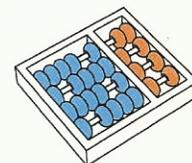


Anderson Toshiyuki Sasaki

**“Algoritmos aproximados para cobertura de objetos
geométricos por discos”**

CAMPINAS
2014



Universidade Estadual de Campinas
Instituto de Computação

Anderson Toshiyuki Sasaki

“Algoritmos aproximados para cobertura de objetos geométricos por discos”

Orientador(a): Prof. Dr. Pedro Jussieu de Rezende

Co-Orientador(a): Prof. Dr. Flávio Keidi Miyazawa

Dissertação de Mestrado apresentada ao Programa
de Pós-Graduação em Ciência da Computação do Instituto de Computação da
Universidade Estadual de Campinas para obtenção do título de Mestre em
Ciência da Computação.

ESTE EXEMPLAR CORRESPONDE À VERSÃO
FINAL DA DISSERTAÇÃO DEFENDIDA POR
ANDERSON TOSHIYUKI SASAKI, SOB
ORIENTAÇÃO DE PROF. DR. PEDRO JUS-
SIEU DE REZENDE.

A handwritten signature in blue ink, reading "Pedro de Rezende", written over a horizontal line.

Assinatura do Orientador(a)

CAMPINAS

2014

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Maria Fabiana Bezerra Muller - CRB 8/6162

Sasaki, Anderson Toshiyuki, 1988-
Sa78a Algoritmos aproximados para cobertura de objetos geométricos por discos /
Anderson Toshiyuki Sasaki. – Campinas, SP : [s.n.], 2014.

Orientador: Pedro Jussieu de Rezende.
Coorientador: Flávio Keidi Miyazawa.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de
Computação.

1. Algoritmos. 2. Otimização combinatória. 3. Teoria de computação. 4.
Geometria computacional. I. Rezende, Pedro Jussieu de, 1955-. II. Miyazawa,
Flávio Keidi, 1970-. III. Universidade Estadual de Campinas. Instituto de
Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Approximation algorithms for coverage of geometric objects by disks

Palavras-chave em inglês:

Algorithms

Combinatorial optimization

Theory of computation

Computational geometry

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Pedro Jussieu de Rezende [Orientador]

Luis Augusto Angelotti Meira

Eduardo Candido Xavier

Data de defesa: 24-02-2014

Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 24 de fevereiro de 2014, pela
Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Luis Augusto Angelotti Meira
FT / UNICAMP - Limeira



Prof. Dr. Eduardo Candido Xavier
IC / UNICAMP



Prof. Dr. Pedro Jussieu de Rezende
IC / UNICAMP

Algoritmos aproximados para cobertura de objetos geométricos por discos

Anderson Toshiyuki Sasaki

24 de fevereiro de 2014

Banca Examinadora:

- Prof. Dr. Pedro Jussieu de Rezende (*Orientador*)
- Prof. Dr. Luis Augusto Angelotti Meira
Faculdade de Tecnologia - UNICAMP-Limeira
- Prof. Dr. Eduardo Candido Xavier
Instituto de Computação - UNICAMP
- Prof. Dr. Daniel Morgato Martin
Centro de Matemática, Computação e Cognição - UFABC (*Suplente*)
- Prof. Dr. Orlando Lee
Instituto de Computação - UNICAMP (*Suplente*)

Abstract

The Minimum Set Cover problem (MSC) can be described as: given a set \mathcal{L} of elements and a collection of sets \mathcal{R} , find a subcollection $\mathcal{S} \subseteq \mathcal{R}$ that is a minimum-cost covering for \mathcal{L} , i.e., $\mathcal{L} \subseteq \bigcup_{R \in \mathcal{S}} R$, and the sum of the costs of the sets $R \in \mathcal{S}$ is minimum. Among the variants of this problem, there are those that arise from a geometric configuration in which both the elements of \mathcal{L} and the sets contained in \mathcal{R} are geometric objects. As such problems are, in general, NP-hard, if $\mathbb{P} \neq \text{NP}$, it is impossible to find polynomial-time exact algorithms for them. Thus, the use of efficient approximation algorithms to find high quality solutions becomes a good approach. In this dissertation, we study different versions of the Minimum Disk Cover problem (MDC), in which the sets in \mathcal{R} are disks, and we seek to develop approximation algorithms. These versions are related to practical problems, such as base station positioning problem for wireless network design and placement of devices in a sensor network. For the case in which the set of geometric objects to be covered is given by a single line segment, we present an FPTAS. For a more general case, where the set of geometric objects is given by a system of algebraic polynomial inequalities, we propose an approximation algorithm which we prove to be a PTAS.

Resumo

No problema de cobertura mínima por conjuntos (MSC - *Minimum Set Cover*), são dados um conjunto \mathcal{L} de objetos e uma coleção \mathcal{R} de conjuntos e deseja-se encontrar uma sub-coleção $\mathcal{S} \subseteq \mathcal{R}$ que seja uma cobertura de \mathcal{L} de custo mínimo, ou seja, $\mathcal{L} \subseteq \bigcup_{R \in \mathcal{S}} R$ com a soma dos custos dos conjuntos $R \in \mathcal{S}$ sendo mínima. Entre as variantes desse problema, existem aquelas advindas de configurações geométricas, em que tanto os elementos de \mathcal{L} quanto os conjuntos contidos em \mathcal{R} são objetos geométricos. Como tais problemas são, em geral, NP-difíceis, se $\mathbb{P} \neq \text{NP}$, não é possível encontrar algoritmos exatos de tempo polinomial para os mesmos. Assim, torna-se interessante a busca por algoritmos aproximados eficientes para obtenção de soluções com garantia de qualidade. Nesta dissertação, estudamos diferentes versões do problema de cobertura mínima por discos (MDC - *Minimum Disk Cover*), em que o conjunto \mathcal{R} é um conjunto de discos, e o objetivo é projetar algoritmos aproximados. Tais versões do problema estão relacionadas com a solução de problemas práticos, como o posicionamento de estações-base em projeto de redes sem fio ou de dispositivos em redes de sensores. Para o caso em que o conjunto de objetos geométricos \mathcal{L} é constituído de um único segmento de reta no plano, apresentamos um FPTAS. Para outra versão mais geral, na qual o conjunto de objetos geométricos é dado por um sistema de inequações polinomiais algébricas, propomos um algoritmo aproximado, o qual demonstramos ser um PTAS.

Agradecimentos

Agradeço a meus pais por tudo, sobretudo por me proporcionarem educação e pelo apoio incondicional a todas as minhas escolhas. A meus orientadores, professores Pedro e Flávio, pela paciência e dedicação ao me guiar. Aos meus amigos, por fazerem parte da minha vida, pela alegria e companheirismo. Aos meus irmãos, que são minha inspiração, motivação e orgulho, aos quais dedico este trabalho.

Sumário

Abstract	ix
Resumo	xi
Agradecimentos	xiii
1 Introdução	1
1.1 Preliminares	2
1.2 Trabalhos relacionados	3
2 Cobertura de Pontos	7
2.1 Um PTAS para a cobertura de pontos	7
2.1.1 Definições	8
2.1.2 Descrição do algoritmo de Bilò et al.	10
2.1.3 Análise do algoritmo	13
2.2 Um Algoritmo exato para a cobertura de pontos no plano	17
2.2.1 Estrutura da solução ótima e lemas iniciais	18
2.2.2 Descrição do algoritmo de Gibson et al.	21
2.2.3 Análise do algoritmo	25
3 Cobertura de Segmento	29
3.1 Um Algoritmo exato para a cobertura de um segmento	30
3.1.1 Transformação da instância em um grafo direcionado	30
3.1.2 Obtenção de uma solução ótima e análise do algoritmo	33
3.2 Um FPTAS para a cobertura de um segmento	34
3.2.1 Descrição do algoritmo proposto	34
3.2.2 Análise do algoritmo	38
4 Cobertura de Objetos Geométricos	41
4.1 Um PTAS para cobertura de objetos geométricos	41

4.1.1	Descrição do algoritmo proposto	42
4.1.2	Análise do algoritmo	47
4.1.3	Alternativa para um caso mais restrito	50
5	Conclusões	53
	Referências Bibliográficas	55
A	Lista de acrônimos	57

Lista de Figuras

2.1	Exemplo de construção da grade para $\lambda = 3$	9
2.2	Linhas ativas e j -squares para $\lambda = 3$	10
2.3	Quadrados relevantes, com um exemplo de pai e filho.	11
2.4	O conjunto I^Q (linha contínua) do j -square Q destacado. Os dois discos maiores de linha destacada compõem $I_{<j}^Q$ e são a projeção do pseudoagrupamento em $I_{<j}^Q$	12
2.5	Um disco de A (linha contínua) e o disco de A^* do qual se originou. Se um ponto de X_2 foi atribuído ao disco original por ψ^* , então o disco aumentado cobre o quadrado inteiro.	17
2.6	Um separador para um retângulo está distante pelo menos $1/3$ do comprimento de qualquer um dos lados mais curtos.	19
2.7	Um retângulo de comprimento a pode ser envolvido por um disco B de raio a . B' é concêntrico a B e possui raio $2a$	20
2.8	A área de um disco de raio $\frac{5a}{2}$ é 25 vezes maior que um disco de raio $\frac{a}{2}$	20
2.9	Em cada setor de ângulo $2\pi/15$, \mathcal{S}' deve conter apenas um disco centrado no exterior de B' que intercepta B	21
2.10	Linhas críticas (pontilhadas) e canônicas (contínuas).	22
2.11	Ilustração para o Lema 2.2.3.	26
3.1	Exemplo da construção de um grafo a partir de uma instância.	33
3.2	Exemplo da discretização de raios.	35
3.3	Exemplo da substituição do disco original (linha pontilhada) por um disco centrado sobre o segmento (linha cheia).	36
4.1	Exemplo de arranjo gerado a partir da discretização dada por R	45
4.2	Exemplo de pontos representantes para um polígono.	46
4.3	Passos do algoritmo proposto.	47

Capítulo 1

Introdução

O problema de cobertura por conjuntos (MSC - *Minimum Set Cover*) é um dos mais estudados problemas de otimização. Neste problema, são dados um conjunto de elementos $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$ e uma coleção de conjuntos $\mathcal{R} = \{R_1, R_2, \dots, R_m\}$, e o objetivo é encontrar uma subcoleção mínima $\mathcal{S} \subseteq \mathcal{R}$ que seja uma cobertura para \mathcal{L} , ou seja, $\mathcal{L} \subseteq \bigcup_{R \in \mathcal{S}} R$. Recentemente, variantes geométricas deste problema têm recebido maior atenção devido à sua aplicação direta para solução de muitos problemas práticos. Entre estas variantes está o problema de cobertura mínima por discos (MDC - *Minimum Disk Cover*).

No problema de cobertura mínima por discos, cada conjunto pertencente à coleção \mathcal{R} é um disco no plano. Muitos problemas práticos podem ser modelados como problemas de cobertura por discos. Uma aplicação, por exemplo, é o problema do posicionamento de antenas que provêm sinal de telefonia celular num determinado raio com o objetivo de cobrir uma região, permitindo acesso à rede. Este problema pode ser modelado como um problema de cobertura por discos em que o conjunto de antenas é representado por uma coleção de discos \mathcal{R} , a região a ser coberta é representada por um conjunto \mathcal{L} de pontos no plano e deve ser determinado o posicionamento dos centros dos discos que cubram \mathcal{L} , minimizando o número desses discos.

O problema de cobertura por discos, por sua vez, possui muitas variantes, nas quais são adicionadas restrições ao problema mais geral, como a forma dos elementos geométricos a serem cobertos, restrições no tamanho dos raios dos discos, limitações no posicionamento dos centros dos discos, entre outras.

Nesta dissertação, serão apresentados os resultados obtidos para diferentes versões do problema de cobertura por discos. Em uma das versões, é considerado o problema da cobertura de um segmento de reta no plano por discos cujos centros são dados por um conjunto de pontos. O problema, então, consiste na determinação dos raios dos discos centrados nos pontos dados de forma que cubram todos os pontos do segmento com custo

mínimo. O custo total é dado pela soma dos custos dos discos, especificados por funções polinomiais dos raios dos discos.

Em outra versão, mais geral, é considerado o problema de cobertura de objetos geométricos descritos por um sistema de inequações polinomiais algébricas. Tais objetos compreendem uma grande variedade de objetos, cujas bordas são descritas por polinômios. Por exemplo, um polígono pode ser descrito por um sistema cujas equações são polinômios lineares, ou seja, o objeto é limitado por retas. Nessa versão, assim como na anterior, são dados pontos nos quais devem ser centrados os discos, sendo o objetivo cobrir todos os pontos dos objetos geométricos. O custo de uma solução é dado pela soma dos custos dos discos, descritos por funções polinomiais dos raios dos discos.

Para cada um dos casos foi elaborado um algoritmo aproximado e tais resultados foram publicados na revista *Operations Research Letters* [8].

Nas próximas seções deste capítulo, serão apresentados os conceitos e definições necessárias para a leitura do restante da dissertação e os trabalhos relacionados encontrados na literatura. No Capítulo 2, são apresentados algoritmos para cobertura de pontos que são utilizados como subrotinas para o projeto dos algoritmos que desenvolvemos no Capítulo 4. No Capítulo 3, propomos um algoritmo aproximado para solução do problema de cobertura de um segmento. Em seguida, no Capítulo 4, apresentamos um algoritmo aproximado para solução do problema de cobertura de objetos geométricos.

A escolha da ordem em que os problemas são apresentados se deve à complexidade crescente dos objetos a serem cobertos. No caso pontos neste capítulo; segmentos de reta no Capítulo 3; objetos geométricos descritos por um sistema de inequações polinomiais no Capítulo 4.

Ao final, no Capítulo 5, apresentamos as nossas considerações finais sobre esta dissertação.

1.1 Preliminares

Muitos dos problemas estudados em otimização combinatória, como é o caso da cobertura mínima por discos, são NP-difíceis, o que significa que é impossível encontrar um algoritmo de tempo polinomial sob a hipótese de que $\mathbb{P} \neq \text{NP}$, que encontre solução ótima exata para o problema. Neste contexto, torna-se interessante a aplicação de algoritmos de aproximação.

Algoritmos de aproximação são algoritmos que não necessariamente encontram soluções ótimas para o problema, mas computam eficientemente soluções viáveis com uma garantia de qualidade. Ou seja, sacrifica-se a otimalidade das soluções obtidas, permitindo que sejam encontradas soluções cujo custo é limitado por um fator da solução ótima, em tempo polinomial. Os algoritmos de aproximação são classificados conforme as garantias dadas para a qualidade das soluções encontradas.

Seja I uma instância de um problema de minimização. O custo da solução dada por um algoritmo A para a instância I do problema é denotada por $A(I)$. Seja $OPT(I)$ o custo de uma solução ótima para a instância I . Se houver uma garantia de que existe uma constante k tal que, para qualquer instância I , $|A(I) - OPT(I)| \leq k$, diz-se que A tem aproximação absoluta. Isto quer dizer que, independentemente do tamanho da instância I , a solução encontrada pelo algoritmo A sempre estará distante da solução ótima no máximo de uma constante k .

