conexão entre o capacete e o equipamento afixado na cabeça do paciente, para que não haja erro de precisão na localização do *shot*. Para muitos pacientes, esse processo torna-se bastante tedioso e demorado. Além disso, a qualidade do plano de tratamento depende altamente da experiência e paciência dos profissionais responsáveis.

Por essas razões, há o desejo de automatização do processo de planejamento de tratamento via *Gamma Knife*, objetivando encontrar um bom recobrimento do volume do tumor e acelerar o processo de planejamento do tratamento. Otimizar a quantidade de *shots*, a posição dos seus centros e cada um de seus tamanhos pode reduzir as não-homogeneidades e evitar que tecidos saudáveis sejam afetados pela irradiação, além de alcançar o recobrimento desejado. Abordagens para esse

problema podem ser encontradas em [17] [18] [19].

Na seção 4.3, introduziremos novas informações ao problema, de forma a torná-lo mais compatível com a aplicação apresentada, a radiocirurgia *Gamma Knife*.

4.3 Ajustando o problema à aplicação

Nesta seção, acrescentaremos informações ao problema de forma a torná-lo mais semelhante com o problema de planejamento do tratamento por radiocirurgia *Gamma Knife*, apresentado na seção 4.2.

Seja T a estrutura a ser recoberta, representando o tumor. Assumiremos que os *shots* de radiação podem perfeitamente ser modelados por esferas de raios 2mm, 4mm, 7mm e 9mm. Duas novas informações surgem da aplicação: não é desejável que as esferas ocupem a parte externa de T , visto que isso representa irradiação em tecidos saudáveis; deve-se evitar sobreposição excessiva das esferas, pois isso representa radiação dobrada na região de interseção entre as esferas. No exemplo citado no início deste capítulo, teria-se preferência, então, pela solução representada na Figura 4.1a.

Uma característica importante que deve ser observada com a introdução dessas informações ao problema é o fato de que, agora, nem sempre é possível recobrir completamente a região T . Não permitir que as esferas ocupem a parte externa de T , não permitir excesso de sobreposições e, ainda assim, recobrir T completamente são objetivos conflitantes.

Dessa forma, temos quatro metas a serem cumpridas:

- (a) maximizar o recobrimento;
- (b) minimizar a quantidade de esferas na solução;
- (c) minimizar o volume das esferas na região exterior a T ; e
- (d) minimizar a sobreposição das esferas.

Muitos autores atacam o problema de planejamento do tratamento da radiocirurgia *Gamma Knife* sob o ponto de vista de um problema de empacotamento, não permitindo que as esferas se sobreponham, levando ao extremo a condição (d). Dessa forma, o problema é descrito como um modelo matemático de empacotamento de esferas em regiões tridimensionais, almejando uma densidade de empacotamento superior a 0.9 [13]. Como já observado no Capítulo 3, o empacotamento de esferas idênticas em um espaço euclidiano tridimensional ocupa, no máximo, 74% desse espaço. Como o tratamento em questão disponibiliza quatro diferentes tamanhos de esferas, uma densidade mais alta é possível de ser obtida.

Contudo, desejamos, nesse trabalho, priorizar as condições (a) e (b), visando recobrir ao máximo possível a região T com o mínimo possível de esferas. Assim, o enfoque é mais matemático, direcionado ao problema original (P). No capítulo 5, proporemos um modelo para o problema de recobrimento incorporando características relacionadas ao tratamento por *Gamma-Knife*.

Capítulo 5

Modelo Proposto

Neste capítulo, proporemos um modelo para o problema de recobrimento de uma região tridimensional por esferas de raios diferentes, inspirado no problema prático de planejamento de tratamento pela radiocirurgia conhecida como *Gamma Knife*, conforme apresentado nas seções 4.2 e 4.3.

5.1 O Modelo

Começaremos com algumas definições. Seja S um conjunto de esferas, possivelmente de raios diferentes. Conforme (P), que denotaremos de problema *clássico*, para futuras referências no texto, o problema consiste em recobrir um sólido $T \subset \mathbb{R}^3$ com o menor número possível de esferas pertencentes a S . Ou, de outra forma, deve-se posicionar a menor quantidade possível de esferas de S de modo que cada ponto $p \in T$ pertença, também, a alguma dessas esferas. Importante ressaltar que, como explicado na seção 4.3, se considerarmos a aplicação apresentada, nem sempre será possível obter um recobrimento total. Propomos, então, uma descrição formal do problema, em termos de uma formulação de programação matemática.

Seja S um conjunto formado por n esferas, $S = \{S_1, \dots, S_n\}$. Assim, para cada $i \in \{1, \dots, n\}$, seja S_i uma esfera centrada em $x^i \in \mathbb{R}^3$ e com raio $r_i > 0$. Assumiremos que um recobrimento de T por esferas de S existe. Ou, de outra forma, assumiremos que há disponível em S esferas suficientes capazes de recobrir T . Seja y_i uma variável binária representando a utilização ou não da esfera S_i na solução. Temos, então, as seguintes variáveis:

- $x \in T^n$, e
- $y \in \{0, 1\}^n$, sendo $y_i = \begin{cases} 1, & \text{se a esfera } i \text{ está sendo utilizada} \\ 0, & \text{caso contrário} \end{cases}$, $\forall i$.

A variável $x = \{x^1, \dots, x^n\}$ representa um vetor contendo o centro de cada esfera e a variável binária $y = \{y_1, \dots, y_n\}$ determina quais esferas do conjunto S estão presentes na solução.

A partir da idéia de dualidade em programação matemática, montaremos um modelo bastante parecido com um modelo de empacotamento, mas com as restrições de não-sobreposição relaxadas, de forma a possibilitar um recobrimento. Tal relaxação é controlada pelos parâmetros α_{ij} , com $1 \leq i < j \leq n$, conforme representado na Figura 5.1.

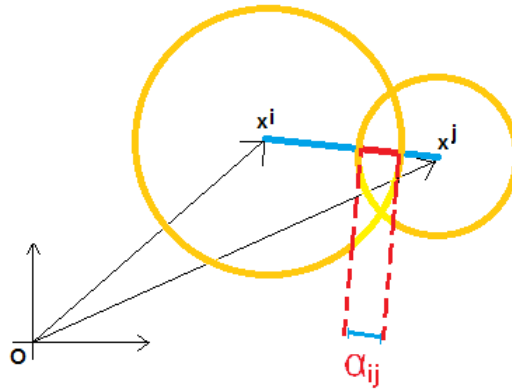


Figura 5.1: Representação dos parâmetros α_{ij} , que correspondem à máxima interpenetração das esferas.

Ao invés de impormos que a distância entre os centros de duas determinadas esferas seja maior que a soma de seus raios, como nas equações (3.5) e (3.14) dos modelos de empacotamento, permitiremos que essas esferas se sobreponham a uma quantidade máxima α_{ij} .

Definiremos α_{ij} como segue:

$$\alpha_{ij} = \beta \cdot \min\{r_i, r_j\}, \quad 0 \leq \beta < 1. \quad (5.1)$$

Para duas esferas S_i e S_j , a distância mínima entre seus centros é, então, dada por:

$$d_{ij} = r_i + r_j - \alpha_{ij} \quad (5.2)$$

Assim, as restrições do modelo serão:

$$\|x^i - x^j\| \geq d_{ij}, \quad \forall 1 \leq i < j \leq n. \quad (5.3)$$

Porém, como a função norma euclideana apresenta pontos de não-diferenciabilidade, elevaremos ambos os termos da inequação (5.3) ao quadrado. O lado esquerdo torna-se então uma função de classe C^∞ .

Ainda, essas restrições devem ser aplicadas somente a pares de esferas S_i e S_j que estiverem presentes na solução. A posição das esferas que não participam do recobrimento é irrelevante. Caso considerássemos todas as esferas, o problema facilmente tornaria-se inviável. Logo, substituiremos as restrições (5.3) por:

$$\|x^i - x^j\|^2 \geq d_{ij}^2 (y_i + y_j - 1), \quad \forall 1 \leq i < j \leq n. \quad (5.4)$$

Quando ambas as esferas S_i e S_j estiverem sendo utilizadas, teremos $y_i = y_j = 1$. Substituindo na inequação acima, obteremos a restrição desejada (5.3). Para os outros casos, teremos:

- apenas $y_i = 1$: $\|x^i - x^j\|^2 \geq 0$;
- $y_i = y_j = 0$: $\|x^i - x^j\|^2 \geq -d_{ij}^2$.

Visto que a norma assume sempre um valor não-negativo, a restrição continuará válida, apenas tornando-se redundante.

Por fim, tomemos os parâmetros c_i como sendo o custo de cada esfera na solução. Como temos por objetivo a menor quantidade possível de esferas na solução, priorizaremos as esferas de maior raio, atribuindo a elas um custo maior, visto que a formulação será um problema de maximização.

Propomos, então, a seguinte formulação:

$$\max \sum_{i=1}^n c_i y_i \quad (5.5)$$

$$\text{s. a } \|x^i - x^j\|^2 \geq d_{ij}^2 (y_i + y_j - 1), \quad \forall 1 \leq i < j \leq n \quad (5.6)$$

$$x \in T^n \quad (5.7)$$

$$y \in \{0, 1\}^n \quad (5.8)$$

A partir dessa formulação, escrevemos o seguinte teorema, que define a existência de um conjunto de valores para os parâmetros, com os quais podemos retomar a solução do problema clássico:

Teorema. Existe um conjunto de parâmetros $\alpha_{ij} \geq 0$, com $1 \leq i < j \leq n$, e um conjunto de valores $c_i \geq 0$, com $1 \leq i \leq n$, para os quais a solução ótima de (5.5)-(5.8) é, também, solução ótima para o problema clássico de recobrimento (P).

Demonstração: Seja (x^*, y^*) uma solução ótima do problema clássico de recobrimento. Para cada par de esferas da solução, calculemos a distância D_{ij} entre seus centros:

$$D_{ij} = \|(x^i)^* - (x^j)^*\|, \quad \forall 1 \leq i < j \leq n.$$

Se $D_{ij} > r_i + r_j$, tome $\alpha_{ij} = 0$. Caso contrário, tome

$$\alpha_{ij} = r_i + r_j - D_{ij}.$$

Se uma dada esfera k não está sendo utilizada, ou seja, se $(y_k)^* = 0$, atribua qualquer valor permitido para os parâmetros $\{\alpha_{ij}\}$ a ela relacionados. Por exemplo, atribua valor zero.

Tome $c_i = y_i^*$, para todo $1 \leq i \leq n$. Então, afirmamos que (x^*, y^*) é uma solução ótima do problema (5.5)-(5.8), com os parâmetros $\{\alpha_{ij}\}_{1 \leq i < j \leq n}$ e $\{c_i\}_{1 \leq i \leq n}$ como acabamos de definir.

De fato, utilizando os parâmetros agora escolhidos, é fácil verificar que a solução (x^*, y^*) também será viável em (5.5)-(5.8). Basta mostrar, então, que ela terá o melhor valor da função objetivo. De fato, como os y_i^* assumem valores 0 ou 1, para todo i , e estamos usando $c_i = y_i^*$, temos que

$$\sum_{i=1}^n c_i y_i = \sum_{i|y_i^*=1} y_i.$$

Podemos observar que esse somatório atingirá seu valor máximo se tomarmos $y_i = 1$ quando $y_i^* = 1$, ou seja, quando tivermos $y_i = y_i^*$. Portanto, (x^*, y^*) é uma solução ótima de (5.5)-(5.8) com os parâmetros tais como definidos. ■

O modelo apresentado é um modelo de programação matemática não-linear inteira mista (MINLP) cuja relaxação contínua é não-convexa, devido às restrições (5.6). Embora alguns métodos de solução existam, essa classe de problemas está entre as mais difíceis de serem resolvidas. O estado da arte de tais métodos [20] não está em um estágio onde instâncias diversas podem ser resolvidas efetivamente ou asseguradamente.

Ainda, a menos que seja possível escrever matematicamente a restrição (5.7), não é possível resolver diretamente o modelo proposto. Há casos mais simples, como apresentado anteriormente, com a representação de T através de um politopo, em que fazemos uso de restrições como (3.7). Um exemplo simples de politopo é o

paralelepípedo. Para essa forma geométrica, temos:

$$T_{\text{paralelepipedo}} = \{ (x_1, x_2, x_3) \in \mathbb{R}^3 \mid 0 \leq x_k \leq b_k, \quad 1 \leq k \leq 3 \}$$

ou translações e rotações desses, sendo b_1, b_2 e b_3 as medidas dos lados do paralelepípedo.

Em especial, muitos trabalhos relacionados ao tratamento por *Gamma Knife* aproximam a região T , correspondente ao tumor cerebral, por elipsóides. Utilizando um modelo bastante semelhante ao aqui proposto, DO NASCIMENTO *et al.* [21][22] trabalharam no recobrimento de elipsóides a partir de um conjunto de esferas de diferentes diâmetros. Os autores apresentam uma reformulação como um problema de programação geométrica signomial. Não utilizam técnicas de otimização discreta, tratando até mesmo os raios das esferas de forma contínua. Nesses casos, temos:

$$T_{\text{elipsoide}} = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 \mid \sum_{k=1}^3 \left(\frac{x_k}{b_k} \right)^2 \leq 1 \right\}$$

ou translações e rotações desses, sendo b_1, b_2 e b_3 os raios do elipsóide.

Na Seção 5.2, acrescentaremos mais informações ao modelo, a fim de torná-lo mais compatível com a aplicação ao tratamento de tumores.

5.2 A Região de Segurança

Na aplicação apresentada, a radiocirurgia conhecida com *Gamma Knife*, é necessário uma medida de segurança aos tecidos saudáveis ao redor do tumor. Não é aconselhável que esses tecidos recebam uma quantidade considerável de radiação. É preciso restringir ainda mais o domínio dos centros das esferas, em (5.7), a fim de respeitar a condição (c), na Seção 4.3, segundo a qual devemos minimizar o volume das esferas na região exterior a T .

A fim de evitar que a solução do nosso modelo contenha esferas posicionadas de forma a danificar tecidos saudáveis, utilizaremos um novo parâmetro ε , como pode ser visto na Figura 5.2. Nos referiremos a essa região como *região de segurança* (RS) e a definiremos como segue:

$$RS = \{ x \in \mathbb{R}^3 : \exists y \in T \mid x \in B(y, \varepsilon) \}. \quad (5.9)$$

Ou, em outras palavras, um ponto $x \in \mathbb{R}^3$ pertencerá também a RS caso exista algum ponto em T distando menos que ε de x . Ou, ainda:

$$x \in RS \iff d(x, T) \leq \varepsilon \iff \exists y \in T \mid d(x, y) \leq \varepsilon. \quad (5.10)$$

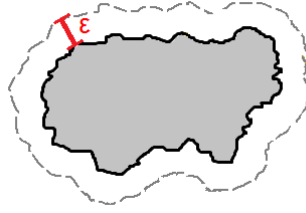


Figura 5.2: Representação da região de segurança. A região em cinza representa o volume a ser recoberto.

Idealmente, a fronteira da região de segurança deve representar o lugar geométrico dos pontos que distam ε do conjunto T , em sua parte externa. Porém, devido à dificuldade em representar esses pontos matematicamente, para os casos específicos de paralelepípedos e elipsóides, como nos exemplos apresentados anteriormente, utilizaremos as seguintes aproximações:

$$RS_{paralelepipedo} = \{ (x_1, x_2, x_3) \in \mathbb{R}^3 \mid -\varepsilon \leq x_k \leq b_k + \varepsilon, \quad 1 \leq k \leq 3 \} \quad \text{e}$$

$$RS_{elipsoide} = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 \mid \sum_{k=1}^3 \left(\frac{x_k}{b_k + \varepsilon} \right)^2 \leq 1 \right\},$$

ou translações e rotações desses, sendo b_1, b_2 e b_3 as medidas dos lados do paralelepípedo ou os raios do elipsóide.

O domínio das variáveis x^i será, então, restringido, de acordo com a região de segurança. Isso pode ser visto na Figura 5.3. Em sua forma atual, podemos encontrar, no modelo, características tanto do problema de recobrimento quanto do problema de empacotamento. Características do primeiro aparecem com relação ao volume T , que desejamos recobrir. Para o empacotamento, basta considerar a região de segurança como um contêiner, onde devem estar contidas as esferas.

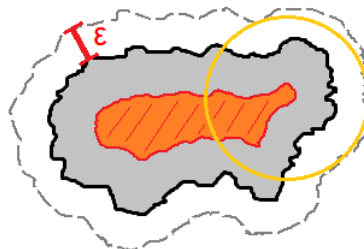


Figura 5.3: A região em cinza representa o volume a ser recoberto. Como as esferas devem estar contidas na região de segurança, o domínio da variável que representa o seu centro é restrito à região em laranja.

Nos caso de paralelepípedos ou de elipsóides, substituiremos, respectivamente, as restrições (5.7) por:

$$-\varepsilon + r_i \leq x_k^i \leq b_k + \varepsilon - r_i, \quad \forall i \in \{1, \dots, n\} \quad (5.11)$$

$$\sum_{k=1}^3 \left(\frac{x_k^i}{b_k + \varepsilon - r_i} \right)^2 \leq 1, \quad \forall i \in \{1, \dots, n\} \quad (5.12)$$

ou, novamente, por respectivas translações e rotações desses.

No Capítulo 9, apresentaremos uma idéia de discretização para trabalhar com instâncias onde a região a ser recoberta não possui uma equação matemática simples que a descreva. Em particular, no problema do tratamento por *Gamma Knife*, a região do tumor provavelmente terá um formato irregular. Sua obtenção é através de imagens de ressonância magnética ou de tomografias computadorizadas. Os dados dessas imagens são de natureza discreta e justificam um estudo nessa direção.

5.3 Melhorias no modelo

Nesta seção, tentaremos reduzir o tamanho da combinatória do problema. Analisando somente as variáveis binárias e a árvore de enumeração de soluções, o problema de recobrimento em estudo admite muitas soluções semelhantes. Veja exemplo na Figura 5.4.

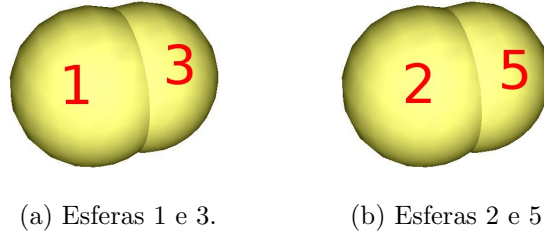


Figura 5.4: Soluções distintas que apresentam mesmos raios e centros.

Como exemplo, em uma instância onde são disponibilizadas 10 esferas de mesmo raio para o recobrimento e a solução utiliza 4 dessas esferas, teríamos $10 \cdot 9 \cdot 8 \cdot 7 = 5040$ soluções que podem ser consideradas idênticas. Ordenaremos, portanto, as esferas de mesmo raio e acrescentaremos mais um conjunto de restrições ao modelo, como segue:

$$y_i \geq y_{i+1}, \quad \forall i \mid r_i = r_{i+1}. \quad (5.13)$$

Dessa forma, as esferas de menores índices serão utilizadas primeiro na formação da solução.

No exemplo da Figura 5.4, que mostra duas soluções equivalentes, uma com as esferas indexadas por 1 e 3 e outra com as esferas indexadas por 2 e 5, a solução que obteríamos ao empregar as equações (5.13) seria a que usa as esferas indexadas por 1 e 2.

Por fim, levando em consideração as variáveis contínuas do problema, que representam os centros das esferas, ainda poderíamos ter soluções semelhantes, como mostrado na Figura 5.5.

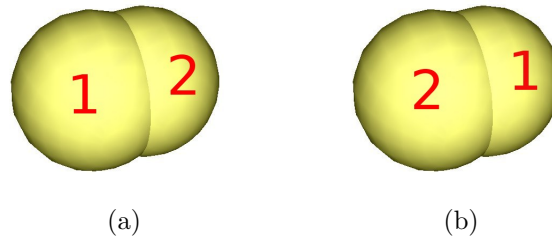


Figura 5.5: Soluções distintas que apresentam mesmos raios e centros.

Adicionaremos, então, a seguinte família de restrições ao modelo:

$$x_1^i \leq x_1^{i+1}, \quad \forall i \mid r_i = r_{i+1} . \quad (5.14)$$

Considerando o sentido em que lemos o texto como o sentido positivo do eixo x do sistema de coordenadas, ao utilizarmos as equações (5.14), a solução encontrada será a representada na Figura 5.5a.

Capítulo 6

Alguns Resultados

Nesta seção, serão apresentados alguns resultados obtidos para o problema de recobrimento, com a utilização do modelo não-linear e não-convexo proposto (5.5)-(5.8). O estado da arte dos algoritmos para essa classe de problemas ainda não é satisfatório. Dentre os solvers comerciais existentes, testamos o COUENNE [23, 24], direcionado à resolução de problemas de programação matemática não-linear e não-convexos. Testamos, ainda, o BONMIN [25, 26] e o XPRESS [27, 28], cujos algoritmos permitem dar uma solução aproximada para essa classe de problemas. Com a finalidade de visualização e melhoria manual das soluções, utilizamos um programa por nós desenvolvido, de onde foram retiradas todas as imagens que aparecem no restante deste trabalho.

Ressaltamos que um recobrimento total é possível de ser atingido, desde que as condições (c) e (d), da seção 4.3, sejam relaxadas. Como explicado anteriormente, essas condições são contrárias ao recobrimento. A condição (c) refere-se à ocupação das esferas na região externa a T e a condição (d) refere-se à sobreposição das esferas. Ou seja, através da notação usada neste texto, conseguiremos um recobrimento total desde que os parâmetros ϵ e os parâmetros α sejam grande o suficiente. O primeiro define o tamanho da região de segurança, enquanto o segundo define a máxima interseção permitida entre cada par de esferas.

O problema de recobrimento em estudo apresenta uma certa dificuldade quanto à qualificação das soluções. Sabemos que, dado um modelo de programação matemática, podemos comparar soluções através dos seus respectivos valores da função objetivo. Porém, com relação à aplicação, o tratamento via *Gamma Knife*, ao analisarmos duas soluções distintas de uma mesma instância do problema, não é evidente dizer qual apresenta melhor qualidade. Na verdade, não há, ainda, um critério de comparação entre soluções.

Como exemplo, sejam duas soluções com um mesmo número de esferas no recobrimento. A primeira pode apresentar um percentual de recobrimento maior, mas obtido através de uma maior ocupação das esferas na parte externa da região-

alvo ou mesmo através de um volume muito grande de interseções entre esferas. Não é trivial julgar, matematicamente, qual solução é mais adequada. Assim, introduziremos três grandezas, como em WEI *et al.* [29], cuja finalidade é analisar melhor as características da solução. Seja T o volume a ser recoberto pelas esferas e U o volume da união das esferas. Calcularemos:

- ‘**cov**’ : Percentual do volume de T recoberto pelas esferas;
- ‘**overlap**’ : Percentual do volume de T recoberto por mais de uma esfera; e
- ‘**miscov**’ : Percentual do volume de U localizado na parte externa de T .

Essas grandezas podem ser melhor entendidas com o auxílio da Figura 6.1.

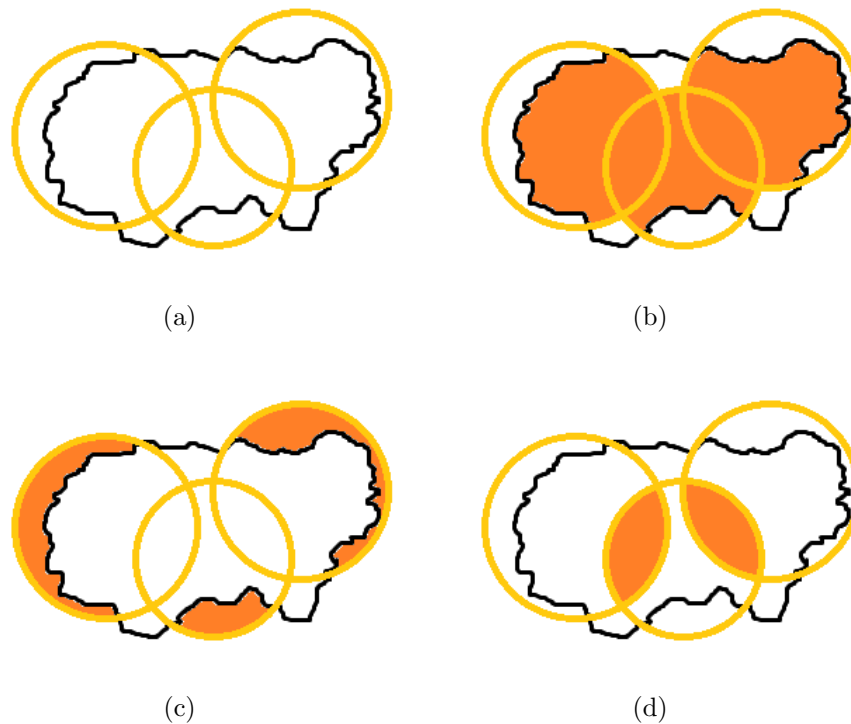


Figura 6.1: Exemplo bidimensional mostrando as regiões que correspondem às grandezas utilizadas na descrição de soluções. Em (a), a região a ser recoberta e as bolas presentes no recobrimento. As regiões hachuradas representam os parâmetros (b) cov, (c) miscov e (d) overlap.

Para o cálculo dessas grandezas, podemos utilizar ou uma malha fina ou um algoritmo probabilístico conhecido como método de Monte Carlo [30]. O método, voltado ao cálculo do volume de uma região dada, consiste na geração aleatória de pontos e no cálculo da proporção de pontos gerados pertencentes à região. O método pode ser estendido para o cálculo do volume da união de vários objetos. Neste trabalho, os objetos são esferas de diferentes raios.

Para o cálculo do percentual do volume de T coberto pelas esferas, representado por ‘cov’, geramos uma malha de pontos em T e calculamos a proporção de pontos que pertencem a pelo menos uma das esferas. Esse mesmo algoritmo pode ser aproveitado para o cálculo do volume das interseções das esferas no interior de T , representado por ‘overlap’, verificando apenas se o ponto pertence a mais de uma esfera.

Para calcular o volume percentual de U externo a T , que chamamos de ‘miscov’, primeiro geramos uma malha circular em cada uma das esferas e repetimos o mesmo procedimento, a fim de obter o volume que cada esfera ocupa externamente a T . Após a soma desses valores, teremos o volume da união das esferas na parte externa a T . O volume interno da união pode ser calculado como segue:

$$Vol_{interno} = cov * Vol(T) .$$

Fazemos

$$Vol_{Total} = Vol_{Interno} + Vol_{Externo}$$

e, por fim,

$$miscov = \frac{Vol_{externo}}{Vol_{Total}} .$$

Atualmente, não há, para esse tipo de problema, um repositório de instâncias com o qual se possa trabalhar e realizar testes. Dessa forma, utilizamos, em nossos testes, um paralelepípedo de dimensões 14mm, 12mm e 10mm, como na Figura 6.2. Para a região de segurança, atribuímos ao parâmetro correspondente o valor $\varepsilon = 1$. Três conjuntos de esferas foram disponibilizados, separados em três diferentes conjuntos de dados, conforme Tabela 6.1.

	Dados1	Dados2	Dados3
$ S $	15	30	40
$r = 4\text{mm}$	6	7	7
$r = 2\text{mm}$	9	23	33

Tabela 6.1: Diferentes conjuntos de esferas disponibilizadas.

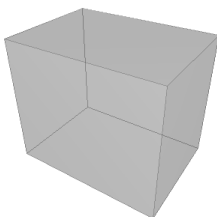


Figura 6.2: Paralelepípedo de dimensões 14mm, 12mm e 10mm, usado como região a ser recoberta.

Para os outros parâmetro do modelo, consideramos, nos testes, os seguintes valores:

$$c_i = r_i^3$$

e

$$\alpha_{ij} = \frac{1}{2} \min\{r_i, r_j\}.$$

Os c_i são os custos na função objetivo e representam o peso de cada esfera no recobrimento. Como queremos minimizar a quantidade de esferas na solução, maior peso é dedicado às esferas de maior volume, pois recobrem um volume maior. E, como o objetivo é o recobrimento, maximizaremos a densidade do empacotamento, como definida na equação 3.1. Dessa forma, atribuímos aos custos os valores dos volumes das esferas, com omissão da constante $(4/3)\pi$ e omissão do volume do contêiner.

Os testes foram realizados em uma máquina Intel Xeon CPU X5675 3.07GHz, com 48 GB de memória RAM e um único processador de 6 *cores* e 12 *threads*.

6.1 COUENNE

O COUENNE (*Convex Over and Under ENvelopes for Nonlinear Estimation*) [23, 24] é um código open-source para resolver problemas de otimização global da forma

$$\min f(x) \tag{6.1}$$

$$\text{s. a } g_j(x) \leq 0 \quad \forall j \in M \tag{6.2}$$

$$x_i^l \leq x_i \leq x_i^u \quad \forall i \in N_0 \tag{6.3}$$

$$x_i \in \mathbb{Z} \quad \forall i \in N_0^I \subseteq N_0, \tag{6.4}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ e, para todo $j \in M$, $g_j : \mathbb{R}^n \rightarrow \mathbb{R}$ são funções multivariadas, possivelmente não-convexas.

O COUENNE consiste no algoritmo conhecido como *spatial Branch and Bound*, onde cada problema é subdividido através das equações 6.3. Ou seja, as ramificações na árvore do *Branch and Bound* são realizadas por divisões nos domínios das variáveis. O pacote implementa técnicas de linearização, técnicas de ramificação, heurísticas para encontrar soluções viáveis e técnicas de melhoria de limites inferiores para o problema. O problema inicial é reformulado, introduzindo novas variáveis chamadas de variáveis auxiliar. A reformulação não torna o problema mais fácil de ser resolvido, mas permite obter melhores limites inferiores para o problema. Em

seguida, através de linearizações, obtém-se uma relaxação linear onde facilmente aplica-se o algoritmo de *Branch and Bound*.