Caso exista uma garantia que, para qualquer instância I , exista um α tal que $A(I) \leq \alpha OPT(I)$, diz-se que A é um algoritmo de aproximação com fator α ou que A é uma α -aproximação. Caso α seja uma constante, A é um algoritmo de aproximação de fator constante. Isto quer dizer que, independentemente da instância I do problema, o algoritmo A encontrará soluções em um fator de, no máximo, α da solução ótima. Definições análogas existem para problemas de maximização, ou seja, para um problema de maximização, se existir um α tal que $A(I) \geq \alpha OPT(I)$, A é um algoritmo com fator de aproximação α ou α -aproximado. Para problemas de minimização, $\alpha \geq 1$ e para problemas de maximização, $0 < \alpha \leq 1$. Quando $\alpha = 1$, A encontra soluções ótimas para o problema.

Para um problema de minimização, uma família de algoritmos polinomiais A_ε , tal que A_ε é uma $(1 + \varepsilon)$ -aproximação para todo $\varepsilon > 0$, é um esquema de aproximação de tempo polinomial (PTAS - *Polynomial Time Approximation Scheme*). De forma análoga, para um problema de maximização, uma família de algoritmos em que A_ε é uma $(1 - \varepsilon)$ -aproximação para todo $\varepsilon > 0$ é um PTAS. Caso um PTAS A_ε execute em tempo polinomial no tamanho da entrada e em $\frac{1}{\varepsilon}$, A_ε é um esquema de aproximação de tempo completamente polinomial (FPTAS - *Fully Polynomial Time Approximation Scheme*). Um PTAS permite que sejam encontradas soluções aproximadas tão próximas da solução ótima quanto se desejar, sendo necessário considerar um ε tão pequeno quanto se queira. No entanto, diminuir o valor de ε resulta no aumento do tempo necessário para encontrar uma solução, devendo-se, portanto, analisar a viabilidade da obtenção de uma solução muito próxima da ótima.

1.2 Trabalhos relacionados

Há diversos trabalhos na literatura que estudam o problema de cobertura por discos, os quais normalmente utilizam diferentes restrições sobre o problema. As restrições mais comuns são as que limitam os raios ou o posicionamento dos centros dos discos. Um exemplo de limitação sobre os raios dos discos é a exigência de que todos os discos tenham o mesmo raio. Neste caso o objetivo seria encontrar o posicionamento do menor número de discos que cubram os objetos dados. Já uma restrição sobre o posicionamento dos

centros dos discos poderia ser a exigência de que sejam escolhidos dentre um conjunto finito de pontos. Assim, os centros dos discos seriam dados na entrada do problema.

No artigo [6], Das et al. estudam o problema da cobertura por discos unitários discretos (DUDC - *Discrete Unit Disk Cover*) em que se deseja cobrir pontos no plano com discos de raio unitário cujos centros só podem ser posicionados em pontos previamente definidos. O problema consiste, então, em selecionar o menor número de discos para a composição de uma cobertura. Para tal problema, apresentam um algoritmo aproximado que fornece um fator de aproximação constante igual a 18. A versão tratada no artigo está relacionada com a versão que consideramos nesta dissertação pelas restrições aplicadas sobre o posicionamento dos centros dos discos, os quais devem ser posicionados apenas em pontos contidos num dado conjunto finito.

Para o mesmo problema DUDC, Mustafa e Ray [17] desenvolveram um PTAS. O algoritmo se baseia em busca local e tem complexidade de tempo $O(m^{2(\frac{\epsilon}{\epsilon})^2+1}n)$, onde $n = |\mathcal{L}|$ é o número de pontos a serem cobertos e $m = |\mathcal{R}|$ é o número de discos. O tempo de execução do algoritmo é muito elevado, não sendo possível utilizá-lo na prática. Por esse motivo, os algoritmos aproximados de fator constante e tempo polinomial continuam interessantes.

O problema da cobertura por discos é conhecido também como o problema do posicionamento de estações-base (BSPP - *Base Station Placement Problem*). Este problema é tratado em sua versão discreta, MDDC (*Minimum Discrete Disk Cover*), em que o conjunto de pontos em que os centros dos discos (ou estações-base) podem ser posicionados é finito, por Lev-Tov e Peleg [15] e também por Alt et al. [3]. Em [15], é estudado o caso em que a função de custo é linear, no qual deseja-se minimizar a soma dos raios dos discos. Primeiramente é considerado um caso especial em uma dimensão, no qual tanto os pontos a serem cobertos como os centros dos discos encontram-se sobre uma linha. Para este caso é apresentado um algoritmo exato de programação dinâmica, que encontra soluções ótimas em tempo $O((n+m)^3)$, em que $n = |\mathcal{L}|$ é o número de pontos a serem cobertos e $m = |\mathcal{R}|$ é o tamanho do conjunto de centros de discos. São consideradas duas versões para o caso bidimensional, em que tanto os pontos a serem cobertos como os centros dos discos são pontos distribuídos do plano. Na primeira, são utilizados discos de raios unitários e o conjunto dos centros dos discos \mathcal{R} é um subconjunto dos pontos a serem cobertos, ou seja, $\mathcal{R} \subseteq \mathcal{L}$. Já na segunda versão, mais geral, os raios dos discos são variáveis e o conjunto \mathcal{R} não é necessariamente um subconjunto de \mathcal{L} . Para as duas versões, são apresentados PTAS s.

Para o mesmo problema MDDC, mas considerando funções de custo quadráticas, Freund e Rawitz [11] apresentaram um algoritmo de tempo polinomial 9-aproximado, baseado no método primal-dual.

Em [3], são apresentados resultados para vários problemas relacionados. Para o mesmo

caso 1-D considerado em [15], é apresentado um algoritmo de tempo linear que fornece uma 3-aproximação para o problema e um outro algoritmo 2-aproximado de tempo $O(m + n \log(m))$. Para outros problemas, são consideradas diferentes restrições para o posicionamento dos centros dos discos. Neste artigo também é provado que, para qualquer função de custo superlinear, o problema discreto MDDC é NP-difícil, reforçando a dificuldade do problema que havia sido provada anteriormente em [5].

Bilò et al.[5], além de mostrar que o caso discreto MDDC é NP-difícil para qualquer função de custo superquadrática, apresentou um PTAS para o caso 2-D com funções de custo superquadráticas. Este algoritmo será apresentado em detalhes na Seção 2.1, pois o utilizamos como subrotina para o algoritmo aproximado que propomos no Capítulo 4.

Além das restrições sobre os discos, podem ser aplicadas restrições sobre o conjunto de pontos a serem cobertos. Assim, o mesmo pode ser dado não por um conjunto finito de pontos, mas por aqueles contidos em objetos geométricos, como um segmento de reta. Devido à característica contínua do segmento, existem infinitos pontos a serem cobertos, o que exige um tratamento diferente daquele dado para cobertura de um conjunto discreto de pontos.

No artigo [2], Agnetis et al. tratam do problema de cobertura de um segmento utilizando no máximo n discos de raio variável e minimizando o custo total, sendo a função de custo dos discos quadrática nos raios dos discos. São considerados dois casos: no primeiro, todos os discos têm a mesma função de custo, e no segundo, são consideradas funções de custo arbitrárias para cada disco. Para o primeiro caso, é proposto um algoritmo exato simples, baseado em busca binária, de complexidade de tempo polinomial $O(C \log(n))$, onde C é igual ao máximo custo computacional para o cálculo da função de custo ω para um dos discos. Quando as funções de custo são arbitrárias, o problema se torna mais complexo, mas sua complexidade não foi analisada. Para este caso, foi elaborado um algoritmo exato de *branch and bound* para solução do problema o qual utiliza heurísticas de tempo polinomial para encontrar limitantes superiores e inferiores.

Já no artigo [7] de Dash et al., considera-se um problema de cobertura de segmentos por discos de raios iguais a uma constante r , no qual a cobertura recebe uma definição diferente da considerada usualmente: um segmento de reta é considerado coberto caso um dos pontos pertencentes ao segmento esteja coberto. Inicialmente, provam que o problema é NP-completo. Em seguida, tratam uma versão do problema em que os segmentos a serem cobertos devem ter orientações paralelas aos eixos. Para esse caso, apresentam um algoritmo aproximado de fator constante 12 e um PTAS. Para outra versão, em que os segmentos podem estar em qualquer orientação mas têm tamanho limitado, é proposto um PTAS.

Em outro caso, os objetos geométricos a serem cobertos são restritos a polígonos no plano. Para este caso, existem trabalhos que buscam uma k -cobertura para os pontos no

interior dos polígonos, ou seja, cada ponto deve ser coberto por no mínimo k discos.

Um dos trabalhos que tratam da k -cobertura de polígonos no plano é apresentado por Abu-Affash et al. [1]. Nesse trabalho, consideram os custos dos discos proporcionais às suas áreas, ou seja, são funções quadráticas dos raios. São considerados dois casos: a k -cobertura de pontos e a k -cobertura de um polígono. Para a k -cobertura de pontos, apresentam um algoritmo que fornece soluções de custo no máximo $(23, 02 + 63, 95(k - 1))$ vezes o custo de uma solução ótima. Para a k -cobertura de um polígono, é proposto um algoritmo que fornece soluções de custo no máximo $(63, 94 + 177, 64(k - 1))$ vezes o custo de uma solução ótima. Tal algoritmo utiliza uma redução do problema de cobertura de um polígono para a cobertura de pontos. Para o caso em que $k = 1$ e utilizando o algoritmo proposto por Freund e Rawitz em [11] como subrotina, obtém-se uma 25-aproximação para a 1-cobertura do polígono.

Bar-Yehuda e Rawitz [4] também estudaram o problema da k -cobertura de um polígono. De maneira semelhante a [1], o algoritmo proposto se baseia na redução do problema de cobertura de um polígono para o de cobertura de pontos. A versão considerada em [4] para o problema de cobertura de pontos permite que cada ponto tenha um requisito diferente de cobertura, ou seja, cada ponto p_i deve ser coberto por k_i discos. Define-se por k_{\max} o maior k_i . Para esse problema, propõem um algoritmo $3^\alpha k_{\max}$ -aproximado, onde o custo de um disco centrado em um ponto p_i é dado por $\omega_i r_i^\alpha$, sendo $\omega_i \geq 1$ o parâmetro de potência e r_i o seu raio. Demonstra-se que o algoritmo proposto é uma $(3^\alpha k + \varepsilon)$ -aproximação para o problema de cobertura de um polígono para qualquer $\varepsilon \geq 0$. Assim, para a cobertura simples de um polígono, o algoritmo de [4] fornece uma $(3^\alpha + \varepsilon)$ -aproximação.

Capítulo 2

Cobertura de Pontos

Neste capítulo serão apresentados algoritmos para o problema de cobertura de pontos por discos. Tal problema pode ser considerado uma versão geométrica do problema de cobertura por conjuntos, onde os conjuntos são dados por discos e os pontos a serem cobertos são posicionados sobre o plano cartesiano. Problemas práticos, como o posicionamento de estações base para prover conectividade para terminais posicionados em determinados pontos ou de sensores para o monitoramento de uma determinada região, podem ser modelados como uma instância desse problema.

De maneira mais formal, seja \mathcal{C} um conjunto de n pontos no plano que determinam os centros dos discos e \mathcal{L} o conjuntos de pontos no plano a serem cobertos. Para cada ponto $c_i \in \mathcal{C}$, considere uma função de custo $\Omega_i(r) = F_i + \omega(r)$, onde F_i é a parcela fixa do custo para um disco centrado em c_i e $\omega(r)$ é a parcela variável, dada por uma função do raio do disco centrado em c_i . Caso o raio de um disco seja nulo, o seu custo é zero, independente da parcela fixa da função Ω_i . O objetivo é determinar uma lista de valores $\mathcal{S} = \{r_1, r_2, \dots, r_n\}$ para os raios dos discos centrados nos pontos $c_i \in \mathcal{C}$ de forma que todos os pontos de \mathcal{L} sejam cobertos com custo mínimo. O custo de uma solução é dado pela soma dos custos dos discos determinados por \mathcal{S} . Foi demonstrado em [3] que este problema é NP-difícil para qualquer função de custo superlinear.

Na Seção 2.1 será apresentado um algoritmo aproximado proposto em [5] e na Seção 2.2 será apresentado um algoritmo exato de tempo polinomial para um caso mais restrito, proposto em [12]. Os algoritmos apresentados em tais artigos serão utilizados como subrotina para o algoritmo proposto no Capítulo 4.

2.1 Um PTAS para a cobertura de pontos

Em [5], Bilò et al. consideram o problema *Min-Size k -Clustering Problem* (MSCP), ou Problema de k -Clusterização de Tamanho Mínimo, o qual é definido a seguir. Dado um

conjunto \mathcal{L} de n pontos no espaço d -dimensional, seja \mathcal{F} um conjunto de custos fixos F_p para cada ponto $p \in \mathcal{L}$. Seja $f(r) = r^\alpha$ uma função em que $\alpha \geq 1$ é uma constante e seja k uma constante inteira. O problema MSCP consiste em determinar os raios r_i para no máximo k discos centrados em pontos $p_i \in \mathcal{L}$ de forma que todos os pontos de \mathcal{L} sejam cobertos com custo total mínimo. É utilizada a denominação *cluster* como um sinônimo para disco e *clusterização* ou *clustering* como sinônimo de cobertura. O custo é dado por $\omega = \sum_{p_i \in \mathcal{L} | r_i > 0} f(r_i) + F_{p_i}$. Uma instância para este problema é dada pela tupla $(\mathcal{L}, \mathcal{F}, d, \alpha)$.

Bilò et al. apresentam um PTAS para o MSCP com complexidade de tempo $O\left(n^{\frac{\alpha}{\varepsilon} O(d)}\right)$. Em particular, para $d = 2$, apresentam um PTAS com complexidade $O\left(n^{\frac{\alpha^4}{\varepsilon^6}}\right)$. Também apresentam uma demonstração de que o problema MSCP é NP-difícil para qualquer $\alpha \geq 2$.

Serão apresentados restante desta seção o algoritmo proposto em [5] e sua análise.

2.1.1 Definições

O algoritmo utiliza a idéia da subdivisão do plano baseada no algoritmo de Erlebach et al. [10] o qual apresenta uma aproximação para o problema da cobertura mínima de vértices de um grafo de discos (Minimum Vertex Cover of Disk Graphs).

Considerando uma instância $(\mathcal{L}, \mathcal{F}, 2, \alpha)$ e $\lambda > 1$ um inteiro positivo. Seja \mathcal{D} o conjunto de todos os n^2 possíveis discos centrados em um ponto de \mathcal{L} e cujo raio é a distância entre o centro e cada um dos outros pontos. Dentre todos os possíveis raios não-nulos de \mathcal{D} , sejam r_{\min} e r_{\max} respectivamente o menor e o maior dos raios. Particiona-se \mathcal{D} em $L + 1$ níveis, onde $L = \lceil \log_{\lambda+1}(r_{\max}/r_{\min}) \rceil$. Para $0 \leq j \leq L$, o nível j consiste em todos os discos D_i cujo raio r_i é tal que $(\lambda + 1)^{-j} r_{\max} \geq r_i > (\lambda + 1)^{-(j+1)} r_{\max}$. Dessa forma, quanto menor for o nível, maior serão os raios dos discos contidos. Portanto, os discos de raio r_{\min} estará contido no nível L . Assume-se que os discos de raio nulo também estão contidos no nível L .