Aplicamos o modelo proposto (5.5)-(5.8) ao COUENNE para os conjuntos de esferas presentes em Dados1, Dados2 e Dados3. Forçamos a parada do algoritmo após 3600 segundos de execução. As melhores soluções encontradas estão representadas na Tabela 6.2.

	Dados1	Dados2	Dados3
z^*	264	144	?
$ S $	12	4	?
t	950	1700	?
cov	59.21	30.51	?
miscov	4.48	6.94	?
overlap	0.95	1.54	?

Tabela 6.2: Soluções encontradas pelo COUENNE.

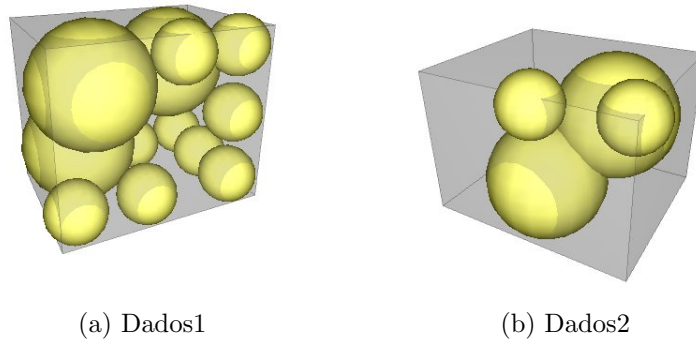


Figura 6.3: Imagens das soluções encontradas pelo COUENNE.

X	Y	Z	Raio
-0.334272	0.249808	0.171778	0.4
0.368994	0.272942	-0.169904	0.4
0.366087	-0.266802	0.172442	0.4
-0.537980	-0.450674	-0.364987	0.2
-0.551223	-0.449327	0.005482	0.2
0.086532	-0.451458	-0.353585	0.2
-0.151779	-0.454847	0.358583	0.2
-0.554228	0.008047	-0.351633	0.2
-0.560350	-0.412196	0.369279	0.2
0.562720	-0.445903	-0.352377	0.2
-0.151325	0.462952	-0.360341	0.2
-0.563514	0.460728	-0.340447	0.2

Tabela 6.3: Resultado do COUENNE para o conjunto Dados1.

X	Y	Z	Raio
-0.035156	-0.5	-0.4	0.2
-0.6	-0.37557	-0.4	0.2
-0.33348	0.0	-0.2	0.4
0.0	-0.3	0.2	0.4

Tabela 6.4: Resultado do COUENNE para o conjunto Dados2.

Na Tabela 6.2, o valor da função objetivo está representado por z^* . O tempo t em que a melhor solução foi encontrada é dado em segundos e $|S|$ representa a quantidade de esferas na solução. A terceira coluna não apresenta resultados porque o COUENNE não foi capaz de encontrar uma solução no tempo especificado.

Nas Tabelas 6.3 e 6.4, os valores apresentados estão todos divididos por 10, estando o centro geométrico do paralelepípedo na origem.

Se, ao invés de utilizarmos as restrições 5.6, trabalharmos o modelo com as seguintes restrições:

$$\|x^i - x^j\| \geq (r_i + r_j - \alpha_{ij})(y_i + y_j - 1), \quad \forall 1 \leq i < j \leq n,$$

ou seja, não utilizarmos o quadrado da norma, então, mesmo após 1 hora de execução do algoritmo, o resolvidor não conseguiu encontrar qualquer solução viável. Foi utilizado o conjunto Dados1 nesse teste.

6.2 BONMIN

O BONMIN (*B*asic *O*pen-source *N*onlinear *M*ixed *I*Nteger programming) [25, 26] é um código experimental para resolver problemas gerais de programação não-linear inteira mista (MINLP) da forma

$$\min f(x) \tag{6.5}$$

$$\text{s. a } g_L \leq g(x) \leq g_U \tag{6.6}$$

$$x_L \leq x \leq x_U \tag{6.7}$$

$$x_i \in \mathbb{Z} \quad \forall i \in I \tag{6.8}$$

$$x_i \in \mathbb{R} \quad \forall i \notin I \tag{6.9}$$

onde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ e $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ são funções contínuas e duas vezes diferenciáveis e I é um subconjunto de $\{1, \dots, n\}$ formado pelos índices das variáveis inteiras. Os diferentes métodos que o BONMIN implementa são algoritmos exatos quando as funções f e g são convexas. Porém, quando alguma dessas funções é não-convexa, o BONMIN funciona como uma heurística.

Há várias escolhas possíveis de algoritmos que podem ser selecionados com o BONMIN:

- B-BB: é um algoritmo de *Branch and Bound* baseado em programação não-linear;
- B-OA: é um algoritmo de decomposição baseado em aproximação externa;
- B-QG: é uma implementação do algoritmo de *Branch and Cut* de Quesada e Grossman; e
- B-Hyb: é um algoritmo híbrido de *Branch and Cut* e aproximação externa.

Para MINLPs convexos, os autores realizaram experimentos com um conjunto grande de testes e o método B-Hyb conseguiu resolver a maioria dos problemas no tempo estabelecido e tornou-se o método padrão. Para MINLPs não-convexos, os autores recomendam o uso do método B-BB. Portanto, escolhemos esse método para testar o modelo que propomos.

Aplicamos o modelo proposto (5.5)-(5.8) ao BONMIN para os conjuntos de esferas presentes em Dados1, Dados2 e Dados3. As soluções encontradas estão representadas na Tabela 6.5.

	Dados1	Dados2	Dados3
z^*	328	-	-
$ S $	13	-	-
t	9	317	591
cov	70.22	-	-
miscov	5.44	-	-
overlap	3.46	-	-

Tabela 6.5: Soluções encontradas pelo BONMIN.

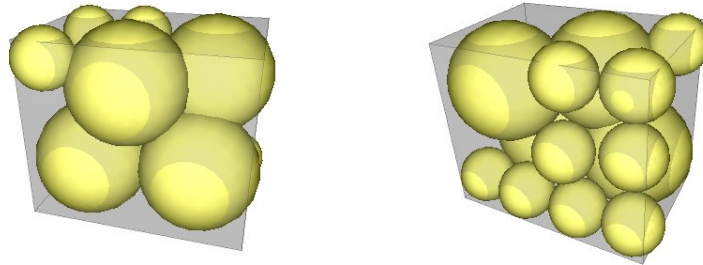


Figura 6.4: Imagens da solução encontrada pelo BONMIN.

Na Tabela 6.5, o valor da função objetivo está representado por z^* . O tempo t de execução do algoritmo é dado em segundos e $|S|$ representa a quantidade de esferas na solução.

Para os conjuntos de esferas disponibilizados em Dados2 e Dados3, o BONMIN retornou que o problema é inviável. Porém, claramente há solução, visto que as esferas de Dados1 são um subconjunto das esferas de Dados2 e também de Dados3. Logo, a solução encontrada para Dados1 é também solução, ao menos local, para Dados2 e Dados3.

Na Tabela 6.6, é apresentada a solução encontrada. Os valores estão todos divididos por 10, estando o centro geométrico do paralelepípedo na origem.

X	Y	Z	Raio
0.363273	-0.267663	0.178730	0.4
-0.374953	-0.282536	0.177565	0.4
-0.113281	0.280399	0.182765	0.4
0.387102	0.272057	-0.180515	0.4
-0.571201	-0.459726	-0.342947	0.2
-0.576943	0.461564	0.366366	0.2
-0.116019	-0.042294	-0.363792	0.2
0.576425	-0.446153	-0.342643	0.2
-0.188707	-0.468404	-0.350566	0.2
-0.567482	0.040927	-0.341714	0.2
0.220852	-0.462593	-0.354966	0.2
-0.564839	0.462235	-0.310491	0.2
-0.157085	0.468458	-0.363733	0.2

Tabela 6.6: Resultado do BONMIN para o conjunto Dados1.

6.3 Xpress-SLP

O Xpress-SLP [27, 28] é um resolvedor para problemas de programação não-linear. Ele implementa o método de programação linear sequencial, que foi desenvolvido a partir de técnicas utilizadas nas indústrias de processamento, e é capaz de resolver problemas grandes com milhares de variáveis. Também permite resolver problemas de programação não-linear inteira mista (MINLP).

A programação linear sequencial consiste em construir uma aproximação linear para o problema original não-linear, resolver essa aproximação até o valor ótimo e tentar validar esse resultado contra o problema original. Se a solução ótima da aproximação é suficientemente próxima de uma solução do problema original, então diz-se que o método convergiu e o procedimento termina. Caso contrário, uma nova aproximação é criada e o processo se repete. Em inglês, o método é conhecido como *successive linear programming* (SLP) ou *sequential linear programming*. Embora a solução seja o resultado de uma otimização da aproximação linear do problema, não

há uma garantia de que ela será uma solução ótima do problema original não-linear. Ela pode ser apenas um ótimo local.

Aplicamos o modelo proposto (5.5)-(5.8) ao XPRESS-SLP para os conjuntos de esferas presentes em Dados1, Dados2 e Dados3. As soluções encontradas estão representadas na Tabela 6.7.

	Dados1	Dados2	Dados3
z^*	328	440	504
$ S $	13	27	35
t	0	2	5
cov	63.71	80.67	89.15
miscov	8.83	9.87	9.39
overlap	5.27	10.44	16.59

Tabela 6.7: Soluções encontradas pelo XPRESS-SLP.

Nas Tabelas 6.8, 6.9 e 6.10, cada uma das soluções é apresentada. Os valores estão todos divididos por 10, estando o centro geométrico do paralelepípedo na origem.

X	Y	Z	Raio
0.0472136	0.3	-0.2	0.4
0.4	0.0251514	0.2	0.4
-0.4	0.3	0.2	0.4
-0.4	-0.26747	0.00512062	0.4
-0.6	-0.5	0.4	0.2
0.3	-0.5	-0.4	0.2
-0.6	-0.5	-0.4	0.2
0.6	-0.5	0.4	0.2
0.459524	0.5	-0.4	0.2
-0.6	0.265964	-0.4	0.2
0.6	-0.465964	0	0.2
-0.6	0.5	-0.212311	0.2
0.6	-0.5	-0.4	0.2

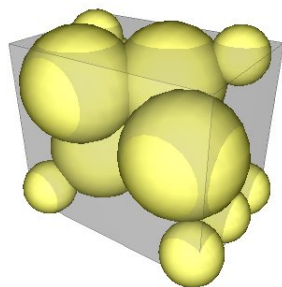
Tabela 6.8: Resultado do XPRESS-SLP para o conjunto Dados1.

X	Y	Z	Raio	X	Y	Z	Raio
0.0472136	0.3	-0.2	0.4	-0.427124	0.16188	-0.280325	0.2
-0.4	0.3	0.2	0.4	-0.0511685	-0.5	0.4	0.2
-0.4	-0.26747	0.00512058	0.4	0.3	0.5	0.4	0.2
0.286598	-0.0777651	0.2	0.4	-0.140864	-0.5	-0.4	0.2
0.6	0.2027	-0.4	0.2	-0.6	-0.0532789	-0.4	0.2
0.6	-0.5	0.4	0.2	0.539178	-0.202342	-0.4	0.2
0.6	0.5	0.4	0.2	0.299995	-0.5	-0.4	0.2
0.459524	0.5	-0.4	0.2	0.253273	-0.110101	-0.4	0.2
-0.6	-0.5	-0.4	0.2	0.6	-0.00399433	-0.182562	0.2
0.371546	-0.5	-0.103179	0.2	0.6	-0.5	-0.4	0.2
-0.6	-0.5	0.4	0.2	0.0318186	-0.5	0.109236	0.2
0.07943	-0.5	-0.190049	0.2	-0.365097	0.5	-0.4	0.2
0.6	-0.5	0.1	0.2	0.6	0.5	-0.134922	0.2
0.6	-0.297738	-0.121569	0.2				

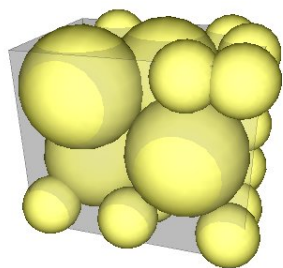
Tabela 6.9: Resultado do XPRESS-SLP para o conjunto Dados2.

X	Y	Z	Raio	X	Y	Z	Raio
-0.4	0.3	0.0505912	0.4	0.490296	0.5	-0.4	0.2
-0.4	-0.2811	0.2	0.4	-0.103498	0.5	0.4	0.2
0.185387	0.194555	0.129354	0.4	0.565512	0.374154	0.4	0.2
0.046814	-0.3	-0.2	0.4	-0.6	0.0839634	-0.353548	0.2
-0.374099	-0.5	-0.4	0.2	-0.223471	0.174282	-0.4	0.2
0.509329	-0.39907	0.132433	0.2	0.00259497	-0.5	0.4	0.2
0.6	-0.5	-0.135134	0.2	0.6	0.383104	-0.146425	0.2
-0.361547	-0.0920546	-0.4	0.2	-0.302452	0.5	-0.4	0.2
-0.6	-0.5	-0.202595	0.2	0.595385	0.219019	-0.397532	0.2
0.6	-0.5	0.4	0.2	0.198268	0.431299	-0.4	0.2
0.534536	-0.208448	-0.108448	0.2	0.401672	-0.0100423	-0.4	0.2
0.6	0.0761434	0.4	0.2	0.051734	-0.204052	0.4	0.2
0.6	-0.115469	0.169165	0.2	0.459125	-0.5	-0.4	0.2
-0.6	-0.274099	-0.4	0.2	-0.6	0.5	-0.361719	0.2
0.6	0.5	0.129864	0.2	0.6	-0.235134	-0.4	0.2
0.413329	-0.158705	0.4	0.2	0.239405	-0.5	0.215823	0.2
0.0760494	0.157324	-0.4	0.2	0.6	0.083722	-0.127184	0.2
-0.0473791	0.5	-0.242084	0.2				

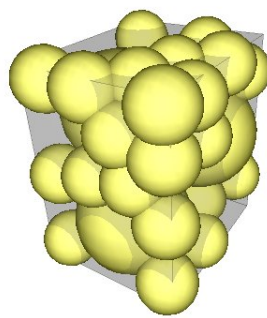
Tabela 6.10: Resultado do XPRESS-SLP para o conjunto Dados3.



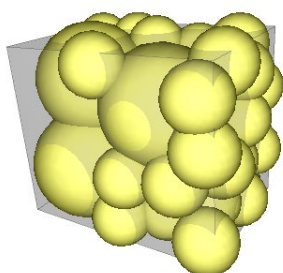
(a) Dados1



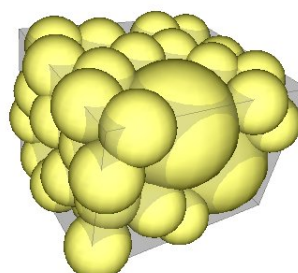
(b) Dados2



(c) Dados2



(d) Dados3



(e) Dados3

Figura 6.5: Imagens das soluções encontradas pelo XPRESS-SLP.

6.4 Análise dos resultados

Dentre os algoritmos testados, o COUENNE é o único com o objetivo de resolver problemas de programação matemática não-lineares e não-convexos. Porém, seus resultados foram insatisfatórios com relação ao tempo de execução, como pode-se constatar pela Tabela 6.2. Isso se deve ao tamanho da árvore gerada pelo *Spatial Branch and Bound*, não sendo capaz de encontrar boas soluções em tempos razoáveis.

O BONMIN apresenta uma certa inconsistência nas soluções. A solução encontrada por esse resolvedor para o conjunto de esferas Dados1, como na Tabela 6.5, é também uma solução viável para o problema com as esferas de Dados2 e Dados3. Apesar disso, o resolvedor retorna que o problema é inviável nesses dois últimos casos e diminui a confiabilidade nos resultados.

Por fim, o XPRESS-SLP apresentou soluções em tempos razoavelmente pequenos. Além disso, ao adicionarmos esferas ao conjunto S de esferas disponíveis, ou seja, ao passarmos de Dados1 para Dados2 e para Dados3, os resultados foram coerentes, apresentando um aproveitamento dessas esferas adicionais e, conseqüentemente, aumento no valor da função objetivo e do percentual de recobrimento, conforme Tabela 6.7.

Como observação, analisemos duas soluções encontradas para o conjunto de esferas em Dados1, uma pelo BONMIN e outra pelo XPRESS-SLP.

	BONMIN	XPRESS-SLP
z^*	328	328
$ S $	13	13
t	9	0
cov	70.22	63.71
miscov	5.44	8.83
overlap	3.46	5.27

Tabela 6.11: Comparação entre soluções.

Conforme Tabela 6.11, apesar de o XPRESS-SLP ter encontrado a solução em tempo muito menor, a solução do BONMIN se adequa melhor à aplicação, ou seja, ao tratamento por *Gamma Knife*. O BONMIN posicionou as mesmas 13 esferas de forma que elas recobrissem melhor o volume T , apresentando 70.22% de recobrimento contra 63.71% encontrado pelo XPRESS-SLP. Além disso, o primeiro algoritmo também posicionou as esferas de forma a ocupar menos a parte externa à T e de forma a haver menos interseções entre elas. Essas observações correspondem a valores menores para os parâmetros *miscov* e *overlap*.

No Capítulo 8, construiremos uma aproximação do nosso modelo, através de técnicas de discretização das variáveis, a fim de eliminar a não-linearidade e, principalmente, a não-convexidade do modelo, inerente do problema. No Capítulo 7, uma heurística será apresentada, com o intuito de obtenção de melhores soluções em tempos aceitáveis.

Capítulo 7

Heurística Proposta

7.1 A Heurística

Conforme mostrado no Capítulo 6, vimos que os resolvedores testados não conseguem boas soluções para o modelo ou não o fazem em tempo razoável. Desenvolvemos, então, uma heurística, a partir da necessidade de obtenção de soluções que recobrissem o sólido com um bom percentual de recobrimento em um tempo suficientemente curto. O procedimento será descrito a seguir.

Consideraremos a seguinte hipótese: esferas de diâmetros maiores são mais interessantes na solução, pois para um dado volume a ser recoberto, é necessário uma quantidade menor de esferas. Inicialmente, será disponibilizada apenas uma esfera para realizar o recobrimento. Tomaremos a esfera de maior raio possível. Seja esse raio r_1 . Assim, nosso conjunto inicial de esferas disponíveis será:

$$S = \{S_{r_1}\},$$

onde S_{r_1} representa uma esfera de raio r_1 . Tentaremos encontrar um posicionamento para esse conjunto de esferas, assumindo que todas as esferas de S estarão presentes na solução. Portanto, de acordo com o modelo proposto (5.5)-(5.8), teremos todas as variáveis binárias y iguais a 1. Substituindo no modelo, temos:

$$\max \quad 1 \tag{7.1}$$

$$\text{s. a } \|x^i - x^j\|^2 \geq (r_i + r_j - \alpha_{ij})^2, \quad \forall 1 \leq i < j \leq n \tag{7.2}$$

$$x \in T^n \tag{7.3}$$

É um problema não-linear e não-convexo, onde desejamos apenas encontrar uma solução viável. Portanto, a função a ser maximizada ou minimizada é irrelevante. O objetivo ao usar essa estratégia é conseguir um bom ponto inicial no momento em

que o conjunto S for composto de um número grande de esferas.

Dado um determinado conjunto S de esferas, pode ou não haver solução para o modelo (7.1)-(7.3). Seja $R = \{r_1, \dots, r_m\}$ os possíveis raios, ordenados do maior para o menor, e assumamos que tenhamos $S = \{S_{r_1}, S_{r_1}\}$. Temos duas situações:

- há solução: será adicionada ao conjunto S uma esfera de mesmo raio que a esfera anterior, obtendo

$$S = \{S_{r_1}, S_{r_1}, S_{r_1}\}.$$

A solução encontrada servirá como ponto inicial para uma nova otimização com esse novo conjunto. O ponto inicial da nova esfera será gerado de forma aleatória.

- não há solução: a última esfera previamente adicionada, responsável pela inviabilidade, será substituída por uma de raio menor. Teremos:

$$S = \{S_{r_1}, S_{r_2}\}.$$

Como ponto inicial, será utilizada a mais recente solução encontrada. Novamente, para a nova esfera, o ponto inicial será gerado de forma aleatória.

O algoritmo termina no momento em que uma solução não é encontrada e não há um valor menor de raio para substituir a última esfera adicionada.

7.2 Resultados no IPOPT

O IPOPT (*I*nterior *P*oint *O*ptimizer) [31] é um algoritmo para resolver problemas de otimização não-linear de grande escala. Ele foi projetado para encontrar soluções de problemas matemáticos da forma:

$$\min f(x) \tag{7.4}$$

$$\text{s. a } g_L \leq g(x) \leq g_U \tag{7.5}$$

$$x_L \leq x \leq x_U \tag{7.6}$$

onde $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ é a função objetivo e $g(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ são as restrições. Os vetores g_L e g_U denotam os limites inferiores e superiores das restrições, respectivamente. Os vetores x_L e x_U são os limites das variáveis x . As funções f e g podem ser não-lineares e não-convexas, mas devem ser contínuas e duas vezes diferenciáveis. Restrições de igualdade também são permitidas, bastando igualar as componentes correspondentes de g_L e g_U .

Utilizamos, para o teste da heurística com o IPOPT, o mesmo sólido descrito no Capítulo 6, um paralelepípedo de dimensões 14mm, 12mm e 10mm. Mantivemos, para a região de segurança, um valor $\varepsilon = 1$. Ilustrações da solução obtida podem ser vistas na Figura 7.1. Informações sobre a solução encontrada são apresentadas na Tabela 7.1.

IPOPT	
z^*	512
$ S $	29
t	15
cov	91.41
miscov	10.53
overlap	16.46

Tabela 7.1: Solução da heurística usando IPOPT.

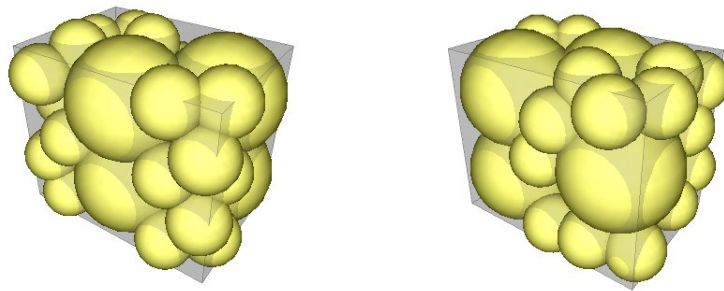


Figura 7.1: Imagens da solução da heurística usando IPOPT.

A solução da heurística foi a melhor solução encontrada, dentre todos os resolvedores e os métodos de discretização testados. Porém, da forma como foi apresentada, utilizando diretamente um resolvidor para problemas não-lineares, sua utilização só é possível quando as restrições 5.8 puderem ser escritas matematicamente na forma de equações ou inequações, como nos exemplos mostrados na Seção 5.2. Portanto, devido à natureza dos dados provenientes de tomografias computadorizadas e ressonâncias magnéticas, ao trabalharmos com o modelo visando a aplicação ao tratamento por *Gamma Knife*, o sólido T a ser recoberto é descrito por um conjunto finito de pontos. Por essa razão, no Capítulo 9, estudaremos uma reformulação do problema de recobrimento para tratar o caso em que o volume T é um conjunto discreto.

X	Y	Z	Raio
-0.399870	-0.015175	-0.196135	0.400000
0.399853	-0.299973	-0.199086	0.400000
-0.049273	-0.299981	0.199417	0.400000
0.399858	0.299973	-0.189098	0.400000
-0.057517	0.299980	0.199773	0.400000
0.599678	0.498885	0.224425	0.200000
0.056542	0.003356	-0.399848	0.200000
0.587800	-0.231268	0.390360	0.200000
-0.599495	0.147517	0.232979	0.200000
-0.599635	-0.472158	-0.312377	0.200000
-0.599790	0.441830	-0.149292	0.200000
-0.511155	0.499732	0.131630	0.200000
-0.464380	-0.499252	0.398061	0.200000
-0.598657	-0.395806	0.145826	0.200000
0.587982	0.234710	0.390913	0.200000
-0.419343	-0.499717	-0.072611	0.200000
0.363615	-0.499263	0.399710	0.200000
-0.599276	0.397513	0.399805	0.200000
-0.186864	0.499810	-0.240188	0.200000
-0.400785	-0.004638	0.399536	0.200000
0.592656	-0.001533	0.175216	0.200000
-0.013217	-0.499274	-0.399127	0.200000
-0.599383	-0.230288	0.397965	0.200000
-0.311948	-0.467262	-0.398821	0.200000
0.355718	0.499122	0.399636	0.200000
-0.053606	0.283182	-0.399783	0.200000
0.346715	0.007065	0.388155	0.200000
-0.434455	0.441655	-0.399848	0.200000
0.599734	-0.499094	0.214175	0.200000

Tabela 7.2: Resultado da heurística usando IPOPT.

Capítulo 8

Discretização

O modelo proposto é um modelo de programação matemática não-linear inteira mista (MINLP) cuja relaxação contínua é não-convexa, devido às restrições (5.6). Devido à dificuldade de obtenção de boas soluções por parte dos resolvidores atuais para essa classe de problemas, conforme apresentado no Capítulo 6, utilizaremos uma técnica de discretização de variáveis contínuas a fim de trabalhar com um modelo linear. A vantagem da programação linear é ser uma área bastante estudada e existirem atualmente algoritmos bastante eficientes para a resolução de seus problemas.

8.1 Primeira abordagem

Para exemplificar o modo como trataremos a discretização das variáveis, suponha que o modelo possua uma variável x real pertencente ao intervalo $[1, 4]$, ou seja,

$$1 \leq x \leq 4.$$

Suponha, ainda, que queremos discretizar esse intervalo e fazer com que a variável só assuma valores 1.5, 3 ou 4. Essa é uma aproximação muito ruim, apenas com a intenção de facilitar o entendimento. Uma boa aproximação é obtida com um número suficientemente grande de pontos na discretização dos intervalos, mas isso aumenta consideravelmente a quantidade de variáveis, como veremos adiante, e, por consequência, o custo computacional na resolução do problema. Adicionaremos uma variável binária λ ao modelo para cada ponto da discretização. Dentre elas, a que assumir valor 1 terá seu ponto associado como o valor escolhido para x . Dessa forma, temos:

$$x = 1.5\lambda_1 + 3\lambda_2 + 4\lambda_3 .$$

Observe que apenas essa equação não é suficiente para termos unicidade de representação. Por exemplo, para termos x assumindo valor 4, de acordo com a idéia apresentada, basta termos $\lambda_3 = 1$, enquanto as variáveis binárias restantes na equação assumem valor 0. Porém, também conseguimos $x = 4$ com $\lambda_1 = \lambda_2 = 1$ e $\lambda_3 = 0$. Ainda, poderíamos ter valores indesejados, como quando λ_2 e λ_3 assumem valor 1, por exemplo. O resultado seria $x = 7$, valor não admissível. Essas situações não ocorrerão se adicionarmos a seguinte equação ao modelo:

$$\lambda_1 + \lambda_2 + \lambda_3 = 1.$$

Ou seja, apenas uma das variáveis binárias poderá assumir valor 1. Definiremos, agora, formalmente.

Seja x_k^i a k -ésima coordenada do centro da esfera i . Seja $[a_k^i, b_k^i]$ o intervalo de variação da variável x_k^i , com a_k^i e b_k^i conhecidos, ou seja,

$$a_k^i \leq x_k^i \leq b_k^i.$$

Podemos discretizar cada um dos intervalos de definição das variáveis com a quantidade de pontos desejada. Através do uso de variáveis binárias λ , escrevemos:

$$x_k^i = w_{k,1}^i \lambda_{k,1}^i + \dots + w_{k,L_k^i}^i \lambda_{k,L_k^i}^i \quad (8.1)$$

e

$$\sum_{j=1}^{L_k^i} \lambda_{k,j}^i = 1, \quad (8.2)$$

onde

- L_k^i é a quantidade de pontos usados na discretização do intervalo $[a_k^i, b_k^i]$;
- $w_{k,j}^i$ são escolhidos como possíveis valores para a variável;
- $a_k^i \leq w_{k,1}^i < \dots < w_{k,L_k^i}^i \leq b_k^i$;
- $\lambda_{k,j}^i \in \{0, 1\}$, $\forall j \in \{1, \dots, L_k^i\}$.

Observe que essa idéia aplica-se caso todas as variáveis do modelo possuam limites superiores e inferiores conhecidos. O resultado será um reticulado, não necessariamente regular e podendo ser diferente para cada esfera. Uma outra abordagem, que também utiliza técnicas de discretização, será discutida na próxima seção, que pode ser usada quando a apresentada não é adequada.

8.2 Segunda abordagem

Caso os limites das variáveis não sejam conhecidos ou caso o reticulado não seja adequado, podemos utilizar uma outra abordagem. Daremos, aqui, outro exemplo, dentro do contexto do problema de recobrimento. Ao invés de discretizarmos o domínio das variáveis, discretizaremos agora o conjunto T . Assim, determinaremos a melhor localização das esferas dentre um conjunto finito V de pontos, com $V \subset T$.

Como exemplo, seja $V = \{v_1, v_2, v_3, v_4\}$, com

$$v_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad v_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad v_3 = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}, \quad v_4 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

Criaremos uma variável de decisão λ para cada ponto v , que assumirá valor 1 caso a esfera esteja centrada no ponto correspondente. Faremos

$$x = \lambda_1 v_1 + \lambda_2 v_2 + \lambda_3 v_3 + \lambda_4 v_4$$

$$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1.$$

Assim, em cada um das coordenadas, temos:

$$x_1 = \lambda_2 + \lambda_4$$

$$x_2 = 2\lambda_3 + \lambda_4$$

$$x_3 = \lambda_4.$$

De forma geral, seja $V = \{v_1, \dots, v_n\}$ o conjunto dos pontos que formam a discretização de T . Temos, para uma dada esfera i :

$$x_k^i = \lambda_1^i v_{1,k} + \dots + \lambda_n^i v_{n,k} \tag{8.3}$$

e

$$\sum_{p=1}^n \lambda_p^i = 1, \tag{8.4}$$

onde $v_{i,k}$ é a k -ésima coordenada do ponto v_i .