Para cada nível j , é estabelecida uma grade formada por linhas verticais e horizontais distantes de $2(\lambda + 1)^{-j} r_{\max}$ entre si. Assim a v -ésima linha vertical, cujo índice $v \in (-\infty, \infty)$ é um inteiro, localiza-se na posição $x = 2v(\lambda + 1)^{-j} r_{\max}$ e, da mesma forma, a h -ésima linha horizontal de índice inteiro $h \in (-\infty, \infty)$ se encontra na posição $y = 2h(\lambda + 1)^{-j} r_{\max}$. Diz-se que um disco D_i de centro (x_i, y_i) e raio r_i intercepta a linha vertical em $x = a$ caso $a - r_i < x_i \leq a + r_i$. De maneira análoga, diz-se que tal disco intercepta a linha horizontal em $y = b$ se $b - r_i < y_i \leq b + r_i$. Ou seja, um disco intercepta uma linha se possui uma intersecção não nula com a linha, exceto se apenas tangenciar a linha pela esquerda ou por baixo. Nota-se que um disco intercepta no máximo uma linha vertical e uma linha horizontal no nível em que se encontra. A Figura 2.1 apresenta um exemplo da construção da grade para $\lambda = 3$.

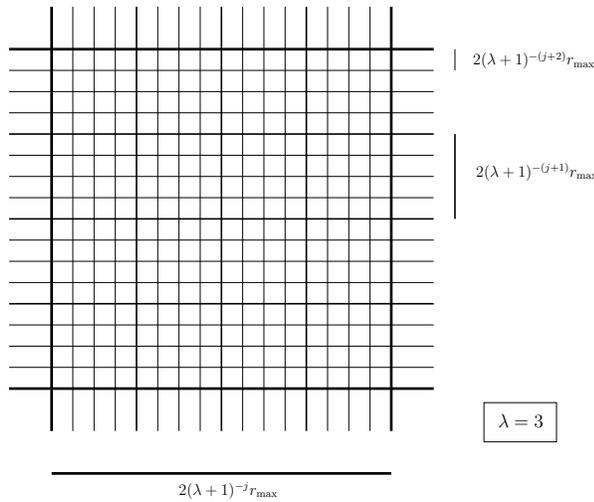


Figura 2.1: Exemplo de construção da grade para $\lambda = 3$.

Sejam $0 \leq s, t < \lambda$. Diz-se que as linhas verticais cujos índices módulo λ são iguais a s e as linhas horizontais cujos índices módulo λ são iguais a t são ativas para (s, t) . Para um dado nível j , as linhas ativas para (s, t) particionam o plano em quadrados, chamados de j -squares. Observa-se que todo $(j + 1)$ -square encontra-se totalmente contido em algum j -square. A Figura 2.2 ilustra um exemplo destacando linhas ativas e j -squares para diferentes níveis.

Um j -square Q é dito relevante caso exista um disco de nível j em \mathcal{D} que contenha um ponto $p \in Q \cap \mathcal{L}$. Observa-se que o número de quadrados relevantes é polinomial em n , já que o número de discos é n^2 e cada disco pode cobrir pontos de no máximo 4 quadrados no seu nível. Isso se deve ao fato de que um disco intercepta no máximo uma linha vertical e uma linha horizontal no seu nível. Para um j -square relevante Q e um j' -square relevante Q' , onde $j' > j$, diz-se que Q é pai de Q' (e que Q' é filho de Q) caso Q' esteja contido em Q e não exista um j'' -square relevante Q'' , com $j' > j'' > j$ e tal que Q' esteja contido em Q'' e Q'' esteja contido em Q . Observa-se que existem apenas 4 0-squares relevantes, os quais são os únicos que não possuem um pai. A Figura 2.3 destaca j -squares relevantes para diferentes níveis e um exemplo de quadrados relevantes pai e filho.

Considerando um j -square Q , seja I^Q o conjunto de discos em \mathcal{D} que têm intersecção não nula com Q . Denota-se por $I_{<j}^Q$ o conjunto de discos em I^Q de nível menor que j e, analogamente, se define $I_{\leq j}^Q, I_{=j}^Q, I_{\geq j}^Q$ e $I_{>j}^Q$. Um conjunto $C \subseteq I^Q$ é um pseudoagrupamento (*pseudoclustering*) de Q se para cada ponto $p \in \mathcal{L} \cap Q$ existe um disco em C que contém p . Ou seja, um pseudoagrupamento para um j -square é um conjunto de discos que cobrem todos os pontos de \mathcal{L} contidos no j -square. Para um pseudoagrupamento C

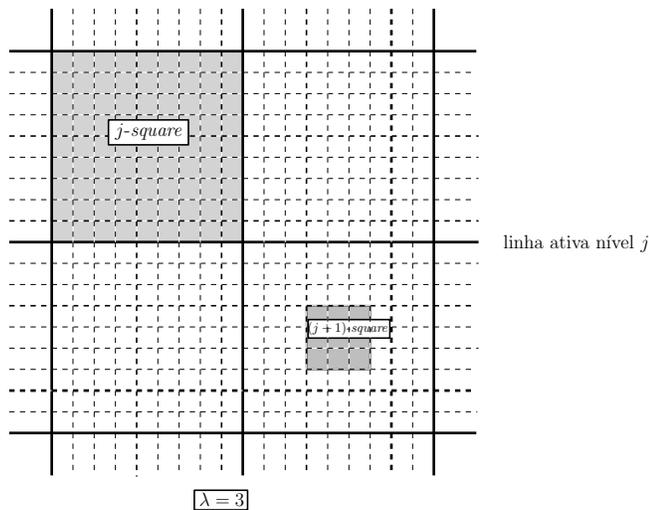


Figura 2.2: Linhas ativas e j -squares para $\lambda = 3$.

de Q , diz-se que o conjunto $I_{<j}^Q \cap C$ é a projeção de C em $I_{<j}^Q$ (e similarmente para $I_{\leq j}^Q$). Assim, a projeção de um pseudoagrupamento C em $I_{<j}^Q$ é o subconjunto de C cujos discos têm nível menor que j , ou seja, são os maiores discos do pseudoagrupamento.

A Figura 2.4 apresenta o conjunto I^Q (formado pelos discos de linha contínua) para o j -square Q destacado. Os dois discos maiores, de linha destacada, compõem a projeção do pseudoagrupamento dado por I^Q em $I_{<j}^Q$.

2.1.2 Descrição do algoritmo de Bilò et al.

Dada uma instância $(\mathcal{L}, F, 2, \alpha)$ do problema MSCP, o algoritmo atribui um nível para cada um dos possíveis discos definidos por \mathcal{L} e implicitamente define as linhas horizontais e verticais da grade, como descrito na subseção anterior. Então, para cada possível valor de $s, t \in \{0, \dots, \lambda - 1\}$, é executada uma iteração na qual uma k -cobertura é computada. A melhor solução encontrada dentre todas as iterações é dada como a solução final.

Em cada iteração associada a (s, t) , o algoritmo processa todos os quadrados relevantes definidos pela subdivisão do plano de acordo com s e t em ordem decrescente de níveis. Ou seja, os níveis maiores, que contém os discos de menor raio, são processados prioritariamente. Num j -square relevante Q , as projeções de um número polinomial de pseudoagrupamentos de Q são enumeradas. Durante esse processo de enumeração, a tabela $Table_Q$ é construída consultando as respectivas entradas armazenadas nas tabelas dos filhos de Q . A entrada $Table_Q(P, i)$ para a projeção $P \subseteq I_{<j}^Q$ de um pseudoagrupamento de Q em $I_{<j}^Q$ e um inteiro i , tal que $1 \leq i \leq k$, será um conjunto $J \subseteq I_{\geq j}^Q$ tal que $P \cup J$ é um pseudoagrupamento de Q com exatamente i discos. Ou seja, a entrada $Table_Q(P, i)$

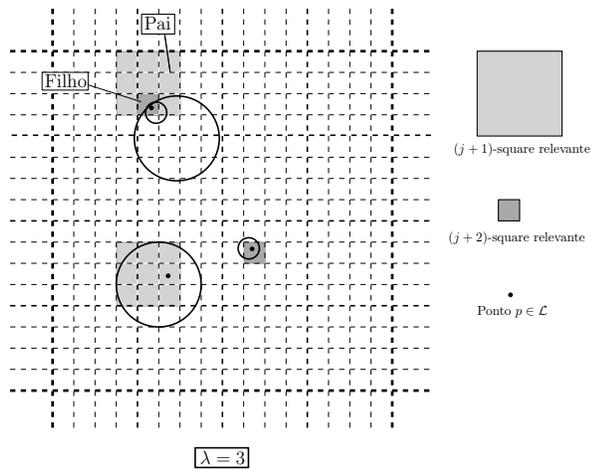


Figura 2.3: Quadrados relevantes, com um exemplo de pai e filho.

armazenará um conjunto J de discos de tamanho igual ou menor que os do nível j , tais que, unidos à projeção P , cobrem, com o menor custo, todos os pontos de \mathcal{L} contidos no j -square Q utilizando i discos. Ao final de uma iteração, o algoritmo computa uma k -cobertura de \mathcal{L} por meio da enumeração de todas as coberturas obtidas pela escolha de entradas da tabela $Table_Q$ sobre todos os 4 quadrados relevantes Q que não têm pais.

O pseudocódigo para a construção da tabela $Table_Q$ a partir das tabelas $Table_{Q'}$ construídas para todos os filhos Q' de Q é apresentado no Algoritmo 1. O parâmetro ξ é utilizado para restringir o tamanho dos pseudoagrupamentos de Q considerados e $\omega(\cdot)$ denota o custo de um disco ou de um conjunto de discos.

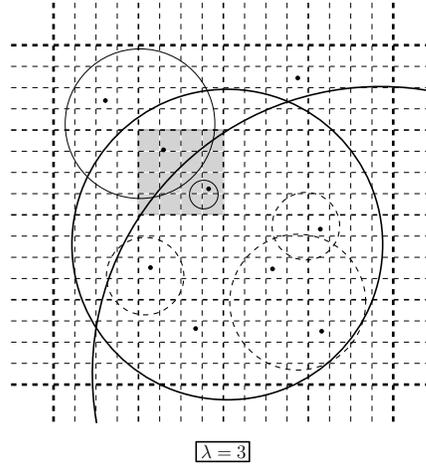


Figura 2.4: O conjunto I^Q (linha contínua) do j -square Q destacado. Os dois discos maiores de linha destacada compõem $I_{<j}^Q$ e são a projeção do pseudoagrupamento em $I_{<j}^Q$.

Algoritmo 1: Construção de $Table_Q$ para j -square relevante Q

Entrada: j -square relevante Q , $Table_{Q'}$ construídas para todos os filhos Q' de Q , inteiro k

Saída: $Table_Q$, que contém uma entrada $Table_Q(P, i)$ para cada par (P, i) , sendo $P \subseteq I_{<j}^Q$ uma projeção de Q em $I_{<j}^Q$, $1 \leq i \leq k$. Cada entrada $Table_Q(P, i)$ recebe $J \subseteq I_{\geq j}^Q$ tal que $P \cup J$ é um pseudoagrupamento de Q com exatamente i discos

```

1  $Table_Q \leftarrow \emptyset$ ;
2  $I_{\leq j}^Q \leftarrow$  todos os discos de  $\mathcal{D}$  de nível no máximo  $j$  que intersectam  $Q$ ;
3 para todo  $Q \subseteq I_{\leq j}^Q$  tal que  $|Q| \leq \min\{\xi, k\}$  faça
4    $J \leftarrow \{D \in Q \mid D \text{ tem nível } j\}$ ;
5    $P \leftarrow \{D \in Q \mid D \text{ tem nível menor que } j\}$ ;
6   se  $Q$  não tem filhos então
7      $Table_Q(P, |Q|) \leftarrow J$ ;
8   senão
9     Sejam  $Q_1, Q_2, \dots, Q_t$  os filhos de  $Q$ ;
10    para todo filho  $Q_y$  de  $Q$  faça
11       $P'(Q_y) \leftarrow \{D \in Q \mid D \text{ intersecta } Q_y\}$ ;
12    para toda combinação de  $(i_1, i_2, \dots, i_t)$  com  $1 \leq i_y \leq k$  para  $y = 1, \dots, t$ 
13      faça
14         $J' \leftarrow J \cup \bigcup_{y=1}^t Table_{Q_y}(P'(Q_y), i_y)$ ;
15         $i' = |P \cup J'|$ ;
16        se  $i' \leq k$  e  $P \cup J'$  é um pseudoagrupamento de  $Q$  então
17          se  $Table_Q(P, i')$  não está definido ou  $\omega(J') < \omega(Table_Q(P, i'))$  então
             $Table_Q(P, i') \leftarrow J'$ ;

```

2.1.3 Análise do algoritmo

Bilò et al. demonstram em [5] que o Algoritmo 1 por eles apresentado é um PTAS para o problema MSCP. A seguir serão apresentados os lemas e os teoremas por eles demonstrados, com comentários sobre a idéia utilizada para a prova dos mesmos.

Lema 2.1.1. *Cada quadrado relevante tem no máximo $O(\lambda^4)$ filhos.*

O Lema 2.1.1 é demonstrado baseando-se na construção da grade, utilizando a limitação das distâncias entre os centros dos discos para limitar o número de filhos de um quadrado relevante em cada nível. Para um quadrado relevante Q de nível j e lado ℓ , observa-se que existem no máximo $(\lambda + 1)^4$ filhos de níveis $j + 1$ e $j + 2$. Isso se deve ao fato de que cada quadrado de nível j ser dividido por $(\lambda + 1)$ linhas horizontais e $(\lambda + 1)$ linhas verticais, totalizando $(\lambda + 1)^2$ quadrados de nível $j + 1$. Cada quadrado de nível $j + 1$ resultantes dessa divisão também é dividido em $(\lambda + 1)^2$ quadrados de nível $j + 2$, resultando assim no limitante de $(\lambda + 1)^4$ quadrados. Se Q tem mais de $(\lambda + 1)^4$ filhos, então Q possui, necessariamente, filhos em quadrados de nível $j + 3$ ou maior. Seja Q' um filho de Q o qual possui o menor nível $j' \geq j + 3$ dentre os filhos de Q e seja p um ponto de Q' . Então, todos os outros pontos estarão a uma distância de p menor que $\frac{\ell}{2(\lambda+1)^{j'-j+1}}$ ou pelo menos $\frac{\ell}{2(\lambda+1)^2}$, pois, em caso contrário, um quadrado Q'' contendo Q' seria o filho de Q ao invés de Q' . Observa-se, então, que num disco de raio $\frac{\ell}{4(\lambda+1)^2}$ centrado em p existem no máximo quatro filhos de Q de nível pelo menos $j + 3$, incluindo Q' , pois este é o número máximo de quadrados que podem ter um ponto a uma distância menor que $\frac{\ell}{2(\lambda+1)^{j'-j+1}}$ de p . Repetindo recursivamente esse procedimento para todos os filhos de Q de nível pelo menos $j + 3$ que não estão contidos no disco até que todos os filhos de nível pelo menos $j + 3$ tenham sido incluídos em discos. Os discos não se interceptam, pois isso significaria que um centro de um disco, o qual é um ponto de um quadrado de nível $j + 3$ está a uma distância menor que $\frac{\ell}{2(\lambda+1)^2}$ e pelo menos $\frac{\ell}{2(\lambda+1)^{j'-j+1}}$ de outro ponto. Além disso, todos estão centrados em pontos de Q e, portanto, estão contidos num quadrado de lado $(1 + \frac{1}{2(\lambda+1)^2})\ell$. O número máximo de discos em tais condições é dada pela divisão da área do quadrado envoltório pela área dos discos:

$$\frac{\left(1 + \frac{1}{2(\lambda+1)^2}\right)^2 \ell^2}{\pi \left(\frac{\ell}{4(\lambda+1)^2}\right)^2} \leq \frac{4}{\pi} \left(2(\lambda + 1)^2 + 1\right)^2.$$

Portanto, como cada disco contém, no máximo, 4 filhos de Q de nível pelo menos $j + 3$, o número máximo de tais filhos é $\frac{16}{\pi}(2(\lambda + 1)^2 + 1)^2$. \square

O Algoritmo 1 executa λ^2 iterações. Em cada iteração, no máximo $O(n^2)$ quadrados relevantes são processados. Utilizando o Lema 2.1.1, observa-se que o tempo necessário para computar a tabela de entradas para os quadrados relevantes é no máximo $n^{O(\lambda^4 + \xi)}$.

Como o número de quadrados relevantes que não possuem pais é no máximo 4, a última etapa de cada iteração é executada em tempo polinomial. O tempo total de execução do algoritmo é $n^{O(\lambda^4 + \xi)}$.

Seja $(\mathcal{L}, F, 2, \alpha)$ uma instância do problema MSCP. Considere todas as soluções em que qualquer quadrado relevante de lado ℓ contenha no máximo ξ discos de raio no mínimo $\frac{\ell}{2(\lambda+1)^2}$, os quais podem cobrir todos os pontos de \mathcal{L} que estejam no quadrado. Tais soluções são denominadas ξ -soluções para a instância $(\mathcal{L}, F, 2, \alpha)$. Para qualquer j -square relevante definido pela subdivisão do plano de acordo com o par (s, t) , uma ξ -solução para $(\mathcal{L}, F, 2, \alpha)$ contém no máximo ξ discos de nível no máximo j cobrindo todos os pontos de \mathcal{L} no quadrado. A prova da eficiência do algoritmo se baseia na comparação entre o custo da solução obtida por meio do algoritmo com o custo da melhor ξ -solução.

No Lema 2.1.2 é demonstrado que o custo da solução obtida por meio do algoritmo numa iteração associada ao par (s, t) é limitada superiormente por uma quantidade definida como uma função do custo da melhor ξ -solução e pela subdivisão do plano definida por s e t . No Lema 2.1.3 é demonstrado que existem valores de s e t para os quais o custo da melhor solução computada pelo algoritmo não é muito maior que o custo da melhor ξ -solução.

Seja C^* a melhor ξ -solução de uma instância $(\mathcal{L}, F, 2, \alpha)$. Para qualquer j -square Q , seja $C^*(Q)$ os discos de nível j em C^* que intersectam Q .

Lema 2.1.2. *Sejam $s, t \in \{0, \dots, \lambda - 1\}$ e $\mathcal{Q}(s, t)$ o conjunto de quadrados relevantes definidos por s, t e \mathcal{L} . Numa iteração associada com (s, t) , o algoritmo computa uma k -cobertura $A(s, t)$ de \mathcal{L} de custo $\omega(A(s, t)) \leq \sum_{Q \in \mathcal{Q}(s, t)} \omega(C^*(Q))m$.*

Como C^* é uma ξ -solução, cada ponto pode ser atribuído a exatamente um disco, de forma que o número de discos que intersectam Q e para os quais foi atribuído um ponto seja no máximo ξ . Os discos para os quais foram atribuídos um ponto de Q são ditos associados a Q .

Para um j -square relevante Q , seja C^Q o conjunto de discos em C^* associados a Q . Define-se $C^Q_{<j}$, $C^Q_{\leq j}$ e $C^Q_{=j}$ de maneira análoga às projeções em um pseudoagrupamento. Afirma-se que, após a tabela $Table_Q$ ser computada:

$$\omega(Table_Q(C^Q_{<j}, |C^Q|)) \leq \sum_{Q' \prec Q} \omega(C^*(Q')), \quad (2.1)$$

onde $Q' \prec Q$ denota que Q' é um quadrado relevante contido em Q .

A prova desta afirmação é dada por indução na ordem em que os quadrados relevantes são processados na iteração. Quando Q não tem filhos, a afirmação é trivialmente verdadeira, pois a entrada $\omega(Table_Q(C^Q_{<j}, |C^Q|))$ é vazia, tendo custo zero. Da mesma forma o custo $\sum_{Q' \prec Q} \omega(C^*(Q'))$ é zero, pois não há filhos de Q .

Assuma que no momento do processamento de Q , a Equação 2.1 é verdadeira para todos os quadrados processados antes de Q . Em uma das iterações do Algoritmo 1, $Q = C_{\leq j}^Q$ (e conseqüentemente, $J = C_{=j}^Q$). Ou seja, Q é exatamente o conjunto de discos associados a Q da ξ -solução ótima C^* de nível no máximo j . Nessa iteração, considere a combinação (i_1, i_2, \dots, i_t) tal que $P'(Q_y) = C_{\leq j}^{Q_y}$ e $i_y = |C^{Q_y}|$ para todo $1 \leq y \leq t$. Observa-se que para cada j' -square filho de Q , $C_{\leq j}^{Q'} = C_{< j'}^{Q'}$. Além disso, $C = j^Q \subseteq C^*(Q)$. Portanto, o conjunto J' de custo mínimo tal que $P \cup J'$ é um pseudoagrupamento de Q e $|P \cup J'| = |C^Q|$ tem custo no máximo:

$$\sum_{Q' \text{ filhode } Q} \omega(\text{Table}_{Q'}(C_{\leq j}^{Q'}, |C^{Q'}|)) + \omega C^*(Q) \leq \sum_{Q' < Q} \omega(C^*(Q')),$$

e, dessa maneira, 2.1 é válido também para Q .

Ao final, utilizando o mesmo argumento e pelo fato de que $C_{< j}^Q = \emptyset$ para todo quadrado relevante Q que não tem pais, demonstra-se que:

$$\omega(A(r, s)) \leq \sum_{Q \in \mathcal{Q}_0(r, s)} \omega(\text{Table}_Q(\emptyset, |C^Q|)) \leq \sum_{Q \in \mathcal{Q}(r, s)} \omega(C^*(Q)), \quad (2.2)$$

onde $\mathcal{Q}_0(r, s)$ denota o conjunto dos quadrados relevantes que não têm pais. \square

Lema 2.1.3. *Existem $s, t \in \{0, 1, \dots, \lambda-1\}$ tais que $\sum_{Q \in \mathcal{Q}(s, t)} \omega(C^*(Q)) \leq \left(1 + \frac{6}{\lambda}\right) \omega(C^*)$.*

Uma afirmação semelhante ao Lema 2.1.3 é demonstrado em [15]. Dados s e t , seja \mathcal{C}_j^s os discos de nível j pertencentes a C^* que interceptam uma linha vertical ativa e \mathcal{C}_j^t o conjunto de discos de nível j pertencentes a C^* que interceptam uma linha horizontal ativa t . Define-se $\mathcal{C}^s = \bigcup_j \mathcal{C}_j^s$ e $\mathcal{C}^t = \bigcup_j \mathcal{C}_j^t$. Pela construção da grade, se um disco intercepta uma linha vertical ativa s , o mesmo não interceptará nenhuma linha ativa para $s' \neq s$. Da mesma forma, se um disco intercepta uma linha horizontal ativa t , o mesmo não interceptará nenhuma linha ativa para $t' \neq t$. Assim, os conjuntos \mathcal{C}^s são disjuntos para $0 \leq s < \lambda$, bem como \mathcal{C}^t são disjuntos para $0 \leq t < \lambda$. Portanto existe pelo menos um s para o qual $\omega(\mathcal{C}^s) \leq \frac{\omega(C^*)}{\lambda}$ e um t para o qual $\omega(\mathcal{C}^t) \leq \frac{\omega(C^*)}{\lambda}$. Dessa forma, $\omega(\mathcal{C}^s \cup \mathcal{C}^t) \leq \frac{2}{\lambda} \omega(C^*)$ para s e t . Como um disco pode ser associado a no máximo 4 j -squares, os discos podem ser contados em $\mathcal{C}^s \cup \mathcal{C}^t$ 3 vezes mais que em C^* . Conclui-se que,

$$\sum_{Q \in \mathcal{Q}(s, t)} \omega(C^*(Q)) \leq \left(1 + \frac{6}{\lambda}\right) \omega(C^*). \quad (2.3)$$

\square

Utilizando os Lemas 2.1.2 e 2.1.3, o custo da melhor solução computada por meio do algoritmo após todas iterações foi limitado por uma função em termos da melhor ξ -solução. Nos Lemas 2.1.4 e 2.1.5 demonstra-se que, para qualquer instância $(\mathcal{L}, F, 2, \alpha)$, uma solução ótima, para $\alpha = 1$, ou uma solução aproximada, para $\alpha > 1$, são ξ -soluções.

Lema 2.1.4. *Para qualquer inteiro $\lambda > 1$, existe uma constante $\xi = \xi(\lambda) = O(\lambda^4)$ tal que a seguinte afirmação é válida: para qualquer quadrado Q de lado ℓ , qualquer solução ótima para uma instância $(\mathcal{L}, F, 2, 1)$ do problema MSCP contém no máximo ξ discos de raio pelo menos $\frac{\ell}{2(\lambda+1)^2}$ que intersectam Q .*

Uma versão semelhante do Lema 2.1.4 foi demonstrado em [15]. Tal prova se baseia em propriedades geométricas da construção da grade, principalmente a distância mínima entre as linhas da grade em um dado nível. É feita uma subdivisão da grade, de forma que cada região determinada por tal subdivisão contenha apenas um centro de disco que intersecta o quadrado Q e o número de regiões determinadas por tal subdivisão seja $O(\lambda^4)$.

Lema 2.1.5. *Para quaisquer constantes $\gamma > 0$, $\alpha > 1$ e um inteiro $\lambda > 1$, existe uma constante $\xi = \xi(\lambda, \alpha, \gamma) = O\left(\frac{\alpha^2 \lambda^4}{\gamma^2}\right)$ tal que a seguinte afirmação é verdadeira: qualquer instância $(\mathcal{L}, F, 2, \alpha)$ do problema MSCP possui uma solução $(1+\gamma)^\alpha$ -aproximada tal que, para qualquer quadrado Q de lado ℓ , contém um subconjunto de no máximo ξ discos de raio pelo menos $\frac{\ell}{2(\lambda+1)^2}$ que contém todos os pontos de Q .*

A prova do Lema 2.1.5 é dada por contradição. Seja um quadrado Q de lado ℓ , A^* uma solução ótima para a instância do $(\mathcal{L}, F, 2, \alpha)$ do problema e ψ^* a função que atribui a cada ponto de \mathcal{L} um disco de A^* que contém tal ponto. Obtém-se uma solução $(1+\gamma)^\alpha$ -aproximada A aumentando cada um dos discos de A^* em um fator de $(1+\gamma)$. Seja ψ a função que atribui a cada um dos pontos de \mathcal{L} o disco de A de menor raio que o cobre. Demonstra-se que o número de discos de raio pelo menos $\frac{\ell}{2(\lambda+1)^2}$ que são atribuídos a pontos pertencentes a qualquer quadrado de lado ℓ pela função ψ é no máximo

$$\xi(\lambda, \alpha, \gamma) = \left(\left(6\sqrt{2} + \frac{4\sqrt{2}}{\gamma} \right) \frac{\alpha(1+\gamma)(\lambda+1)^2}{\ln 2} + 1 \right)^2 + 1 = O\left(\frac{\alpha^2 \lambda^4}{\gamma^2}\right). \quad (2.4)$$

Ou seja, tal solução aproximada A é uma xi -solução para a instância dada do problema.

Seja Q um quadrado de lado ℓ . Define-se os conjuntos X_1 e X_2 de pontos de Q atribuídos por ψ^* a discos de A^* de raios respectivamente menores que $\ell\sqrt{2}/\gamma$ e pelo menos $\ell\sqrt{2}/\gamma$. Todos os pontos de X_1 são atribuídos por ψ a discos de A de raio menor que $\ell\sqrt{2}(1+1/\gamma)$. Os raios dos discos de A^* para os quais os pontos de X_2 foram atribuídos por ψ^* são aumentados em pelo menos $\ell\sqrt{2}$ e, portanto, os discos resultantes em A cobrem o quadrado Q inteiro. Um exemplo de um disco de A^* e o disco correspondente em A é apresentado na Figura 2.5. Dentre tais discos, seja d o disco de menor raio. Os pontos de X_2 serão atribuídos a d ou a um disco de raio menor que $\ell\sqrt{2}(1+1/\gamma)$.