Observe que essa abordagem cria uma quantidade consideravelmente maior de variáveis binárias em comparação com a abordagem apresentada na seção 8.1.

8.3 Incorporando ao modelo

Nas seções anteriores, vimos duas formas de aproximar por um número finito de pontos uma determinada variável contínua, com o objetivo de transformar o modelo proposto em um modelo aproximado linear. O modelo será modificado no lado esquerdo da restrição (5.6). Temos:

$$\|x^i - x^j\|^2 = \sum_{k=1}^3 (x_k^i - x_k^j)^2 = \quad (8.5)$$

$$= (x_k^i)^2 - 2x_k^i x_k^j + (x_k^j)^2 \quad . \quad (8.6)$$

Tanto as equações (8.1) quanto as equações (8.3) representam uma soma de variáveis binárias. Suponhamos, então, que x_k^i esteja sendo representado, de forma discreta, por:

$$x_k^i = w_{k,1}^i \lambda_{k,1}^i + \dots + w_{k,L_{ik}}^i \lambda_{k,L_{ik}}^i \quad . \quad (8.7)$$

O termo $(x_k^i)^2$, com o uso das equações (8.2) ou (8.4), pode facilmente ser escrito como:

$$(x_k^i)^2 = (w_{k,1}^i)^2 \lambda_{k,1}^i + \dots + (w_{k,L_{ik}}^i)^2 \lambda_{k,L_{ik}}^i \quad . \quad (8.8)$$

Para o outro termo não-linear, $x_k^i x_k^j$, ao substituirmos pelas expressões da discretização, obtemos:

$$x_k^i x_k^j = \sum_{p=1}^{L_{ik}} \sum_{q=1}^{L_{jk}} w_{k,p}^i w_{k,q}^j \lambda_{k,p}^i \lambda_{k,q}^j \quad .$$

Essa substituição mantém a não-linearidade, apresentando ainda produtos de variáveis binárias. Aplicamos, então, uma técnica de linearização bastante conhecida na literatura. Fazemos

$$\delta_{kpq}^{ij} = \lambda_{k,p}^i \lambda_{k,q}^j \quad ,$$

com

- $\delta_{kpq}^{ij} \geq 0$;
- $\delta_{kpq}^{ij} \leq \lambda_{k,p}^i$;
- $\delta_{kpq}^{ij} \leq \lambda_{k,q}^j$; e
- $\delta_{kpq}^{ij} \geq \lambda_{k,p}^i + \lambda_{k,q}^j - 1$.

8.4 Resultados

Os resultados mostrados nesta seção são provenientes de testes utilizando técnicas de discretização do domínio de cada uma das variáveis ou técnicas de discretização do volume T a ser recoberto, conforme apresentadas nas seções 8.1 e 8.2, respectivamente.

Utilizaremos uma grade regular, definida por um parâmetro Δ . Para melhor entendimento, seja $0 \leq x \leq 2$ ou o intervalo $[0,2]$. Para um valor de, por exemplo, $\Delta = 0.5$, a variável x só poderia assumir valores no conjunto $\{0, 0.5, 1, 1.5, 2\}$, que corresponderia também à discretização do intervalo dado.

Formalmente, seja

$$lb \leq x \leq ub \quad \text{ou} \quad [lb, ub]$$

e

$$\text{nPontos} = \left\lfloor \frac{ub - lb}{\Delta} \right\rfloor + 1. \quad (8.9)$$

Temos:

$$x = \sum_{p=1}^{\text{nPontos}} [lb + (p-1)\Delta] \lambda_p \quad (8.10)$$

e

$$\sum_{p=1}^{\text{nPontos}} \lambda_p = 1. \quad (8.11)$$

Os resultados encontrados para $\Delta = 3$ e $\Delta = 1$ podem ser vistos na Tabela 8.1. Utilizamos o XPRESS para resolver ambas as abordagens apresentadas nas seções 8.1 e 8.2. O algoritmo foi interrompido após 3600 segundos.

	Abordagem1		Abordagem2	
	$\Delta = 3$	$\Delta = 1$	$\Delta = 3$	$\Delta = 1$
z^*	328	368	328	-
$ S $	13	11	13	-
t	14	2700	135	-
cov	64.29	72.45	64.28	-
miscov	10.03	8.08	9.61	-
overlap	6.8	6.82	7.08	-

Tabela 8.1: Resultados do XPRESS para os modelos discretizados.

Na Tabela 8.1, é apresentado o valor da função objetivo z^* em cada solução e a quantidade de esferas $|S|$ utilizada. O tempo t em que a melhor solução foi encontrada é dado em segundos.

Como a abordagem que utiliza a discretização do volume T a ser recoberto gera uma quantidade maior de pontos que a discretização do domínio das variáveis x , teremos também uma quantidade maior de variáveis binárias nesse modelo. Como consequência, os resultados foram melhores na primeira abordagem. Para $\Delta = 3$, ambas encontraram a mesma solução, porém em tempos bastante diferentes. Para $\Delta = 1$, a segunda abordagem utilizou toda a memória da máquina e, portanto, não conseguimos terminar a execução do algoritmo.

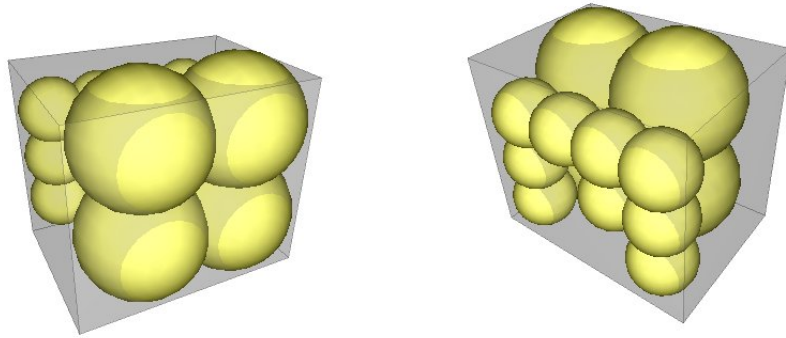


Figura 8.1: Solução encontrada pelo XPRESS para $\Delta = 3$.

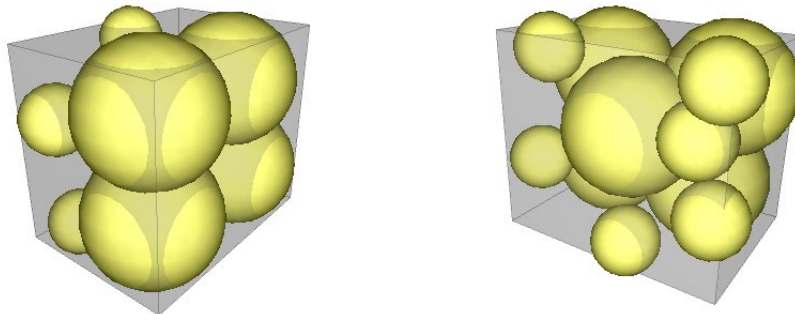


Figura 8.2: Solução encontrada pelo XPRESS para $\Delta = 1$.

Nas Tabelas 8.2 e 8.3, estão representados os resultados do XPRESS com o modelo discretizado para os valores $\Delta = 3$ e $\Delta = 1$, respectivamente. Os valores estão todos divididos por 10, estando o centro geométrico do paralelepípedo na origem.

X	Y	Z	Raio
-0.4	-0.3	-0.2	0.4
-0.4	0.3	-0.2	0.4
0.2	-0.3	-0.2	0.4
0.2	0.3	-0.2	0.4
-0.4	0.3	0.4	0.2
-0.4	0	0.4	0.2
0.5	-0.3	0.4	0.2
-0.1	0.3	0.4	0.2
0.5	0	0.4	0.2
-0.4	-0.3	0.4	0.2
0.5	0.3	0.4	0.2
0.2	0.3	0.4	0.2
0.2	0	0.4	0.2

Tabela 8.2: Resultado do XPRESS para $\Delta = 3$.

X	Y	Z	Raio
0.3	0.3	0.2	0.4
-0.4	-0.3	0.2	0.4
0.4	-0.3	0.2	0.4
0	-0.1	-0.2	0.4
-0.4	0.3	0.2	0.4
-0.6	0.2	-0.3	0.2
-0.5	-0.5	-0.3	0.2
0.1	0.5	-0.4	0.2
0.5	-0.2	-0.4	0.2
0.6	0.2	-0.4	0.2
0.6	-0.5	-0.4	0.2

Tabela 8.3: Resultado do XPRESS para $\Delta = 1$.

Capítulo 9

Abordagem por grafo

9.1 Introdução

Quando o volume T a ser recoberto tem um formato irregular, de forma que a restrição 5.7 não possa ser explicitada, é necessário o uso de técnicas de discretização. Na prática, em particular no problema do tratamento conhecido como *Gamma Knife*, a região do tumor é obtida através de imagens de ressonância magnética ou de tomografias computadorizadas. Essas imagens são representadas por um conjunto finito de pontos e justifica um estudo nessa direção. Porém, mesmo quando a região permite que a restrição 5.7 seja escrita matematicamente, pode-se tirar proveito em discretizar T , de forma que o problema deixe de ser um problema de programação não-linear inteira mista (MINLP) não-convexo e torne-se um problema linear com variáveis binárias. Apresentaremos neste capítulo, uma outra abordagem para o problema de recobrimento, com a utilização de um grafo.

9.2 Grafos e Cliques

Um grafo $G(V, A)$ é uma estrutura formada por dois conjuntos: um conjunto V , cujos elementos são denominados vértices, e um conjunto A formado por pares de elementos de V , chamados arestas. Um exemplo pode ser visto na Figura 9.1.

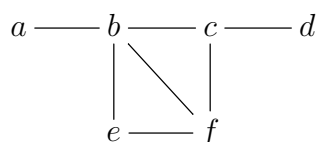


Figura 9.1: Exemplo de grafo, onde $V = \{a,b,c,d,e,f\}$ e $A = \{(a,b), (b,c), (b,e), (b,f), (c,d), (c,f), (e,f)\}$.

Sejam dois vértices v e w . Dizemos que v e w são *adjacentes* se existe uma aresta em A que os conecta. Ou seja,

$$v \text{ é adjacente a } w \iff (v, w) \in A.$$

Neste trabalho, apresentamos apenas arestas não-direcionadas. Portanto, consideraremos $(v, w) = (w, v)$.

O complemento de um grafo $G(V, A)$, denotado por \bar{G} , é um grafo contendo os mesmos vértices e tal que seus vértices são adjacentes se e somente se não o são em G . De outra forma, seja K o conjunto de todos os pares não ordenados de elementos de V . Temos que o grafo complementar de G é o grafo $\bar{G}(V, K \setminus A)$. Um exemplo de grafos complementares pode ser visto na Figura 9.2.

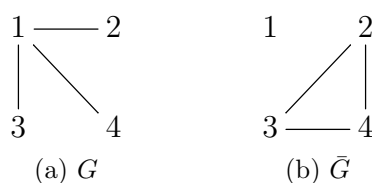


Figura 9.2: Exemplo de grafos complementares.

Uma *clique* em um grafo é qualquer conjunto de vértices dois a dois adjacentes. Na Figura 9.1, temos a clique $\{b, c, f\}$, por exemplo.

Um conjunto de vértices de um grafo é *estável* se seus elementos são dois a dois não-adjacentes, ou seja, se nenhuma aresta tem ambas as pontas no conjunto. No exemplo da Figura 9.1, temos que o conjunto $\{a, d, e\}$ é estável.

Uma clique ou um conjunto estável é dito *maximal* se não faz parte, respectivamente, de uma clique ou de um conjunto estável maior, ou seja, se não é subconjunto de uma outra clique ou de um outro conjunto estável. Matematicamente, dado um grafo $G(V, A)$ e um conjunto C de todas as cliques ou de todos os conjuntos estáveis de G , temos:

$$X \in C \text{ é maximal} \iff \nexists Y \in C \mid X \subset Y.$$

Há uma relação óbvia entre cliques e conjuntos estáveis. Um conjunto C de vértices é uma clique do grafo G se e somente se C é um conjunto estável no grafo complementar \bar{G} .

A cada vértice do grafo pode-se atribuir um determinado valor, chamado de *peso*. Assim, podemos definir a *clique de peso máximo* como sendo a clique cuja soma dos pesos de seus vértices é a maior dentre todas as cliques. Essa definição é equivalente para conjuntos estáveis de peso máximo.

Seja w_i o peso do vértice $i \in V$. Uma formulação de programação matemática para o problema de encontrar uma clique de peso máximo em um grafo $G(V, A)$ pode ser escrita como segue:

$$\max \sum_{i=1}^{|V|} w_i y_i \quad (9.1)$$

$$\text{s. a } y_i + y_j \leq 1, \quad \forall (i, j) \notin A \quad (9.2)$$

$$y_i \in \{0, 1\}, \quad \forall i \in V. \quad (9.3)$$

Em (9.1)-(9.3), a variável y_i , associada ao vértice $i \in V$, assumirá valor 1 se o vértice i fizer parte da clique de peso máximo. De acordo com a restrição 9.2, se não há uma aresta entre dois vértices i e j , então esses vértices não podem pertencer à mesma clique, por definição. Logo, não podem ambos estar presentes na solução.

9.3 Nova abordagem

Abordaremos, nesta seção, uma outra forma de resolver o problema quando o conjunto T a ser recoberto é um conjunto finito de pontos, provenientes de imagens de tomografias computadorizadas ou ressonâncias magnéticas ou mesmo resultado de técnicas de discretização.

Seja R um conjunto finito de raios das esferas e $T \subset \mathbb{R}^n$ um conjunto finito a ser recoberto. Construiremos um grafo onde cada vértice está associado a uma n -upla $(p, r) \in T \times R$. Como exemplo, suponha $R = \{r_1, r_2\}$ e $T = \{p_1, p_2, p_3\}$. Teremos os vértices representados na Figura 9.3.

$$\begin{array}{ccc} (p_1, r_1) & (p_2, r_1) & (p_3, r_1) \\ \\ (p_1, r_2) & (p_2, r_2) & (p_3, r_2) \end{array}$$

Figura 9.3: Exemplo de construção de vértices.

Se um determinado vértice (p_i, r_k) faz parte da solução, então teremos um esfera de raio r_k centrada no ponto p_i .

Dessa forma, teremos que a quantidade de vértices será $|V| = |T| \cdot |R|$. Na verdade, alguns pontos de T não serão utilizados para determinados valores de R , quando, juntos, desrespeitarem a restrição (5.9), que define a região de segurança. Portanto, a quantidade de vértices pode ser um pouco menor.