Agora assume-se que existem mais de $\xi(\lambda, \alpha, \gamma)$ discos de A são atribuídos a pontos de Q por ψ . Isso significa que mais de $\xi(\lambda, \alpha, \gamma) - 1$ discos de raio maior que $\frac{\ell}{2(\lambda+1)^2}$ e no máximo $\ell\sqrt{2}/\gamma$ têm seus centros a uma distância máxima de $\left(\frac{3}{\sqrt{2}} + \frac{\sqrt{2}}{\gamma}\right)\ell$ do centro

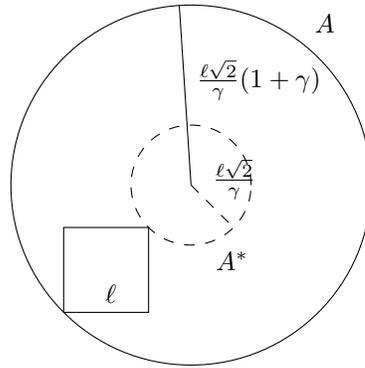


Figura 2.5: Um disco de A (linha contínua) e o disco de A^* do qual se originou. Se um ponto de X_2 foi atribuído ao disco original por ψ^* , então o disco aumentado cobre o quadrado inteiro.

do quadrado Q , do contrário tais discos não cobririam nenhum ponto de Q . Contraíndo tais discos em torno dos seus centros para se obter discos de raio $r' = \frac{2^{1/\alpha}-1}{4(1+\gamma)(\lambda+1)^2}\ell$, obtém-se o conjunto A' de discos contraídos. É demonstrado em [5] que tais discos são disjuntos. Por sua definição, os discos do conjunto A' estão contidos em um disco D' de raio $\left(\frac{3}{\sqrt{2}} + \frac{\sqrt{2}}{\gamma} + r'\right)\ell$ em torno do centro do quadrado Q . No entanto, prova-se que a área de $\xi(\lambda, \alpha, \gamma) - 1$ discos de raio r' é maior que a área do disco D' , contrariando o fato de que os discos de A' estão contidos em D' . Portanto o número de discos de A de raio pelo menos $\frac{\ell}{2(\lambda+1)^2}$ que são atribuídos a pontos de Q não pode exceder $\xi(\lambda, \alpha, \gamma)$. \square

Combinando as provas dos Lemas 2.1.2, 2.1.3, 2.1.4 e 2.1.5, e escolhendo os parâmetros λ e ξ em termos de ε apropriadamente, obtém-se os seguintes teoremas:

Teorema 2.1.1. *Existe um algoritmo para o problema MSCP que computa, para cada instância $(\mathcal{L}, F, 2, 1)$ do problema, uma solução $(1+\varepsilon)$ -aproximada em tempo $n^{O(1/\varepsilon^4)}$ para qualquer $\varepsilon > 0$.*

Teorema 2.1.2. *Existe um algoritmo para o problema MSCP que computa, para cada instância $(\mathcal{L}, F, 2, \alpha)$ do problema, uma solução $(1+\varepsilon)$ -aproximada em tempo $n^{O(\alpha^4/\varepsilon^6)}$ para qualquer $\varepsilon > 0$.*

2.2 Um Algoritmo exato para a cobertura de pontos no plano

Gibson et al. consideram em [12] o problema da k -cobertura de pontos no plano Euclidiano minimizando a soma dos raios dos discos que compõem a cobertura, ou seja, consideram

o custo dos discos dados por uma função linear do raio dos mesmos.

Sejam \mathcal{L} um conjunto de pontos no plano Euclidiano e um inteiro k . Um disco centrado num ponto $c \in \mathcal{L}$ de raio r , é denotado por $D(c, r)$ e definido como o conjunto $\{p \in \mathbb{R}^2 \mid \delta(c, p) \leq r\}$, onde $\delta(c, p)$ denota a distância Euclidiana entre o centro c e um ponto p . Diz-se que um ponto é coberto por um disco D caso esteja contido em D . Analogamente, um conjunto de discos \mathcal{S} cobre um conjunto de pontos \mathcal{P} caso a união dos discos $D_i \in \mathcal{S}$ contenha todos os pontos de \mathcal{P} . O custo de um conjunto de discos é dado pela soma dos raios dos discos do conjunto. O problema consiste em se determinar um conjunto \mathcal{S} de k discos centrados em pontos $c_i \in \mathcal{L}$ tal que \mathcal{S} cubra \mathcal{L} e tenha custo mínimo.

Em [12] é apresentado um algoritmo exato de tempo polinomial sob a suposição de que os custos de duas soluções candidatas podem ser comparadas em tempo polinomial. No problema de k -cobertura de pontos, uma solução ótima é composta por um conjunto de até k discos centrados sobre pontos da entrada e cujos raios são a distância entre dois pontos da entrada. Se os pontos da entrada tiverem coordenadas inteiras, o custo total da solução será dada pela soma de raízes quadradas de inteiros. Assume-se que é possível comparar duas soluções candidatas em tais circunstâncias em tempo polinomial.

Nota-se que em variantes do problema em que a métrica da distância é L_1 ou L_∞ em vez da métrica Euclidiana, é possível de fato comparar duas soluções candidatas em tempo polinomial sem assumir nenhum pressuposto sobre o modelo computacional. Para a métrica Euclidiana, no entanto, não é determinado se duas somas de raízes quadradas de inteiros podem ser comparadas em tempo polinomial, sendo este um problema em aberto.

Os resultados apresentados podem ser expandidos para o problema no plano, considerado por Lev-Tov e Peleg em [15]. Também é possível estender os resultados para qualquer dimensão fixa.

2.2.1 Estrutura da solução ótima e lemas iniciais

Os resultados apresentados em [12] se baseiam na observação da estrutura de soluções ótimas para o problema. No caso planar, observa-se que sempre existe uma linha vertical que intercepta $O(1)$ discos da solução ótima e que separa os pontos de entrada de uma maneira que pode ser explorada, permitindo que uma solução ótima seja encontrada por programação dinâmica.

Apesar dos resultados serem extensíveis para qualquer dimensão fixa, toda a análise é apresentada focando-se o caso planar, em que a entrada é dada por um conjunto de pontos $\mathcal{L} \in \mathbb{R}^2$ e um inteiro k . Define-se o comprimento de um retângulo como o comprimento do seu lado mais longo. Se o retângulo for um quadrado, tal lado é definido arbitrariamente. De forma análoga, a largura de um retângulo é definida como o lado mais curto do retângulo. Um separador para um retângulo G é qualquer linha s que intercepta G , é

perpendicular ao seu comprimento e a distância entre s e qualquer um dos lados mais curtos do retângulo é pelo menos um terço do comprimento de G . A Figura 2.6 apresenta um exemplo de um retângulo e um separador para o mesmo.

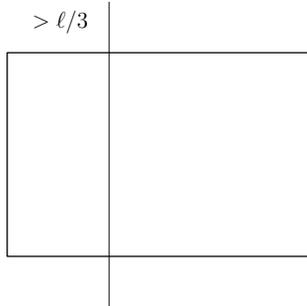


Figura 2.6: Um separador para um retângulo está distante pelo menos $1/3$ do comprimento de qualquer um dos lados mais curtos.

Lema 2.2.1. *Considere uma k -cobertura ótima \mathcal{S}^* para algum conjunto $\mathcal{P} \subseteq \mathcal{L}$ de pontos contidos num retângulo G . O retângulo G possui um separador que intercepta no máximo 12 discos em \mathcal{S}^* .*

Seja l o comprimento do retângulo G . Escolhendo um separador para G aleatoriamente, a probabilidade de tal separador interceptar um disco $D(c, r) \in \mathcal{S}^*$ é no máximo $\frac{2r}{l/3}$. Portanto o número esperado de discos de \mathcal{S}^* interceptados por um separador é no máximo $\frac{6}{l} \sum_{D_i(c_i, r_i) \in \mathcal{S}^*} r_i = \frac{6\omega(\mathcal{S}^*)}{l}$, onde $\omega(\mathcal{S}^*)$ denota o custo de \mathcal{S}^* . O custo $\omega(\mathcal{S}^*)$, dado pela soma dos raios dos discos de \mathcal{S}^* , é no máximo $2l$. Isso se deve ao fato de que um disco de raio $2l$ centrado em qualquer ponto de \mathcal{P} cobre todos os pontos de \mathcal{P} . Portanto o número esperado de discos em \mathcal{S}^* interceptados por um separador escolhido aleatoriamente é 12.

O Lema 2.2.2 apresentado a seguir estabelece uma propriedade estrutural de uma solução ótima no caso planar, semelhante a observações feitas por Lev-Tov e Peleg em [15].

Lema 2.2.2. *Seja \mathcal{S}^* uma k -cobertura ótima para $\mathcal{L} \subseteq \mathbb{R}^2$. Seja G um retângulo de comprimento $a > 0$. O número de discos em \mathcal{S}^* de raio pelo menos a que intercepta G é no máximo 40.*

Seja $\mathcal{S}' \subseteq \mathcal{S}^*$ os discos de \mathcal{S}^* que possuem raio pelo menos a . O retângulo G pode ser envolvido por um disco B de raio a .

A propriedade básica de uma solução ótima \mathcal{S}^* é a de que nenhum disco contido em \mathcal{S}^* possui em seu interior o centro de outro disco contido em \mathcal{S}^* . Dessa forma, os centros

dos discos de \mathcal{S}' têm uma distância de pelo menos a entre si. Considere um disco B' concêntrico a B de raio $2a$. A Figura 2.7 ilustra um retângulo de raio a e os discos B e B'

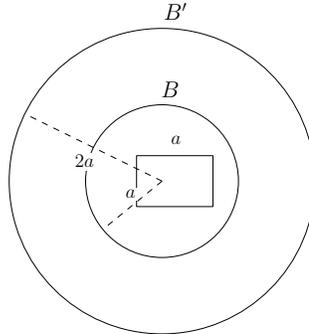


Figura 2.7: Um retângulo de comprimento a pode ser envolvido por um disco B de raio a . B' é concêntrico a B e possui raio $2a$.

Considere os centros dos discos de \mathcal{S}' que estão contidos em B' . Considere bolas de raio $\frac{a}{2}$ em cada um desses centros.

Os interiores de tais bolas são disjuntos e estão contidos numa bola de raio $\frac{5a}{2}$. O número dessas bolas é no máximo 25, pois a área de um disco de raio $\frac{5a}{2}$ é 25 vezes maior que a área de um disco de raio $\frac{a}{2}$, ou seja, $\frac{\pi(5a/2)^2}{\pi(a/2)^2} = 25$. Portanto, no máximo 25 discos de \mathcal{S}' possuem centros contidos em B' . Na Figura 2.8 são apresentadas as dimensões de dos discos B' e os de raio $\frac{5a}{2}$ e $\frac{a}{2}$.

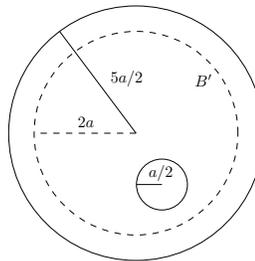


Figura 2.8: A área de um disco de raio $\frac{5a}{2}$ é 25 vezes maior que um disco de raio $\frac{a}{2}$.

Resta limitar o número de discos em \mathcal{S}' cujos centros estão no exterior de B' e que também interceptam B . Particiona-se o plano radialmente em 15 setores de ângulo $2\pi/15$ a partir do centro de B . Sejam dois pontos p e q no exterior de B' , mas contidos no mesmo setor. Assuma, sem perda de generalidade, que p está mais próximo de B do que q . É fácil verificar que a distância $\delta(p, q)$ é menor que a distância entre q e B . Portanto,

se um disco centrado em q intercepta B , então tal disco deve conter p . Como nenhum disco de \mathcal{S}^* contém o centro de outro disco de \mathcal{S}^* , para cada setor deve existir apenas um disco que intercepta B . A Figura 2.9 ilustra um desses setores. Portanto o número de discos de \mathcal{S}^* que interceptam B e possuem raio no mínimo a é no máximo $25 + 15 = 40$.

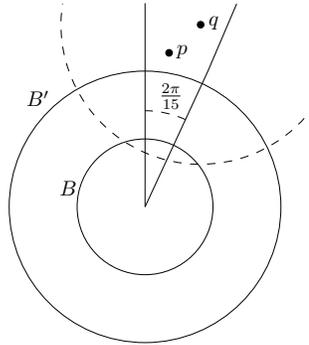


Figura 2.9: Em cada setor de ângulo $2\pi/15$, \mathcal{S}' deve conter apenas um disco centrado no exterior de B' que intercepta B .

2.2.2 Descrição do algoritmo de Gibson et al.

No caso planar, o algoritmo recebe um conjunto \mathcal{L} de n pontos no plano e um inteiro k , computando uma k -cobertura para os pontos de \mathcal{L} assumindo que o custo de duas soluções podem ser computadas em tempo polinomial. Assume-se que $k < n$, pois em caso contrário a cobertura seria dada trivialmente por discos de raio nulo posicionados sobre os n pontos de \mathcal{L} . Seja \mathcal{D} o conjunto de discos centrados em algum ponto $p \in \mathcal{L}$ cujo raio é $\delta(p, q)$ para algum ponto $q \in \mathcal{L}$. Nota-se que \mathcal{D} inclui também os discos de raio 0 e, portanto, $|\mathcal{D}| = n^2$. Observa-se também que existe uma solução ótima para a k -cobertura de \mathcal{L} cujos discos estão contidos em \mathcal{D} .

Uma linha vertical ou horizontal é denominada *crítica*, caso passe por um ponto de \mathcal{L} ou tangencie um disco de \mathcal{D} . Note que se duas linhas l e l' não forem críticas e não houver nenhuma linha crítica entre as duas, então l e l' interceptam o mesmo conjunto de discos em \mathcal{D} . Assim, define-se um conjunto de $\Theta(n^2)$ linhas verticais *canônicas* que inclui exatamente uma linha arbitrária posicionada entre todos os pares de linhas verticais críticas consecutivas, uma linha vertical à esquerda da linha crítica mais a esquerda e uma linha vertical à direita da linha crítica mais a direita. É definido um conjunto de $\Theta(n^2)$ linhas horizontais canônicas de maneira análoga. A Figura 2.10 apresenta um exemplo de linhas críticas e canônicas.

Um retângulo é dito *balanceado* caso sua largura seja pelo menos um terço do seu comprimento. O procedimento $\text{COMPRESS}(G)$ recebe um retângulo balanceado G que

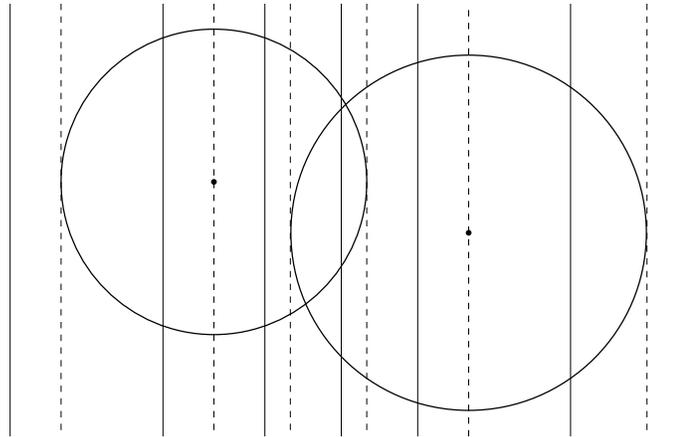


Figura 2.10: Linhas críticas (pontilhadas) e canônicas (contínuas).

contém pelo menos dois pontos de \mathcal{L} e retorna um retângulo balanceado G' tal que: G' esteja contido em G , G' contém $\mathcal{L} \cap G$ e, para qualquer separador de G' , existem pontos de $\mathcal{L} \cap G$ em ambos os semi-planos limitados por tal separador.

Assume-se, sem perda de generalidade, que os separadores são verticais, ou seja, o comprimento do retângulo é horizontal. O procedimento checka inicialmente se há um ponto de $\mathcal{L} \cap G$ à esquerda do separador de G mais à esquerda e se há um ponto à direita do separador de G mais à direita. Em caso positivo, é retornado $G' = G$.

Em caso contrário, seja G' o retângulo minimal que envolve $\mathcal{L} \cap G$. Seja l' o seu comprimento e w' sua largura. Sejam l e w respectivamente o comprimento e a largura de G . Se $w' \geq l'/3$, o retângulo já está balanceado e, portanto, G' é retornado.

Caso $w' \leq l'/3$, é necessário alargar G' , mas mantendo-o contido em G . Existem dois casos a considerar, dependendo da posição entre o comprimento de G e G' .

Caso o comprimento de G e G' sejam paralelos, $l' \leq 2l/3$. A largura de G' deve ser expandida para $l'/3$. Como $l'/3 \leq 2l/9 \leq l/3 \leq w$, existe espaço para tal expansão.

Caso o comprimento de G' seja paralelo à largura de G , $l' \leq w$. A largura de G' deve ser expandida para $l'/3$. Como $l'/3 \leq w/3 \leq l/3 \leq l$, há também espaço para tal expansão.

Portanto, sempre é possível obter um retângulo balanceado.

Será descrito o procedimento $\text{DC}(G, \kappa, T)$ que recebe um retângulo balanceado G , um inteiro $\kappa \geq 0$ e um subconjunto $T \subseteq \mathcal{D}$ e retorna uma κ -cobertura do conjunto

$$\mathcal{P} = \{p \in \mathcal{L} \cap G \mid p \text{ não é coberto por } T\} \quad (2.5)$$

Será demonstrado que o resultado de uma chamada ao procedimento $\text{DC}(S, k, \emptyset)$ computa uma k -cobertura ótima para o conjunto \mathcal{L} para qualquer retângulo balanceado S

que contenha \mathcal{L} .

O resultado de uma chamada a $\text{DC}(G, \kappa, T)$ é armazenado em uma tabela indexada por $\mathcal{L} \cap G, \kappa, T$. Ou seja, a entrada da tabela $\text{Table}(\mathcal{L} \cap G, \kappa, T)$ armazena uma κ -cobertura para o conjunto \mathcal{P} definido acima. É útil definir para a descrição do algoritmo um disco especial I cujo custo é ∞ . Se uma entrada $\text{Table}(\mathcal{L} \cap G, \kappa, T)$ contiver um conjunto que contém I , significa que o algoritmo determinou que não existe uma κ -cobertura para \mathcal{P} . A tabela conterá tal entrada apenas caso $\kappa = 0$.

O Algoritmo 2 contém uma descrição do procedimento $\text{DC}(G, \kappa, T)$.

Observa-se que particionar com um separador um retângulo balanceado resulta em dois retângulos balanceados. Portanto, G_1 e G_2 são balanceados, dado que G' é balanceado.

Nenhum separador considerado no algoritmo passa por um ponto em \mathcal{L} , tornando desnecessário analisar tal caso.

2.2.3 Análise do algoritmo

Em [12] é apresentado o limitante polinomial para o tempo total de execução do algoritmo $\text{DC}(S, k, \emptyset)$, em que S é um retângulo balanceado que contém \mathcal{L} . A seguir serão apresentados os lemas demonstrados em [12], juntamente com comentários sobre a idéia geral utilizada na demonstração dos mesmos.

Note que cada entrada da tabela $\text{Table}(\mathcal{L} \cap G, \kappa, T)$ é indexada por um conjunto de pontos $\mathcal{L} \cap G$ para algum retângulo balanceado G , um $\kappa \leq k$ e um conjunto $T \subseteq \mathcal{D}$ tal que $|T| \leq \beta = 424$. O número de conjuntos $\mathcal{L} \cap G$ é $O(n^4)$, dado pelo número de faces definidas pela combinação de linhas canônicas horizontais e verticais, o número de possíveis $\kappa \leq k$ é $O(n)$ e o número de conjuntos T é $O(n^{2\beta}) = O(n^{848})$. Como existe uma entrada para cada tupla $(\mathcal{L} \cap G, \kappa, T)$, o número total de entradas da tabela é então limitado por $O(n^{853})$. Tal número será utilizado para limitar o número total de chamadas ao procedimento $\text{DC}(G, \kappa, T)$.

O número de chamadas da forma $\text{DC}(G, \kappa, T)$ que criam a entrada correspondente na tabela é também limitado por $O(n^{853})$, já que cada entrada é criada apenas uma vez. As chamadas ao algoritmo que não criam uma entrada na tabela, não realizam chamadas recursivas, levando tempo $O(1)$ em sua execução.

Cada instância de $\text{DC}()$ realiza $O(n^{28})$ chamadas recursivas, já que existem $O(n^2)$ escolhas para o separador l , $O(n^{24})$ escolhas para \mathcal{D}^0 e $O(n)$ escolhas para κ_1 e κ_2 . O número total de chamadas é limitado por $O(n^{881})$.

O limitante para o tempo total de execução é dado por $O(n^{881} \Delta(n))$, onde $\Delta(n)$ é o limitante superior para o tempo necessário para comparar os custos de dois subconjuntos de \mathcal{D} , cada um de tamanho no máximo n .

A prova da corretude do algoritmo, que demonstra que $\text{DC}()$ retorna uma solução ótima para o problema, é mais complexa, principalmente devido ao passo que exige que na instância $\text{DC}(G, \kappa, T)$, $|T|$ seja limitado por β .

Lema 2.2.3. *Seja $\text{DC}(G_t, \cdot, \cdot)$ uma chamada recursiva encadeada em uma sequência de chamadas recursivas $\text{DC}(G_1, \cdot, \cdot)$, $\text{DC}(G_2, \cdot, \cdot)$, \dots , $\text{DC}(G_{t-1}, \cdot, \cdot)$, em que $G_1 = S$. Suponha que $|G_t \cap \mathcal{L}| \geq 2$. Seja G'_j o resultado de uma chamada ao procedimento $\text{COMPRESS}(G_j)$. Seja $l(G'_j)$ o separador escolhido para G'_j para cada $1 \leq j \leq t-1$, de forma que G_{j+1} seja um dos dois retângulos nos quais $l(G'_j)$ particiona G'_j . Além disso, seja $l(G_t)$ qualquer separador de G'_t e G_1 um dos retângulos em que $l(G'_t)$ particiona G'_t . Denota-se por a o*

comprimento de G_1 . A distância horizontal entre dois separadores distintos $l(G'_i)$ e $l(G'_j)$ é pelo menos $a/3$ e a distância vertical entre dois separadores horizontais distintos $l(G'_i)$ e $l(G'_j)$ é pelo menos $a/3$.

Como $G_1 \subseteq G'_t \subseteq G_t \subseteq G'_{t-1} \subseteq \dots \subseteq G'_1 \subseteq G_1$, o comprimento de qualquer G'_i é pelo menos a . Seja G'_i e G'_j tais que $i < j$ e $l(G'_i)$ e $l(G'_j)$ são verticais. Note que G'_j está contido em um dos dois retângulos nos quais $l(G'_i)$ particiona G'_i . Portanto, $l(G'_i)$ está à esquerda ou à direita de G'_j . Já $l(G'_j)$ se encontra entre os dois lados verticais de G'_j a uma distância de pelo menos $1/3$ do comprimento de G'_j de cada lado. Como o comprimento de G'_j é pelo menos a , segue que a distância horizontal entre $l(G'_i)$ e $l(G'_j)$ é pelo menos $a/3$. A Figura 2.11 ilustra os separadores e a distância entre eles.

O caso em que $l(G'_i)$ e $l(G'_j)$ são horizontais é analisado similarmente.

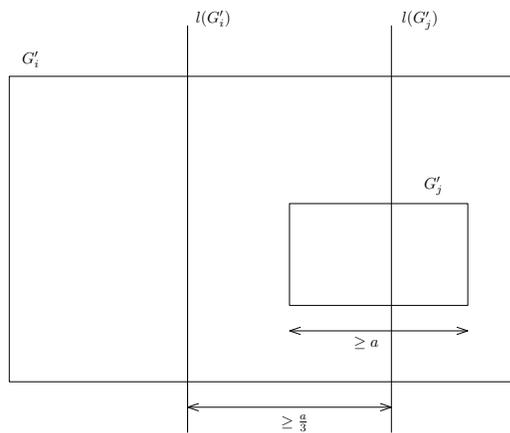


Figura 2.11: Ilustração para o Lema 2.2.3.

A prova da corretude do algoritmo provém do seguinte Lema. Seja fixada uma k -cobertura ótima $\mathcal{S}^* \subseteq \mathcal{D}$ de \mathcal{L} :

Lema 2.2.4. *Suponha que a instância recursiva $DC(G, \kappa, T)$ é chamada (em alguma profundidade da recursão) pela instância de nível mais alto $DC(S, k, \emptyset)$. Suponha também que $T \subseteq \mathcal{S}^*$ e T contém todo disco de \mathcal{S}^* que contenha um ponto em $\mathcal{L} \cap G$ e um ponto em $\mathcal{L} \setminus G$. (T possivelmente contém outros discos de \mathcal{S}^*). Seja $\mathcal{P} = \{q \in \mathcal{L} \cap G \mid q \text{ não é coberto por } T\}$ e seja \mathcal{S}' o conjunto de discos em \mathcal{S}^* que contém pontos em \mathcal{P} . Além disso, suponha que $\kappa \geq |\mathcal{S}'|$. Então, após o retorno da chamada a $DC(G, \kappa, T)$, $Table(\mathcal{L} \cap G, \kappa, T)$ conterá uma κ -cobertura de \mathcal{P} cujo custo é no máximo $\omega(\mathcal{S}')$.*

A prova é dada por indução em $|\mathcal{L} \cap G|$. Pode-se assumir que $Table(\mathcal{L} \cap G, \kappa, T)$ ainda não foi criado quando $DC(G, \kappa, T)$ foi chamado. Em caso contrário, essa entrada da tabela

foi criada pela chamada a $\text{DC}(\hat{G}, \kappa, T)$ para um retângulo \hat{G} tal que $\hat{G} \cap \mathcal{L} = G \cap \mathcal{L}$. Verifica-se que as suposições feitas sobre T e κ são válidas para \hat{G} no lugar de G e também que \mathcal{P} e \mathcal{S}^* são iguais nos dois casos. Será apresentado a seguir os argumentos que permitem concluir que, após uma chamada a $\text{DC}(G, \kappa, T)$, $\text{Table}(\mathcal{L} \cap G = \mathcal{L} \cap \hat{G}, \kappa, T)$ contém uma κ -cobertura para \mathcal{P} cujo custo é no máximo $\omega(\mathcal{S}')$. As chamadas subsequentes a $\text{DC}(G, \kappa, T)$ retornam imediatamente, sem modificar o conteúdo de $\text{Table}(\mathcal{L} \cap G = \mathcal{L} \cap \hat{G}, \kappa, T)$.

Os casos em que a chamada a $\text{DC}(G, \kappa, T)$ retornam nas linhas 9 ou 15 formam o caso base da indução. Note que a chamada não pode retornar pela linha 12, porque se $\mathcal{P} \neq \emptyset$, então $\kappa \geq |\mathcal{S}'|$ deve ser pelo menos 1.

O caso indutivo acontece quando o retorno é feito pela linha 34. Nota-se que $|\mathcal{L} \cap G|, \mathcal{P} \geq 2$ nesse caso. Assume-se, sem perda de generalidade, que os separadores de G' são verticais.

Inicialmente, observa-se que \mathcal{S}' é uma $|\mathcal{S}'|$ -cobertura ótima para \mathcal{P} . Isso se deve ao fato de que os pontos em $(\mathcal{L} \cap G) \setminus \mathcal{P}$ são cobertos por T , sendo que \mathcal{S}' não cobre nenhum ponto de $\mathcal{L} \setminus G$, mas apenas os pontos de \mathcal{P} , atingindo assim a otimalidade. Pelo Lema 2.2.1 conclui-se que um separador l' de G' intercepta no máximo 12 discos em \mathcal{S}' . É fácil verificar que l' pode ser deslocada para uma das linhas de $L(G')$ (computada na linha 20 do algoritmo) sem aumentar o conjunto de discos em \mathcal{D} que ela intercepta. Pode-se considerar, portanto, a escolha de um separador $l \in L(G')$ que intercepta um conjunto $\hat{\mathcal{D}}_0$ que contém no máximo 12 discos de \mathcal{S}' . Considerando o laço mais interno que se inicia na linha 24, onde l é escolhido como descrito. \mathcal{D}_0 é escolhido como $\hat{\mathcal{D}}_0$, κ_1 é escolhido como o número de discos de \mathcal{S}' à esquerda de l e κ_2 é escolhido como o número de discos à direita de l . Sejam \mathcal{S}_1 e \mathcal{S}_2 os conjuntos de discos de \mathcal{S}' à esquerda e à direita de l , respectivamente. Dessa forma, $|\mathcal{S}_1| = \kappa_1$ e $|\mathcal{S}_2| = \kappa_2$.

Sejam G_1, G_2, T_1 e T_2 exatamente como na descrição do Algoritmo 2. Nota-se que $|\mathcal{L} \cap G_1| < |\mathcal{L} \cap G|$ e $|\mathcal{L} \cap G_2| < |\mathcal{L} \cap G|$. Além disso, $T_1 \subseteq \mathcal{S}^*$ e T_1 contém todos os discos de \mathcal{S}^* que contém um ponto de $\mathcal{L} \cap G$ e um ponto de $\mathcal{L} \setminus G$. Observa-se também que \mathcal{S}_1 é o conjunto de pontos de \mathcal{S}^* que contém pontos de $\mathcal{P}_1 = \{q \in \mathcal{L} \cap G_1 \mid q \text{ não é coberto por } T_1\}$. Pode-se assumir a hipótese de indução de que, caso a chamada a $\text{DC}(G_1, \kappa_1, T_1)$ seja feita, após o retorno da chamada, $\text{Table}(\mathcal{L} \cap G_1, \kappa_1, T_1)$ conterà uma κ_1 -cobertura para \mathcal{P}_1 cujo custo é no máximo $\omega(\mathcal{S}_1)$. Da mesma maneira, caso a chamada a $\text{DC}(G_2, \kappa_2, T_2)$ seja feita, $\text{Table}(\mathcal{L} \cap G_2, \kappa_2, T_2)$ conterà uma κ_2 -cobertura para \mathcal{P}_2 cujo custo é no máximo $\omega(\mathcal{S}_2)$ após o retorno da chamada. Será demonstrado que $|T_1|$ e $|T_2|$ são limitados por β e, portanto, essas chamadas são de fato realizadas. Segue-se que, no passo da Linha 32, $\mathcal{D}_0 \cup \text{Table}(\mathcal{L} \cap G_1, \kappa_1, T_1) \cup \text{Table}(\mathcal{L} \cap G_2, \kappa_2, T_2)$ é uma κ -cobertura para \mathcal{P} cujo custo é no máximo $\omega(\mathcal{D}_0) + \omega(\mathcal{S}_1) + \omega(\mathcal{S}_2) = \omega(\mathcal{S}')$. Portanto, na linha 34, $\text{Table}(\mathcal{L} \cap G, \kappa, T)$ recebe uma cobertura para \mathcal{P} de custo no máximo $\omega(\mathcal{S}')$.

Para completar a prova, será demonstrado que $|T_1| \leq \beta = 424$, sendo a prova para T_2

similar.