Para decidir a existência das arestas, tomemos dois vértices (p_i, r_k) e (p_j, r_s) . Eles serão adjacentes se e somente se respeitarem a restrição (5.6), que impõe o máximo de interseção permitida entre cada par de esferas. De outra forma, temos que:

$$((p_i, r_k), (p_j, r_s)) \text{ é aresta} \iff \|p_i - p_j\| \geq (r_k + r_s - \alpha_{ks}).$$

Construímos, então, o grafo $G(V, A)$ tal que:

$$V = \{ (p, r) \in (T \times R) \mid B(p, r) \subseteq RS \} \quad (9.4)$$

e

$$A = \{ ((p, r), (q, s)) \in V^2 \mid \|p - q\| \geq (r + s - \alpha_{rs}) \}. \quad (9.5)$$

Para os pesos dos vértices, assumiremos que todas as esferas de mesmo raio têm a mesma importância na solução. Assim, seja c_r o custo de uma esfera de raio r , como em (5.5). Atribuiremos, a todo vértice $v = (p, r)$, o peso $w_v = c_r$.

Podemos encontrar a solução de (5.5)-(5.8) resolvendo o problema de encontrar a clique de peso máximo no grafo $G(V, A)$, criado como descrito.

9.4 Algoritmo de *Branch and Cut*

A restrição (9.2) representa todos os conjuntos estáveis de tamanho 2. Conseguiríamos um modelo mais forte ao substituir tais restrições por conjuntos estáveis de tamanhos maiores. Essa estratégia é inviável, visto que encontrar um conjunto estável é tão difícil computacionalmente quanto encontrar uma clique. Assim, o algoritmo de *Branch and Cut* utilizado tem por objetivo não usar as restrições (9.2) e, a partir da solução encontrada em cada nó da árvore de Branch and Bound, adicionar, na forma de cortes, conjuntos estáveis que contenham peso maior que 1, através de algum algoritmo de baixo tempo de execução, como, por exemplo, heurísticas [32]. Teremos os seguintes cortes:

$$\sum_{e \in E} y_e \leq 1, \quad (9.6)$$

onde E é um conjunto estável de $G(V, A)$. Além disso, para melhorar a performance, alguns conjuntos estáveis são previamente calculados e adicionados ao nó raiz da árvore de *Branch and Bound* na forma de restrições. Esses conjuntos são calculados a partir da seguinte observação:

$$\{ q \in \bar{T} \mid q \in B(p, \alpha_{rr}/2), w_q = w_p = r \} \text{ é estável.} \quad (9.7)$$

Em (9.7), \bar{T} é o conjunto dos pontos da discretização de T , o volume a ser recoberto. Como, inicialmente, atribuímos a todos os pares de esferas com raios r e s o mesmo valor α_{rs} , então todas as esferas de mesmo raio r terão o mesmo valor α_{rr} , duas a duas. Assim, ao fixarmos um ponto $p \in \bar{T}$ com raio r e tomarmos todos os pontos $q \in \bar{T}$, também com raio r e tais que distam de p menos que a metade de α_{rr} , poderemos afirmar que a máxima distância entre quaisquer dois pontos tomados será menor que α_{rr} . Logo, não pode haver aresta entre quaisquer desses pontos, de acordo com restrição (5.6). Matematicamente, tomaremos todos os pontos $q \in \bar{T}$ tais que

$$\|q - p\| < \frac{\alpha_{rr}}{2}. \quad (9.8)$$

Dessa forma, sejam quaisquer dois pontos q_1 e q_2 satisfazendo (9.8). Pela desigualdade triangular, temos:

$$\|q_1 - q_2\| \leq \|q_1 - p\| + \|q_2 - p\| < \frac{\alpha_{rr}}{2} + \frac{\alpha_{rr}}{2} = \alpha_{rr}. \quad (9.9)$$

Assim, podemos afirmar que o conjunto descrito em (9.7) é um conjunto estável, não necessariamente maximal. A partir desse conjunto, com a idéia de torná-lo maximal, pode-se rodar um algoritmo para encontrar o estável máximo apenas no subgrafo que contém os vértices que não estão ligados a nenhum dos pontos do conjunto estável inicial. A dimensão é bastante menor, permitindo a utilização de um algoritmo exato.

Como exemplo, seja o grafo da Figura 9.4. Tome a clique não maximal $\{a, b, e\}$. Um aumento no tamanho dessa clique pode ser obtido a partir do subgrafo formado por todos os vértices adjacentes em comum. Basta encontrar a clique máxima no subgrafo formado por $\{c, d\}$. Para conjuntos estáveis, a idéia é semelhante, bastando resolver o problema no grafo complementar.

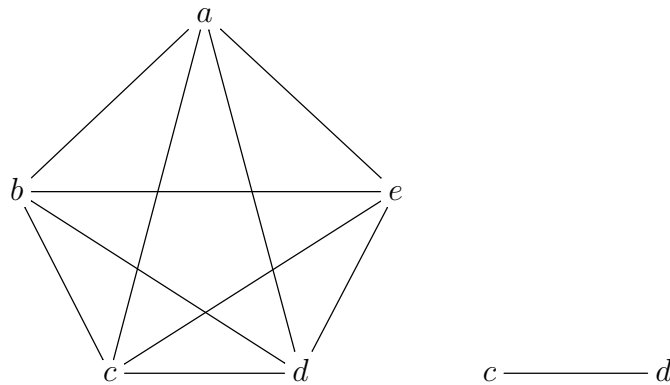


Figura 9.4: Grafo à esquerda e subgrafo à direita.

9.5 Resultados

Para encontrar uma clique de peso máximo em um grafo $G(V, A)$, basta examinar todos os subconjuntos de V . Computacionalmente, esse algoritmo consome tempo proporcional a $2^{|V|}$. Como 2^n cresce muito rápido com o aumento de n , esse algoritmo é insatisfatório na prática.

Os resultados que serão apresentados utilizam o grafo construído conforme descrito na Seção 9.3, para um paralelepípedo de dimensões 14 mm x 12 mm x 10 mm, assim como na Figura 6.2. É utilizado um valor $\varepsilon = 1$ para o parâmetro da região de segurança.

Aplicamos o *Cliquer* [33], um algoritmo exato de *Branch and Bound* para resolver o problema de encontrar a clique de peso máximo em um dado grafo. Aplicamos, também, o XPRESS para resolver o modelo (9.1)-(9.3). Por fim, utilizamos o algoritmo de *Branch and Cut* proposto na Seção 9.4.

Para a discretização do volume a ser recoberto, utilizamos uma grade regular, definida por um parâmetro Δ , como explicado na Seção 8.4. Para $\Delta = 3$, o *Cliquer*, o XPRESS e o algoritmo de *Branch and Cut* encontraram a mesma solução. Características dessa solução, em comparação com o resultado para o modelo discretizado do Capítulo 8, são mostradas na Tabela 9.1. Suas imagens estão mostradas na Figura 9.5.

	Grafo	Abordagem1
z^*	480	328
$ S $	60	13
t	0	14*
cov	78.64	64.29
miscov	8.01	10.03
overlap	21.78	6.8

Tabela 9.1: Soluções para $\Delta = 3$

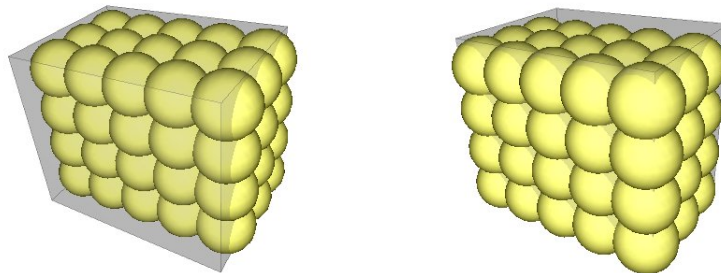


Figura 9.5: Imagens da solução para $\Delta = 3$.

X	Y	Z	Raio	X	Y	Z	Raio
-0.6	-0.5	-0.4	0.2	0	0.1	-0.4	0.2
-0.6	-0.5	-0.1	0.2	0	0.1	-0.1	0.2
-0.6	-0.5	0.2	0.2	0	0.1	0.2	0.2
-0.6	-0.2	-0.4	0.2	0	0.4	-0.4	0.2
-0.6	-0.2	-0.1	0.2	0	0.4	-0.1	0.2
-0.6	-0.2	0.2	0.2	0	0.4	0.2	0.2
-0.6	0.1	-0.4	0.2	0.3	-0.5	-0.4	0.2
-0.6	0.1	-0.1	0.2	0.3	-0.5	-0.1	0.2
-0.6	0.1	0.2	0.2	0.3	-0.5	0.2	0.2
-0.6	0.4	-0.4	0.2	0.3	-0.2	-0.4	0.2
-0.6	0.4	-0.1	0.2	0.3	-0.2	-0.1	0.2
-0.6	0.4	0.2	0.2	0.3	-0.2	0.2	0.2
-0.3	-0.5	-0.4	0.2	0.3	0.1	-0.4	0.2
-0.3	-0.5	-0.1	0.2	0.3	0.1	-0.1	0.2
-0.3	-0.5	0.2	0.2	0.3	0.1	0.2	0.2
-0.3	-0.2	-0.4	0.2	0.3	0.4	-0.4	0.2
-0.3	-0.2	-0.1	0.2	0.3	0.4	-0.1	0.2
-0.3	-0.2	0.2	0.2	0.3	0.4	0.2	0.2
-0.3	0.1	-0.4	0.2	0.6	-0.5	-0.4	0.2
-0.3	0.1	-0.1	0.2	0.6	-0.5	-0.1	0.2
-0.3	0.1	0.2	0.2	0.6	-0.5	0.2	0.2
-0.3	0.4	-0.4	0.2	0.6	-0.2	-0.4	0.2
-0.3	0.4	-0.1	0.2	0.6	-0.2	-0.1	0.2
-0.3	0.4	0.2	0.2	0.6	-0.2	0.2	0.2
0	-0.5	-0.4	0.2	0.6	0.1	-0.4	0.2
0	-0.5	-0.1	0.2	0.6	0.1	-0.1	0.2
0	-0.5	0.2	0.2	0.6	0.1	0.2	0.2
0	-0.2	-0.4	0.2	0.6	0.4	-0.4	0.2
0	-0.2	-0.1	0.2	0.6	0.4	-0.1	0.2
0	-0.2	0.2	0.2	0.6	0.4	0.2	0.2

Tabela 9.2: Resultado para $\Delta = 3$.

O tempo de execução foi mínimo para a abordagem por grafos, enquanto que não foi possível resolver o problema com a discretização do modelo (Abordagem1) dentro do tempo máximo estipulado de 3600 segundos. O valor 14 representado na Tabela 9.1 é o tempo em que a solução foi encontrada.

Apesar de a solução da abordagem por grafos ter apresentado um melhor coeficiente de recobrimento, conseguindo 78.64% contra 64.29%, a mesma utiliza 60 esferas, um número alto quando comparado a apenas 13 esferas na outra solução. A razão é a utilização de apenas esferas de raios 2 mm no recobrimento, conforme Tabela 9.2, o que pode ser contornado ao modificarmos os parâmetros c_i , na função objetivo (5.5), atribuindo maior peso às esferas maiores.

Modificando o parâmetro Δ para um valor 1, a região viável torna-se maior, em quantidade de soluções. Portanto, melhores resultados foram encontrados para $\Delta = 1$, conforme Tabela 9.3. Os três algoritmos foram interrompidos após 3600 segundos de execução. Na tabela, t representa o tempo em que a melhor solução foi encontrada. Para o *Cliquer*, informações não conhecidas são mostradas com um sinal de interrogação. Imagens das soluções encontradas estão contidas nas Figuras 9.6 e 9.7.

	<i>Cliquer</i>	Abordagem1	XPRESS	B&C
z^*	224	368	544	592
$ S $?	11	61	74
t	3400	2700	1957	747
cov	?	72.45	89.85	92.62
miscov	?	8.08	12.82	12.34
overlap	?	6.82	19.27	27.41

Tabela 9.3: Soluções para $\Delta = 1$

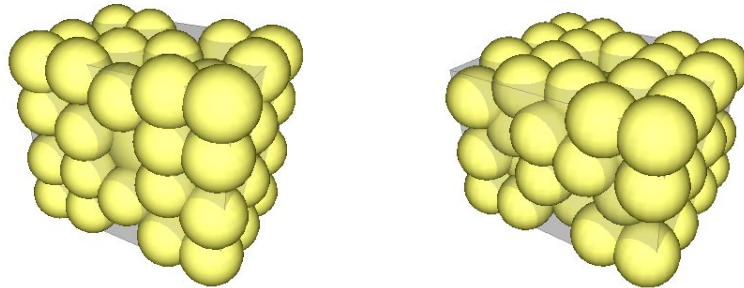


Figura 9.6: Imagens da solução encontrada pelo XPRESS para $\Delta = 1$.

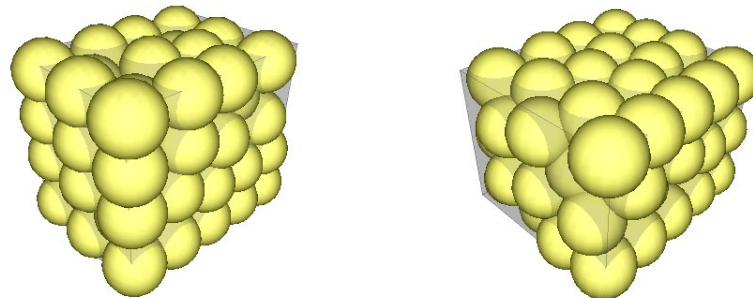


Figura 9.7: Imagens da solução encontrada pelo *Branch and Cut* para $\Delta = 1$.

X	Y	Z	Raio	X	Y	Z	Raio
0	0	0	0.4	0	0.3	-0.4	0.2
-0.6	-0.5	-0.4	0.2	0	0.4	0.4	0.2
-0.6	-0.5	-0.1	0.2	0.1	-0.5	-0.2	0.2
-0.6	-0.5	0.4	0.2	0.1	-0.5	0.1	0.2
-0.6	-0.2	-0.4	0.2	0.1	0.5	0	0.2
-0.6	-0.2	0	0.2	0.3	-0.5	0.4	0.2
-0.6	-0.2	0.4	0.2	0.3	-0.4	-0.4	0.2
-0.6	0.2	-0.4	0.2	0.3	-0.1	-0.4	0.2
-0.6	0.2	0	0.2	0.3	-0.1	0.4	0.2
-0.6	0.2	0.4	0.2	0.3	0.2	-0.4	0.2
-0.6	0.5	-0.4	0.2	0.3	0.2	0.4	0.2
-0.6	0.5	0.1	0.2	0.3	0.5	-0.4	0.2
-0.6	0.5	0.4	0.2	0.3	0.5	0.4	0.2
-0.5	0	-0.2	0.2	0.4	-0.4	-0.1	0.2
-0.5	0	0.2	0.2	0.4	-0.3	0.2	0.2
-0.4	-0.4	0.1	0.2	0.4	0.4	0.1	0.2
-0.4	-0.3	-0.2	0.2	0.5	0	-0.2	0.2
-0.4	0.3	0.2	0.2	0.5	0	0.2	0.2
-0.4	0.4	-0.1	0.2	0.6	-0.5	-0.3	0.2
-0.3	-0.5	-0.4	0.2	0.6	-0.5	0.1	0.2
-0.3	-0.5	0.4	0.2	0.6	-0.5	0.4	0.2
-0.3	-0.2	0.4	0.2	0.6	-0.2	-0.4	0.2
-0.3	-0.1	-0.4	0.2	0.6	-0.2	0	0.2
-0.3	0.1	0.4	0.2	0.6	-0.2	0.4	0.2
-0.3	0.2	-0.4	0.2	0.6	0.2	-0.4	0.2
-0.3	0.5	-0.4	0.2	0.6	0.2	0	0.2
-0.3	0.5	0.4	0.2	0.6	0.2	0.4	0.2
-0.2	-0.5	-0.1	0.2	0.6	0.5	-0.4	0.2
-0.2	0.5	0.1	0.2	0.6	0.5	-0.1	0.2
0	-0.3	-0.4	0.2	0.6	0.5	0.4	0.2
0	-0.3	0.4	0.2				

Tabela 9.4: Resultado do XPRESS para $\Delta = 1$.