Suponha que após uma sequência de chamadas recursivas $\text{DC}(G_1, k, \emptyset)$, $\text{DC}(G_2, \cdot, \cdot)$, \dots , $\text{DC}(G_{t-1}, \cdot, \cdot)$, onde $G_1 = S$, tenha surgido a instância (G, κ, T) . Seja $G_t = G$, G'_j o resultado obtido por meio da chamada ao procedimento $\text{COMPRESS}(G_j)$ e $l(G'_j)$ o separador escolhido para G'_j de forma que, para cada $1 \leq j \leq t-1$, G_{j+1} seja um dos retângulos nos quais $l(G'_j)$ particionou G'_j . Finalmente seja $l(G'_t) = l$ e note que $G'_t = G'$. Portanto, G_1 é um dos retângulos nos quais $l(G'_t)$ particionou G'_t . Além disso, seja $\mathcal{D}_0(G_j)$ a escolha correspondente feita na chamada ao procedimento $\text{DC}(G_j, \cdot, \cdot)$ para $1 \leq j \leq t-1$ e $\mathcal{D}_0(G_t)$ o mesmo \mathcal{D}_0 descrito acima. Note que $T_1 \subseteq \bigcup_{j=1}^t \mathcal{D}_0(G_j)$.

Seja \bar{G} um quadrado de lado $5a$ centrado no centro de G_1 , onde a é o comprimento de G_1 . Pelo Lema 2.2.3, o número de j para os quais $l(G'_j)$ intercepta \bar{G} é limitado por uma constante. Em particular, o número de $l(G'_j)$ verticais é no máximo 16, pois estão distantes pelo menos $a/3$ entre si. Tal observação é válida também para os $l(G'_j)$ horizontais. Portanto a constante pode ser escolhida como 32. Segue que o número de discos em T_1 que pertencem a $\mathcal{D}_0(G_j)$ para tais j é limitado por $32 \times 32 = 384$, já que $|\mathcal{D}_0(G_j)| \leq 12$.

Considerando os discos em T_1 que pertencem a $\mathcal{D}_0(G_j)$ para j tal que $l(G'_j)$ não intercepta \bar{G} . Tais discos interceptam $l(G'_j)$ bem como G_1 , portanto devem possuir raios de pelo menos a . Segue do Lema 2.2.2 que o número de tais discos é no máximo 40. Portanto $|T_1| \leq \beta = 424$.

É imediato do Lema 2.2.4 que, após a chamada ao procedimento $\text{DC}(S, k, \emptyset)$, a entrada $\text{Table}(\mathcal{L}, k, \emptyset)$ conterá uma k -cobertura para \mathcal{L} cujo custo é no máximo $\omega(\mathcal{S}^*)$, o qual é uma k -cobertura ótima.

Teorema 2.2.1. *Existe um algoritmo que, dado um conjunto \mathcal{L} de n pontos no plano e um inteiro $k \geq 1$, executa em tempo $O(n^{881} \cdot \Delta(n))$ e retorna uma k -cobertura ótima para os pontos de \mathcal{L} . $\Delta(n) \geq 1$ denota o limitante superior para o tempo necessário para comparar os custos de dois subconjuntos de \mathcal{D} , cada um de tamanho no máximo n , onde \mathcal{D} é o conjunto de todos os n^2 discos centrados em um ponto $p \in \mathcal{L}$ e cujo raio é $\delta(p, q)$, para um ponto $q \in \mathcal{L}$.*

No próximo capítulo serão apresentados algoritmos para a cobertura de um segmento de reta no plano, apesar dos algoritmos apresentados neste capítulo serem utilizados apenas no Capítulo 4. A escolha da ordem em que os problemas são apresentados se deve à complexidade crescente dos objetos a serem cobertos. Assim, tratamos do caso de cobertura de pontos neste capítulo; de segmentos de reta no Capítulo 3; e de objetos geométricos descritos por um sistema de inequações polinomiais no Capítulo 4.

Capítulo 3

Cobertura de Segmento

Imagine uma situação em que se deseja posicionar sensores para monitorar um trecho de um rio, de forma que todo o trecho esteja no interior da área de alcance dos sensores. Em outra situação, poderia ser necessário distribuir pontos de luz para iluminar uma estrada. As duas situações hipotéticas podem ser modeladas como um problema de cobertura de um segmento de reta por discos. No primeiro caso, os sensores poderiam ser modelados por discos cujos raios são proporcionais ao alcance e o trecho do rio a ser monitorado poderia ser modelado como um segmento de reta. No segundo caso, os pontos de luz poderiam ser modelados por discos que representariam a região iluminada pelo ponto de luz e o trecho da estrada a ser iluminado poderia ser modelado como um segmento de reta.

De maneira mais formal, seja l um segmento de reta de comprimento ℓ , C um conjunto de n pontos nos quais podem ser posicionados os centros dos discos e, para cada centro $c_i \in C$, uma função ω_i que determina o custo de um disco centrado em c_i . Normalmente as funções de custo ω são proporcionais aos raios dos discos. O objetivo é encontrar uma lista $S = (r_1, \dots, r_n)$ de valores para os raios dos discos centrados em $c_i \in C$ de forma que o segmento l seja coberto com custo mínimo, dado pela soma dos custos dos discos definidos por S . Não foi estabelecida, até o momento da escrita desta dissertação, qual a complexidade de tempo de tal problema.

Na Seção 3.1 será apresentado um algoritmo polinomial exato para a cobertura de um segmento de reta por discos centrados sobre o segmento, proposto em [16], e na Seção 3.2 será apresentado um FPTAS que desenvolvemos pela extensão do algoritmo de [16] para uma versão mais geral, no qual os centros dos discos não precisam ser necessariamente posicionados sobre o segmento.

3.1 Um Algoritmo exato para a cobertura de um segmento

Em [16], Li et al. consideram o problema de cobertura de um segmento de reta utilizando discos cujos centros são dados por pontos sobre o segmento. São consideradas duas versões: na primeira, denominada *discreta*, é dado, para cada centro, um conjunto finito de possíveis valores que podem ser atribuídos aos raios; na segunda versão, denominada *contínua*, é dado, para cada centro, um intervalo contínuo de valores que podem ser atribuídos aos raios dos discos. É considerado, sem perda de generalidade, que o segmento é horizontal e que seu extremo esquerdo se encontra na origem do plano cartesiano. Para a versão discreta, o custo de um disco é determinado por uma função do raio atribuído a ele. Observa-se que tais funções podem ser arbitrárias e independentes do valor do raio. Já na versão contínua, são considerados custos dados por uma função proporcional a r^α , para uma constante α , onde r é o raio atribuído ao disco. O objetivo é atribuir um raio a cada centro de forma que os discos assim definidos cubram o segmento com custo total mínimo, dado pela soma dos custos dos discos.

Para a versão discreta, é apresentado em [16] um algoritmo exato de tempo polinomial. Para a versão contínua, são apresentados algoritmos aproximados. Especificamente, para $\alpha = 1$, é apresentado um algoritmo 1.25-aproximado e um FPTAS. Para $\alpha > 1$ é apresentado um algoritmo 2-aproximado.

Nesta dissertação há um maior interesse pelo algoritmo apresentado para a versão discreta, o qual estendemos, obtendo um FPTAS para uma versão mais geral do problema em que os centros dos discos são dados por pontos no plano e não necessariamente sobre o segmento.

3.1.1 Transformação da instância em um grafo direcionado

Seja l um segmento horizontal de comprimento ℓ , cujo extremo esquerdo se encontra na origem, um conjunto C de n pontos sobre o segmento e n conjuntos R_i contendo no máximo μ valores r_j que podem ser atribuídos aos centros $c_i \in C$. Para cada raio $r_j \in R_i$ é dado um custo ω_{ij} para que seja atribuído ao centro $c_i \in C$. O objetivo é determinar para cada centro $c_i \in C$ um raio $r_j \in R_i$ de maneira que o segmento l seja coberto com custo mínimo. A solução é dada por uma lista S de valores s_{ij} , sendo que a atribuição do valor 1 para s_{ij} representa a escolha do raio $r_j \in R_i$ para o centro $c_i \in C$ e o valor 0 para $s_{ij} \in S$ representa que tal disco não foi utilizado na solução. O custo de uma solução S é dado por $\omega(S) = \sum_{s_{ij} \in S} \omega_{ij} s_{ij}$.

O problema pode ser considerado em apenas uma dimensão, com o segmento l e os pontos $c_i \in C$ localizados sobre o eixo X . Dessa maneira os centros c_i podem ser descritos

apenas por sua coordenada x_i . O disco D_{ij} definido por um raio $r_j \in R_i$ e um centro c_i cobre o intervalo $[x_i - r_j, x_i + r_j]$ do segmento. Esta versão do problema é considerada em [16] a versão discreta do problema MCLC (*Min-Cost Linear Coverage*).

Para resolver o problema é utilizado um grafo direcionado $G = (V, E)$ como descrito a seguir. O conjunto de vértices é dado por $V = \{v_{ij} \mid 1 \leq i \leq n, 1 \leq j \leq k\} \cup \{s, t\}$ onde s representa um ponto na posição 0, t representa um ponto na posição ℓ e cada v_{ij} representa a escolha do raio $r_j \in R_i$ para o centro $c_i \in C$. Existirá uma aresta direcionada de um vértice v_{ij} a outro vértice v_{ab} caso os discos representados pelos mesmos cubram o intervalo entre c_i e c_a . No caso particular dos vértices especiais s e t , existirá uma aresta partindo de s até um vértice v_{ij} caso o disco representado por v_{ij} cubra o ponto na posição 0 e existirá uma aresta partindo de um vértice v_{ab} com destino a t caso o disco representado por v_{ab} cubra o extremo do segmento na posição ℓ . As arestas recebem como peso o custo do disco representado pelo vértice de origem, exceto para aquelas que têm como origem o vértice s , que recebem como peso o valor zero.

A construção do grafo é realizada pelo Algoritmo 3.

Algoritmo 3: Construção do grafo $G(V, E)$

Entrada: Segmento l (dado pelos extremos $x = 0$ e $x = \ell$), um conjunto de centros C e um conjunto de raios R_i para cada $c_i \in C$

Saída: Grafo direcionado $G(V, E)$

```

1  $V \leftarrow \{s, t\}$ ;
   //  $s$  representa  $x = 0$  e  $t$  representa  $x = \ell$ 
2  $E \leftarrow \emptyset$ ;
3 para todo  $c_i \in C$  faça
   // Para todos os centros
4   para todo  $r_j \in R_i$  faça
     // Para todos os possíveis raios
5      $V \leftarrow V \cup \{v_{ij}\}$ ;
6     se  $r_j \geq x_i$  então
7        $E \leftarrow E \cup \{e = (s, v_{ij})\}$ ;
8        $w(e) = 0$ ;
9     se  $x_i + r_j \geq \ell$  então
10       $E \leftarrow E \cup \{e = (v_{ij}, t)\}$ ;
11       $w(e) = \omega_{ij}$ ;
12    para todo  $v_{ab} \in V \setminus \{s, t, v_{ij}\}$  faça
      // Para todos os outros vértices, exceto  $s$  e  $t$ 
13      se  $x_a > x_i$  e  $r_b + r_j \geq x_a - x_i$  então
14         $E \leftarrow E \cup \{e = (v_{ij}, v_{ab})\}$ ;
15         $w(e) = \omega_{ij}$ ;
16      senão
17        se  $x_i > x_a$  e  $r_j + r_b \geq x_i - x_a$  então
18           $E \leftarrow E \cup \{e = (v_{ab}, v_{ij})\}$ ;
19           $w(e) = \omega_{ab}$ ;

```

Pela construção do grafo, se houver uma aresta com origem no vértice s e destino no vértice v_{ij} , então o disco definido pela atribuição do raio $r_j \in R_i$ ao centro $c_i \in C$ cobre o intervalo entre o extremo esquerdo do segmento e o centro c_i . De maneira análoga, se existe uma aresta partindo de um vértice v_{ij} e chegando ao vértice t , então o disco representado por v_{ij} cobre o intervalo do segmento entre c_i e ℓ . Se houver uma aresta com origem em um vértice v_{ij} e destino em um vértice v_{ab} , então os dois discos representados por v_{ij} e v_{ab} cobrem o intervalo entre c_i e c_a . Dessa forma, um caminho entre dois vértices no grafo fornece uma cobertura da parcela do segmento entre os pontos representados pelos seus extremos, dada pelos discos representados pelos vértices do mesmo. Assim, se

existir um caminho entre os vértices s e t , então haverá uma cobertura para o segmento dada pelos discos representados pelos vértices que compõem tal caminho.

O problema da cobertura do segmento com custo mínimo se torna equivalente ao problema de encontrar o caminho de menor custo entre s e t no grafo G , pois cada caminho entre s e t corresponde a uma solução viável para a cobertura do segmento e o somatório dos custos das arestas corresponde ao custo dos discos utilizados na solução. Pode-se obter o caminho mínimo no grafo entre s e t utilizando, por exemplo, o algoritmo de Dijkstra. A solução retornada é obtida atribuindo a $s_{ij} \in S$ o valor 1 caso o caminho de custo mínimo obtido contenha o vértice v_{ij} e o valor 0 em caso contrário.

A Figura 3.1 ilustra o grafo construído a partir de uma instância do problema.

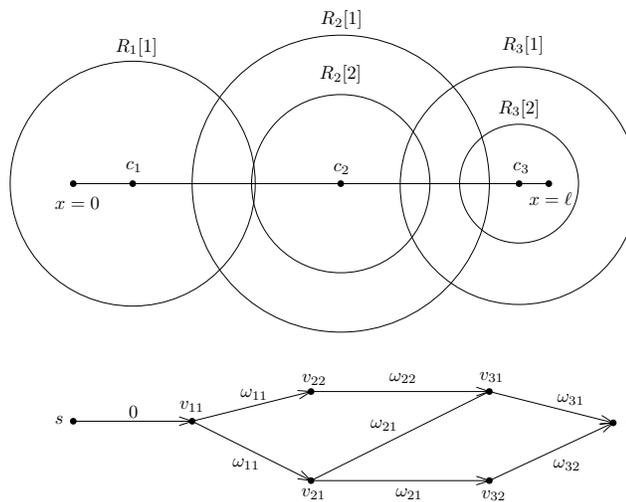


Figura 3.1: Exemplo da construção de um grafo a partir de uma instância.

3.1.2 Obtenção de uma solução ótima e análise do algoritmo

No Algoritmo 3, pode-se verificar que na Linha 5 serão adicionados no máximo $n\mu$ vértices, cada um representando a combinação de um centro $c_i \in C$ e um de seus possíveis raios. Com a adição dos vértices que representam os extremos do segmento na Linha 1, o grafo construído terá no máximo $n\mu + 2$ vértices. O laço da Linha 12 é executado para cada um dos vértices, adicionando arestas nas Linhas 14 ou 18. No pior caso, será executado $O(n^2\mu^2)$ vezes.