X	Y	Z	Raio	X	Y	Z	Raio
-0.6	-0.5	-0.4	0.2	0	-0.2	-0.4	0.2
-0.6	-0.5	0	0.2	0	-0.2	0	0.2
-0.6	-0.5	0.4	0.2	0	-0.2	0.4	0.2
-0.6	-0.1	-0.4	0.2	0	0.3	-0.2	0.2
-0.6	-0.1	0	0.2	0	0.3	0.2	0.2
-0.6	-0.1	0.4	0.2	0.1	0.1	-0.4	0.2
-0.6	0.2	-0.4	0.2	0.1	0.1	0	0.2
-0.6	0.2	0	0.2	0.1	0.1	0.4	0.2
-0.6	0.2	0.4	0.2	0.1	0.5	-0.4	0.2
-0.6	0.5	-0.4	0.2	0.1	0.5	0.4	0.2
-0.6	0.5	0	0.2	0.2	-0.4	0.1	0.2
-0.6	0.5	0.4	0.2	0.2	-0.3	-0.2	0.2
-0.5	-0.3	-0.2	0.2	0.2	0.5	0.1	0.2
-0.5	-0.3	0.2	0.2	0.3	-0.5	-0.4	0.2
-0.4	0.1	-0.2	0.2	0.3	-0.5	0.4	0.2
-0.4	0.1	0.2	0.2	0.3	-0.2	0.4	0.2
-0.4	0.4	-0.2	0.2	0.3	-0.1	0.1	0.2
-0.4	0.4	0.2	0.2	0.3	0	-0.2	0.2
-0.3	-0.5	-0.4	0.2	0.3	0.3	0.3	0.2
-0.3	-0.5	0	0.2	0.3	0.4	-0.2	0.2
-0.3	-0.5	0.4	0.2	0.4	-0.5	-0.1	0.2
-0.3	-0.1	-0.4	0.2	0.4	-0.2	-0.4	0.2
-0.3	-0.1	0	0.2	0.4	0.2	-0.4	0.2
-0.3	-0.1	0.4	0.2	0.4	0.2	0	0.2
-0.2	-0.3	-0.2	0.2	0.5	-0.2	-0.1	0.2
-0.2	-0.3	0.2	0.2	0.5	0.1	0.4	0.2
-0.2	0.2	-0.4	0.2	0.5	0.5	-0.4	0.2
-0.2	0.2	0	0.2	0.5	0.5	0	0.2
-0.2	0.2	0.4	0.2	0.5	0.5	0.4	0.2
-0.2	0.5	-0.4	0.2	0.6	-0.5	-0.4	0.2
-0.2	0.5	0.4	0.2	0.6	-0.5	0.4	0.2
-0.1	0	-0.2	0.2	0.6	-0.4	0.1	0.2
-0.1	0	0.2	0.2	0.6	-0.2	0.4	0.2
-0.1	0.5	0	0.2	0.6	0	-0.3	0.2
0	-0.5	-0.4	0.2	0.6	0	0.1	0.2
0	-0.5	-0.1	0.2	0.6	0.3	-0.2	0.2
0	-0.5	0.4	0.2	0.6	0.3	0.2	0.2

Tabela 9.5: Resultado do *Branch and Cut* para $\Delta = 1$.

Novamente, o alto valor de 592 para a função objetivo, encontrado pelo algoritmo de *Branch and Cut*, corresponde à utilização de apenas esferas de raio 2 mm, conforme Tabela 9.5. Por consequência, observa-se uma enorme quantidade de esferas na solução. Em razão disso, aumentamos o peso dos vértices correspondentes às esferas de raio 4 mm e diminuimos o peso dos vértices correspondentes às esferas de raio 2 mm. Ao invés de 64 e 8, como anteriormente, utilizamos 20 e 1, respectivamente. Imagens da solução encontrada após essas modificações podem ser vistas na Figura 9.8. As características da solução estão descritas na Tabela 9.6, considerando os custos originais, em comparação com a solução encontrada pela heurística descrita no Capítulo 7, cuja principal característica é priorizar esferas de raios maiores.

	B&C	IPOPT
z^*	472	512
$ S $	24	29
t	6	15
cov	88.11	91.41
miscov	9.18	10.53
overlap	10.92	16.46

Tabela 9.6: Solução do *B&C* para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$, em comparação com IPOPT.

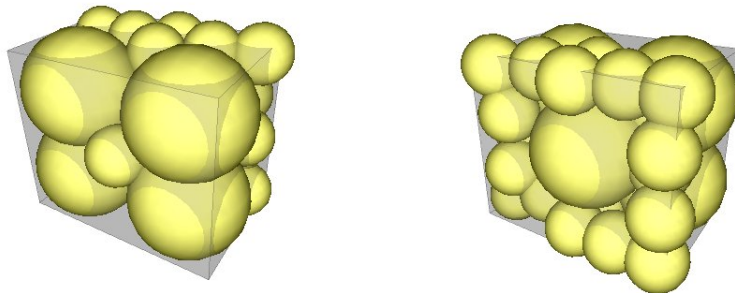


Figura 9.8: Imagens da solução encontrada pelo *Branch and Cut* para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$.

X	Y	Z	Raio
-0.4	-0.3	0.2	0.4
-0.4	0.3	0.2	0.4
0	0	-0.2	0.4
0.4	-0.3	0.2	0.4
0.4	0.3	0.2	0.4
-0.6	-0.5	-0.4	0.2
-0.6	-0.2	-0.4	0.2
-0.6	0.2	-0.4	0.2
-0.6	0.5	-0.3	0.2
-0.5	0	-0.2	0.2
-0.3	-0.5	-0.4	0.2
-0.3	0.5	-0.3	0.2
0	-0.5	-0.4	0.2
0	-0.5	-0.1	0.2
0	0	0.4	0.2
0	0.5	-0.4	0.2
0	0.5	-0.1	0.2
0.3	-0.4	-0.3	0.2
0.3	0.5	-0.3	0.2
0.5	-0.2	-0.4	0.2
0.5	0.2	-0.4	0.2
0.6	-0.5	-0.3	0.2
0.6	0	-0.2	0.2
0.6	0.5	-0.4	0.2

Tabela 9.7: Resultado do *Branch and Cut* para $\Delta = 1$, $c_4 = 20$ e $c_2 = 1$.

9.5.1 Comparação com resultados anteriores

Em LIBERTI *et al.* [1], os autores realizaram testes para recobrimentos de elipsóides. Utilizaram um modelo de programação matemática como descrito na Seção 2.2. Na Tabela 9.8, listamos as 4 instâncias empregadas, onde R_x , R_y e R_z são as dimensões do elipsóide, ε o parâmetro que define a região de segurança e Δ é o comprimento do lado de cada elemento cúbico da grade de discretização.

Nome	R_x	R_y	R_z	ε	Δ
bt1	12	8	6	4	5
bt2	12	8	6	4	3
bt3	12	8	6	4	2
bt4	10	10	10	6	3

Tabela 9.8: Instâncias utilizadas em LIBERTI *et al.* [1].

Utilizamos o algoritmo de *Branch and Cut*, como descrito na Seção 9.4, para resolver as mesmas instâncias do problema. Um quadro comparativo dos melhores resultados obtidos pode ser visto na Tabela 9.9. Nela, a coluna *cov* representa o percentual de recobrimento alcançado e $\#S$ é quantidade de esferas presentes na solução. O tempo é mostrado em segundos.

Nome	LIBERTI <i>et al.</i> [1]			<i>Branch and Cut</i>		
	Tempo	<i>cov</i>	$\#S$	Tempo	<i>cov</i>	$\#S$
bt1	1407	0.874	2	0	0.8912	2
bt2	51168*	0.984	5	0	0.9626	4
bt3	60058*	1	2	1	0.962	8
bt4	58068*	0.907	3	1	0.9812	6

Tabela 9.9: Comparação entre os resultados de [1] e o algoritmo de *Branch and Cut* proposto.

Na Tabela 9.9, os tempos marcados com * indicam o tempo em que a primeira solução foi encontrada. Os tempos apresentados para o algoritmo de *Branch and Cut* não consideram o cálculo referente às restrições que representam as cliques pré-adicionadas. O algoritmo de *Branch and Cut* apresentou soluções com melhor recobrimento nas instâncias bt1 e bt4, apesar de, na última, ter utilizado uma quantidade maior de esferas. O algoritmo utilizado em [1] recobriu melhor o elipsóide nas instâncias bt2 e bt3. Em todas as instâncias, o tempo de execução do algoritmo de *Branch and Cut* foi extremamente rápido quando comparado ao do artigo mencionado.

As instâncias bt1, bt2 e bt3 consistem do mesmo elipsóide com a mesma região região de segurança. Diferem apenas na quantidade de pontos utilizados na discretização. Utilizam $\Delta = 5, 3$ e 2 , respectivamente. Como nosso algoritmo tem apresentado pouco tempo de execução, testamos essa instância com $\Delta = 1$, como descrito na Tabela 9.10. A solução encontrada pelo algoritmo de *Branch and Cut* está representada na Tabela 9.11 e suas imagens podem ser vistas na Figura 9.9. Foi o melhor recobrimento encontrado para o elipsóide de 12 mm x 8 mm x 6 mm e ainda utiliza a menor quantidade de esferas.

Nome	R_x	R_y	R_z	ε	Δ
bt0	12	8	6	4	1

Tabela 9.10: Instância como em bt1, bt2 e bt3, com $\Delta = 1$.

Nome	<i>Branch and Cut</i>		
	Tempo	<i>cov</i>	$\#S$
bt0	12	0.9992	2

Tabela 9.11: Resultado para a instância com $\Delta = 1$.

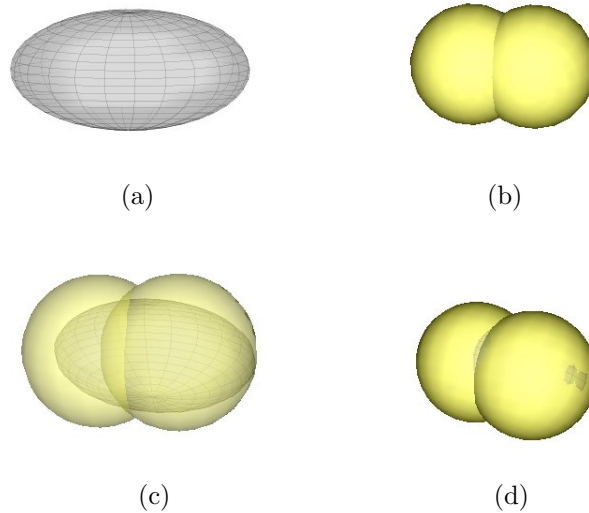


Figura 9.9: Imagens da solução encontrada pelo *Branch and Cut* para a instância bt0: (a) o elipsóide a ser recoberto; (b) as esferas no recobrimento; (c) efeito de transparência para melhor visualização; e (d) visualizando a parte não recoberta.

9.5.2 Caso Real

Através de dados cedidos pela Elekta [34], uma empresa com significativas inovações e soluções clínicas para o tratamento de câncer e distúrbios no cérebro, pudemos realizar testes mais realísticos. Na Figura 9.10, os pontos em azul representam a discretização do tumor e os pontos em vermelho correspondem a um órgão que não pode receber radiação.

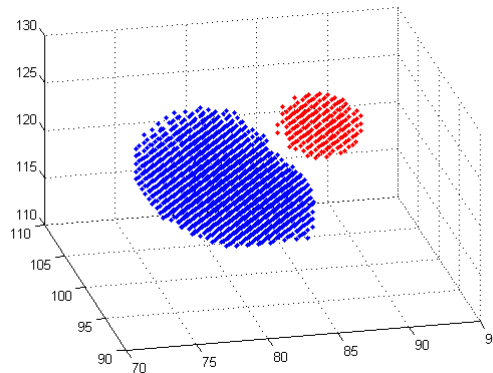


Figura 9.10: Dados reais cedidos pela empresa Elekta.

Os pontos pertencem a um reticulado cúbico com valor 0.8 entre os pontos. Realizamos três testes:

- (a) $\varepsilon = 1$, $c_4 = 64$ e $c_2 = 8$;
- (b) $\varepsilon = 1$, $c_4 = 20$ e $c_2 = 1$; e
- (c) sem região de segurança.

Em (a), mantivemos o coeficiente na função objetivo como sendo o cubo do raio da respectiva esfera. Para priorizar esferas de tamanhos maiores, modificamos esses valores em (b). Em (c), não utilizamos a região de segurança. Ou seja, a única restrição quanto à parte externa do tumor é as esferas não tocarem o órgão em risco.

As características das soluções estão representadas na Tabela 9.12. A efeito de comparação, os valores da função objetivo, representados por z^* , foram calculados utilizando o cubo dos raios. Em (a) o algoritmo foi interrompido após 3600 segundos, sendo 7 segundos o tempo em que a melhor solução foi encontrada.

Imagens de cada uma dessas soluções podem ser vistas nas Figuras 9.11, 9.12 e 9.13, geradas através do Octave [35].

	B&C		
	(a)	(b)	(c)
z^*	208	168	1474
$ S $	26	7	4
t	7*	1	183
cov	82	63	99
miscov	?	?	alto
overlap	13.8	4.13	20.25

Tabela 9.12: Soluções do *B&C* para o teste com dados reais.

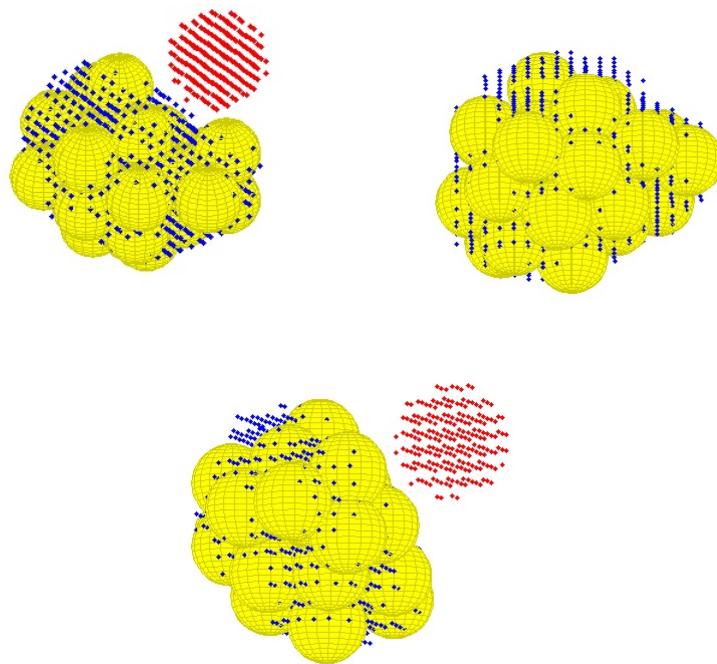


Figura 9.11: Imagens da solução encontrada pelo *Branch and Cut* para $\varepsilon = 1$, $c_r = r^3$.

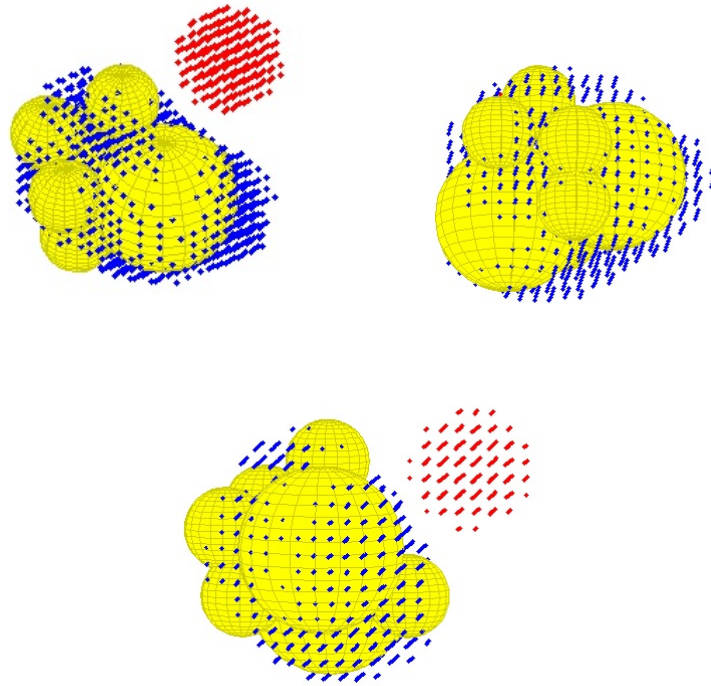


Figura 9.12: Imagens da solução encontrada pelo *Branch and Cut* para $\varepsilon = 1$, $c_4 = 20$ e $c_2 = 1$.

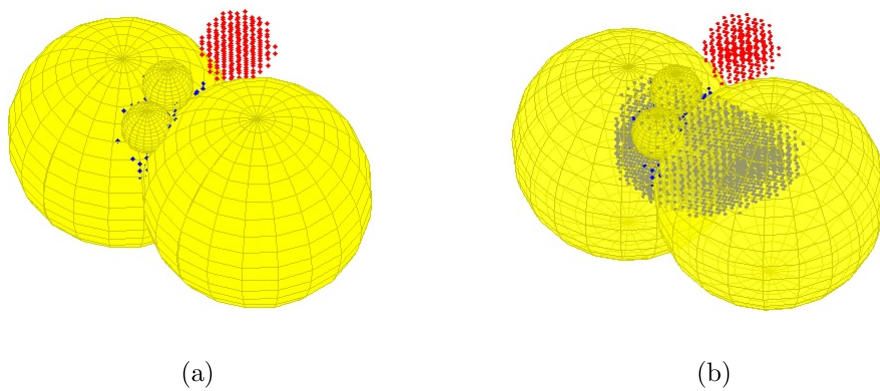


Figura 9.13: Imagens da solução encontrada pelo *Branch and Cut* sem o uso da região de segurança, com efeito de transparência em (b).

Capítulo 10

Conclusão

Estudamos o problema de recobrimento de sólidos por esferas de diferentes raios e formulamos um modelo de programação matemática inteira mista não-linear (MINLP), cuja relaxação contínua é não-convexa. Constatamos que os resolvidores atuais para esse tipo de problema são bastante suscetíveis a ótimos locais de qualidade inferior à desejada. Testamos o modelo proposto com o COUENNE, com o BONMIN e com o Xpress-SLP. O COUENNE, apesar de implementar um algoritmo cujo objetivo é resolver problemas do tipo MINLP com relaxação contínuas não-convexas, conhecido como *Spatial Branch-and-Bound*, não foi capaz de obter bons resultados em tempos curtos. O BONMIN e o Xpress-SLP, para MINLPs não-convexos, são apenas heurísticas, já que implementam métodos baseados em aproximações do modelo original. Dentre esses, o Xpress-SLP foi o que mostrou melhores resultados e em tempos bem curtos.

Da não satisfabilidade com os resultado obtidos, aplicamos técnicas de discretização ao modelo, a fim de torná-lo um problema de programação matemática inteiro e linear, cujos algoritmos de solução já foram bastante estudados e empregados. Porém, ao custo de eliminar a não-linearidade e não-convexidade do modelo, muitas variáveis binárias foram necessárias, resultando em uma explosão combinatória e necessidade de um longo de tempo de execução do algoritmo, quando não ocorreu falta de memória na máquina de teste. As melhores soluções encontradas ainda não são satisfatórias.

Desenvolvemos, então, uma heurística iterativa, baseada em otimizações sucessivas, onde os valores da solução de um problema são utilizados como ponto inicial para a resolução do problema seguinte. A idéia principal é priorizar as esferas de tamanhos maiores no recobrimento. Os resultados da heurística foram os melhores obtidos. Não sabe-se dizer o quão próximo do ótimo global estão tais soluções já que o mesmo não é conhecido e não há um banco de testes para o tipo de problema estudado. Contudo, a heurística somente pode ser utilizada quando o objeto a ser recoberto possui equações ou inequações que o descrevam.

Finalmente, para tratar os casos em que a única forma de se descrever o volume a ser recoberto é utilizando um conjunto finito de pontos como aproximação, desenvolvemos um algoritmo de *Branch and Cut* para encontrar boas soluções em tempos relativamente curtos. É realizado um pré-processamento para gerar cortes específicos, inspirados na idéia de recobrimento por esferas. Estes cortes são adicionados ao nó raiz da árvore de *branch and bound* e eliminam um número considerável de soluções.

Um resumo das soluções encontradas para a instância descrita no Capítulo 6, um paralelepípedo de dimensões 14 mm, 12 mm e 10 mm, pode ser visto na Tabela 10.1. As soluções do COUENNE com o método *spatial branch and bound*, do BONMIN com o método B-BB, um *branch and bound* heurístico, e do XPRESS com o método de *sequential linear programming* (SLP) estão representadas nas primeiras colunas da tabela. Em seguida, estão as soluções do modelo discretizado utilizando $\Delta = 1$ e resolvido pelo Xpress, a solução da heurística com o IPOPT e a solução da abordagem como um grafo pelo método de *branch and cut* proposto.

	COUENNE sB&B	BONMIN B-BB	XPRESS SLP	XPRESS $\Delta = 1$	Heur IPOPT	Grafo B&C
z^*	264	328	504	368	512	592
$ S $	12	13	35	11	29	74
t	950*	9	5	2700*	15	747*
cov	59.21	70.22	89.15	72.45	91.41	92.62
miscov	4.48	5.44	9.39	8.08	10.53	12.34
overlap	0.95	3.46	16.59	6.82	16.46	27.41

Tabela 10.1: Comparação das soluções obtidas para uma mesma instância e diferentes métodos de resolução.

Nas linhas, z^* representa o valor da solução na função objetivo, $|S|$ é a quantidade de esferas presente na solução e t é o tempo que a solução levou para ser encontrada. Um asterisco indica que a execução do algoritmo foi abortada após 3600 segundos. Os parâmetros cov, miscov e overlap são como definidos no Capítulo 6.

Como trabalhos futuros, pretendemos estudar valores para os parâmetros ε , α e c de forma que a solução do modelo seja um recobrimento completo, mas ainda matendo outros objetivos, como menor quantidade possível de esferas, com pouca sobreposição e pouca ocupação da região externa ao sólido a ser recoberto.

Referências Bibliográficas

- [1] LIBERTI, L., MACULAN, N., ZHANG, Y. “Optimal configuration of gamma ray machine radiosurgery units: the sphere covering subproblem”, *Optimization Letters*, v. 3, n. 1, pp. 109–121, 2009.
- [2] SALKIN, H. M. *Integer Programming*. Addison-Wesley Publishing Company, 1975.
- [3] SPECHT, E. Disponível em: <<http://www.packomania.com/>>. Acessado em 16/10/2014.
- [4] XAVIER, A. E., DE OLIVEIRA, A. A. F. “Optimal Covering of Plane Domains by Circles Via Hyperbolic Smoothing”, *Journal of Global Optimization*, v. 31, pp. 493–504, 2005.
- [5] XAVIER, A. E. *Penalização Hiperbólica: Um Novo Método para Resolução de Problemas de Otimização*. M.sc. thesis, PESC - COPPE / Universidade Federal do Rio de Janeiro, Rio de Janeiro, Rio de Janeiro, Brasil, 1982.
- [6] LUBKE, D. C. *Cobertura de Corpos por Esferas Utilizando Suavização Hiperbólica*. M. Sc. dissertation, PESC - COPPE / Universidade Federal do Rio de Janeiro, Rio de Janeiro, Rio de Janeiro, Brasil, 2014.
- [7] GALIYEV, S. “Computational algorithms for the optimum covering plane domains by a prescribed number of ellipses”, *Computational Mathematics and Mathematical Physics*, v. 37, n. 12, pp. 1439–1448, 1995.
- [8] TÓTH, G. F. “New results in the theory of packing and covering”, in: *P.M. Gruber, J.M. Wills (Eds.), Convexity and Its Applications*, pp. 318–359, 1983.
- [9] TÓTH, G. F. “Densest packings of typical convex sets are not lattice like”, *Discrete and Computational Geometry*, v. 14, pp. 1–8, 1995.
- [10] HALES, T. C. “A proof of the Kepler conjecture”, *Annals of Mathematics. Second Series*, v. 162, n. 3, pp. 1065–1185, 2005.

- [11] PINTÉR, J. D., KAMPAS, F. J. “MathOptimizer Professional: Key Features and Illustrative Applications”. In: *Global Optimization: From Theory to Implementation*, v. 84, *Nonconvex Optimization And Its Applications*, Springer, pp. 263–279, 2010.
- [12] STOYAN, Y. G., YASKOV, G. N. “Packing identical spheres into a cylinder”, *International Transactions in Operational Research*, v. 17, n. 1, pp. 51–70, Janeiro 2010.
- [13] SUTOU, A., DAI, Y. “Global Optimization Approach to Unequal Sphere Packing Problems in 3D”, *Journal of Optimization Theory and Applications*, v. 114, n. 3, pp. 671–694, 2002.
- [14] GANZ, J. C. “Gamma knife surgery”, *Springer-Verlag Wien, Austria*, 1997.
- [15] JITPRAPAIKULSARN, S. “An Optimization-based treatment planner for gamma knife radiosurgery”, *Ph.D. Thesis, Case Western Reserve University*, 2005.
- [16] LIM, J. “Optimization in radiation treatment planning”, *Ph.D. Thesis, University of Wisconsin-Madison*, 2002.
- [17] FERRIS, M., SHEPARD, D. “Optimization of gamma knife radiosurgery”, *Discrete Mathematical Problems with Medical Applications, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, v. 55, pp. 27–44, 2000.
- [18] FERRIS, M., VOELKER, M. “Neuro-dynamic programming for radiation treatment planning”, *Annals of Operations Research*, v. 119, pp. 247–260, 2003.
- [19] FERRIS, M., LIM, J., SHEPARD, D. “An optimization approach for the radiosurgery treatment planning”, *SIAM Journal on Optimization*, v. 13, n. 3, pp. 921–937, 2003.
- [20] TAWARMALANI, M., SAHINIDIS, N. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software and Applications*. Kluwer, Dordrecht, 2002.
- [21] DO NASCIMENTO, R. Q., MACAMBIRA, A. F. U., DOS ANJOS FORMIGA CABRAL, L., et al. “The Ellipsoid Covering problem: an Continuous Approach”, *2nd International Conference on Engineering Optimization*, 2010.

- [22] DO NASCIMENTO, R. Q., DOS SANTOS MACAMBIRA, A. F. U., PINTO, R. V., et al. “The Discrete Ellipsoid Covering problem: a Discrete Geometric Programming Approach”, *Discrete Applied Mathematics*, v. 164, pp. 276–285, 2014.
- [23] Disponível em: <<https://projects.coin-or.org/Couenne>>. Acessado em 08/10/2013.
- [24] BELOTTI, P. *COUENNE: a user’s manual*. Disponível em: <<https://projects.coin-or.org/Couenne/browser/trunk/Couenne/doc/couenne-user-manual.pdf>>.
- [25] Disponível em: <<https://projects.coin-or.org/Bonmin>>. Acessado em 08/10/2013.
- [26] BONAMI, P., LEE, J. *BONMIN User’s Manual*, Agosto 2007. Disponível em: <https://projects.coin-or.org/Bonmin/browser/stable/0.1/Bonmin/doc/BONMIN_UsersManual.pdf>.
- [27] Disponível em: <<http://www.fico.com/en/Products/DMTools/xpress-overview/Pages/Xpress-SLP.aspx>>. Acessado em 08/10/2013.
- [28] *Xpress-SLP Program Reference Manual*, Junho 2008. Disponível em: <<http://www.optiware.co.kr/data/slpref.pdf>>.
- [29] WEI, Z., LONG, Y., YE, Y. “A Sphere Packing Model for the Optimal Treatment Plan”, .
- [30] SPIRAKIS, P. G. “The volume of the union of many spheres and point inclusion problems”, *Lecture Notes in Computer Science*, v. 182, pp. 328–338, 1985.
- [31] Disponível em: <<https://projects.coin-or.org/Ipopt>>. Acessado em 31/10/2014.
- [32] Disponível em: <<http://msdn.microsoft.com/en-us/magazine/hh547104.aspx>>. Acessado em 05/11/2014.
- [33] NISKANEN, S., OSTERGARD, P. R. J. *Cliquer User’s Guide, Version 1.0*. Relatório técnico, Communications Laboratory, Helsinki University of Technology, Espoo, Finland, Tech. Rep. T48, 2003.
- [34] Disponível em: <<http://www.elekta.com/>>. Acessado em 11/11/2014.
- [35] Disponível em: <<https://www.gnu.org/software/octave/>>. Acessado em 12/11/2014.