No artigo [16] não é feita uma análise da complexidade do tempo necessária para a determinação do caminho mínimo no grafo construído pelo Algoritmo 3, sendo apenas destacado que pode ser realizada em tempo polinomial devido ao número polinomial de vértices. Pode-se observar que o grafo G construído pelo Algoritmo 3 é um DAG (*Directed*

Acyclic Graph), pois a origem das arestas orientadas representam pontos cuja coordenada X é sempre menor que o ponto representado pelo destino das mesmas. Portanto, não há arestas de retorno que criariam ciclos no grafo. Dado que o grafo é um DAG, é possível determinar um caminho mínimo entre s e t em tempo $\Theta(|V| + |E|)$ utilizando uma ordenação topológica e o algoritmo de Dijkstra.

Portanto é possível encontrar uma solução ótima em tempo total $O(n^2\mu^2)$ para a cobertura do segmento, o qual é polinomial.

3.2 Um FPTAS para a cobertura de um segmento

Por meio de uma extensão do algoritmo exato de Li et al. apresentado em [16], obtivemos um FPTAS para o problema da cobertura de um segmento utilizando discos cujos centros são dados por um conjunto discreto de pontos no plano e cujos raios podem assumir quaisquer valores. Além disso, tratamos um caso mais geral do problema do que o do artigo [16], pois aqui não é considerada a restrição de que os pontos que definem os centros dos discos devem necessariamente estar sobre o segmento.

Seja l um segmento de reta e $C = \{c_1, \dots, c_n\}$ um conjunto de pontos no plano. Considera-se que o custo de um disco é dado por $\omega(r) = r^\alpha$ onde α é constante e r é o raio do disco. O problema consiste em selecionar uma lista $S = (r_1, \dots, r_n)$ de valores para os raios dos discos centrados em $c_i \in C$ de forma que o segmento l seja coberto com custo mínimo, dado pela soma dos custos dos discos definidos por S . O custo de uma solução S será definido como $\omega(S) = \sum_{r_i \in S} \omega(r_i) = \sum_{r_i \in S} r_i^\alpha$.

Esta versão do problema será denominada MLSDCP (*Minimum Line Segment Disk Cover Problem*).

3.2.1 Descrição do algoritmo proposto

A estratégia adotada para encontrar uma solução aproximada para o problema foi definir uma discretização para os possíveis raios dos discos, obtendo uma instância do problema considerado em [16], cuja solução pode ser encontrada por um algoritmo exato. Aplicando-se o Algoritmo 3 sobre tal instância, obtém-se um grafo no qual o caminho mínimo entre dois vértices especiais s e t fornece uma solução aproximada para o problema original. Demonstramos na próxima seção que a solução obtida é uma $(1 + \varepsilon)$ -aproximação, para qualquer $\varepsilon > 0$. O tempo total de execução é polinomial em $\frac{1}{\varepsilon}$ e no tamanho da entrada, concluindo que o algoritmo proposto é um FPTAS.

Assume-se que $0 < \varepsilon < 4\alpha$, e tomam-se $\xi = \varepsilon/(4\alpha)$ e $\ell = \|a, b\|_1$, onde a e b são os extremos de l , ou seja, ℓ recebe o comprimento do segmento l na métrica L_1 . A vantagem da utilização da métrica L_1 ao invés da distância Euclidiana é que a representação da

primeira é menor que a segunda e pelo fato de que a distância entre dois pontos quaisquer na métrica L_1 é sempre maior que a distância entre tais pontos na métrica L_2 , o que facilita a demonstração de limitantes para o custo de uma solução.

A discretização é definida de forma que os valores criados estejam entre os valores r_{\min} , o menor, e r_{\max} , o maior. O valor de r_{\min} é definido de forma que n discos com tal raio tenham custo menor ou igual ao custo de uma solução ótima para o problema, ou seja, o custo de n discos de raio r_{\min} será um limitante inferior para o custo de uma solução ótima. Será demonstrado na próxima seção, no Lema 3.2.2, que o valor $r_{\min} = \xi\ell/(2n)$ satisfaz tal requisito.

Já o valor de r_{\max} é definido de forma que um disco de raio r_{\max} centrado em qualquer ponto $c \in C$ cubra o segmento l , ou seja, tal disco é uma solução viável para o problema. Assim, o custo de um disco de raio r_{\max} é um limitante superior para o custo de uma solução ótima para o problema. No algoritmo proposto é utilizado o valor $r_{\max} = \max_{c \in C, p \in l} \|c, p\|_1$, ou seja, o valor de r'_{\max} é a maior distância entre um centro $c \in C$ e um extremo p de l na métrica L_1 . Nota-se que qualquer raio em uma solução pode ser limitada por r_{\max} pois qualquer ponto p do segmento l está distante de qualquer centro por no máximo $\max_{c \in C, p \in l} \|c, p\|_2 \leq \max_{c \in C, p \in l} \|c, p\|_1$. Ambos r_{\min} e r_{\max} possuem representações computáveis em tempo polinomial.

A discretização para os valores dos raios é definida pelo conjunto $R = \{r_{\min}, r_{\min}(1 + \xi)^1, r_{\min}(1 + \xi)^2, \dots, r_{\min}(1 + \xi)^{m-1}\}$ onde m é o menor inteiro para o qual $r_{\min}(1 + \xi)^{m-1} \geq r_{\max}$. A Figura 3.2 exemplifica a discretização dos valores dos raios.

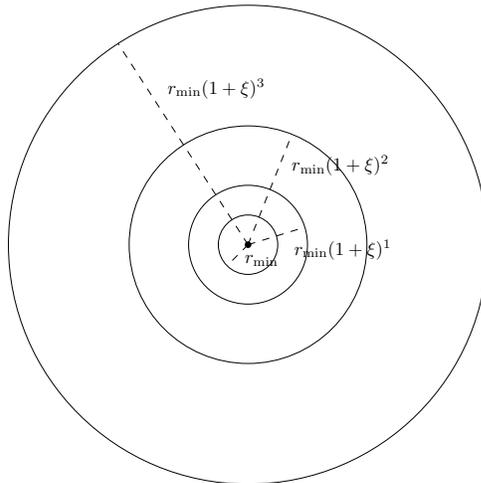


Figura 3.2: Exemplo da discretização de raios.

Utilizando a discretização dada por R , existem m possíveis raios para cada centro, definindo nm discos possíveis. Cada um desses discos que interceptam o segmento l são

substituídos por discos que cobrem a mesma porção de l , mas centrados sobre o segmento de reta, no ponto médio da porção coberta e com raio igual à metade da porção coberta, como ilustrado na Figura 3.3.

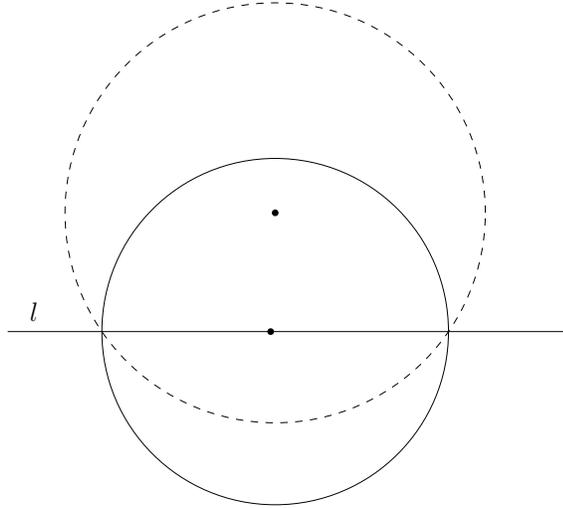


Figura 3.3: Exemplo da substituição do disco original (linha pontilhada) por um disco centrado sobre o segmento (linha cheia).

Seja C' o conjunto dos centros de discos assim obtidos; para cada centro $c'_i \in C'$ um conjunto R'_i contendo os raios dos discos criados centrados em c'_i ; para cada um dos possíveis raios $r'_j \in R'_i$ o custo associado ω_{ij} , o qual é definido como o custo do disco substituído, proveniente da instância original.

Os conjuntos C' , R'_i juntamente com os custos associados ω_{ij} para cada possível raio $r'_j \in R'_i$ configuram uma instância da versão discreta do problema MCLC considerado em [16]. Assim é possível construir um grafo utilizando o algoritmo 3 e encontrar uma solução exata por meio de um algoritmo de caminho mínimo, como o algoritmo de Dijkstra. A partir da solução S' encontrada dessa maneira é possível obter uma solução S para a instância original utilizando os discos correspondentes, os quais foram substituídos na construção da instância reduzida. O custo da solução S é o mesmo da solução S' , pois os custos dos discos criados é o mesmo dos discos correspondentes.

Dada uma instância $(C, l, \alpha, \varepsilon)$, o Algoritmo 4 é utilizado para encontrar uma solução aproximada.

3.2.2 Análise do algoritmo

O custo e a complexidade de tempo do algoritmo dependem da quantidade de valores criados na discretização definida pelo conjunto R , na linha 7 do Algoritmo 4.

Lema 3.2.1. *O número $m = |R|$ é polinomial no tamanho da entrada e em $1/\varepsilon$*

Demonstração. Como m é o menor inteiro para o qual $r_{\min}(1 + \xi)^{m-1} \geq r_{\max}$, tem-se que $m = \lceil \log_{1+\xi}(r_{\max}/r_{\min}) \rceil + 1$. Portanto,

$$\begin{aligned} m &= \lceil \log_{1+\xi}(r_{\max}/r_{\min}) \rceil + 1 \\ &\leq \frac{\log(r_{\max}/r_{\min})}{\log(1 + \xi)} + 2 \end{aligned}$$

Sejam $c \in C$ um centro de disco e p um extremo do segmento tal que $r_{\max} = \|c, p\|_1$, ou seja, c é o centro de disco que possui a maior distância a um dos extremos do segmento. Então:

$$m \leq \frac{\log(2n\|c, p\|_1/(\xi\ell))}{\log(1 + \xi)} + 2$$

Pelo fato de que $\log(1 + \xi) \geq \xi = \varepsilon/4\alpha$ quando $0 \leq \xi \leq 1$, temos:

$$\begin{aligned} m &\leq \frac{\log(2n) + \log(4\alpha/\varepsilon) + \log(\|c, p\|_1/\ell)}{\varepsilon/4\alpha} + 2 \\ &= O\left(\frac{\alpha}{\varepsilon} (\log(n) + \log(\alpha/\varepsilon) + \log(\|c, p\|_1/\ell))\right), \end{aligned}$$

□

No lema seguinte é demonstrado um limitante inferior para a solução ótima de uma instância do problema MLSDCP.

Lema 3.2.2. *Se S^* é uma solução ótima para uma instância do problema MLSDCP, então $\omega(S^*) \geq n\omega(\ell/(4n))$.*

Demonstração. Seja uma RMLSDCP a versão relaxada do problema MLSDCP em que devem-se definir os valores dos raios (r_1, \dots, r_n) de n discos e cada disco pode ter seu centro posicionado em qualquer ponto do plano Euclidiano para a cobertura do segmento. É simples observar que qualquer solução para o problema MLSDCP em que são dados n centros de discos e o mesmo segmento de reta l também é uma solução para o problema RMLSDCP. Uma solução ótima para uma instância do problema RMLSDCP é dada por n discos cujos centros são posicionados igualmente espaçados sobre o segmento, atribuindo a cada disco o raio $\|a, b\|_2/(2n)$, onde a e b são os extremos do segmento a ser coberto. Nota-se que, como ω é uma função convexa, tal solução tem custo mínimo. Como $\|a, b\|_2 \geq \|a, b\|_1/2 = \ell/2$, segue o resultado. □

O próximo lema determina a complexidade de tempo do Algoritmo 4.

Lema 3.2.3. *O Algoritmo 4 tem complexidade de tempo $O(n^2m^2)$.*

Demonstração. Pelo Lema 3.2.1, o número de possíveis valores discretizados para o tamanho dos raios m é polinomial no tamanho da entrada e em $1/\varepsilon$. Portanto, a complexidade de tempo da etapa em que são obtidos os possíveis valores dos raios é $O(m)$. Na criação da instância reduzida são executados dois laços que iteram sobre todos os valores possíveis de raios $r_i \in R$ para cada centro $c_i \in C$ resultando em uma complexidade de tempo $O(nm)$. Ao final é utilizado o algoritmo de Li et al. apresentado na Seção 3.1 para encontrar uma solução exata para a instância reduzida. A complexidade de tal algoritmo é quadrática no número de discos, resultando numa complexidade $O(n^2m^2)$. Portanto a complexidade do Algoritmo 4 é $O(n^2m^2)$. \square

Para concluir a demonstração de que o Algoritmo 4 é um FPTAS, é necessário demonstrar que a solução obtida é uma $(1 + \varepsilon)$ -aproximação para o problema.

Lema 3.2.4. *O Algoritmo 4 fornece uma solução $(1 + \varepsilon)$ -aproximada.*

Demonstração. Seja $I = (C, l, \alpha, \varepsilon)$ uma instância para o problema. Seja $S^* = (r_1^*, \dots, r_n^*)$ uma solução ótima para I e $S = (r_1, \dots, r_n)$ a solução obtida por meio do algoritmo 4. Seja $S' = (r'_1, \dots, r'_n)$ uma solução obtida a partir de S^* por meio do arredondamento de cada raio r_i^* para o menor raio $r'_i \in R$ tal que $r'_i > r_i^*$. Como S' foi obtido a partir de S^* aumentando-se os raios da solução, é claro que S' também é uma solução viável para a instância I .

Seja $B = \{i \in 1, \dots, n : r'_i > r_{\min}\}$ o conjunto dos índices dos raios dos discos determinados por S' maiores que r_{\min} e $T = \{1, \dots, n\} \setminus B$ os índices dos raios menores ou iguais a r_{\min} . Como S é uma solução ótima para a cobertura quando os raios são restritos àqueles contidos em R , tem-se que $\omega(S) \leq \omega(S')$. Assim, resta demonstrar que S' tem custo limitado por um fator $(1 + \varepsilon)$ do custo da solução S^* . Isto é,

$$\omega(S) \leq \omega(S') = \sum_{i \in B} \omega(r'_i) + \sum_{i \in T} \omega(r_{\min})$$

Pelo fato de ω ser uma função não decrescente e $r'_i \leq (1 + \xi)r_i^*$, obtemos:

$$\omega(S) \leq \sum_{i \in B} \omega((1 + \xi)r_i^*) + \sum_{i \in T} f\left(\frac{\xi \ell}{4n}\right)$$

Como $\omega(r) = r^\alpha$:

$$\omega(S) \leq (1 + \xi)^\alpha \sum_{i \in B} \omega(r_i^*) + \xi^\alpha n f\left(\frac{\ell}{4n}\right)$$

Do Lema 3.2.2, segue-se:

$$\begin{aligned}\omega(S) &\leq (1 + \xi)^\alpha \omega(S^*) + \xi^\alpha \omega(S^*) \\ &= \left(\left(1 + \frac{\varepsilon/4}{\alpha}\right)^\alpha + \left(\frac{\varepsilon/4}{\alpha}\right)^\alpha \right) \omega(S^*)\end{aligned}$$

Já que $(1 + x/\alpha)^\alpha \leq e^x$ quando $|x| \leq \alpha$ e $\alpha \geq 1$, obtém-se:

$$\leq (e^{\varepsilon/4} + \varepsilon/4) \omega(S^*)$$

Utilizando-se do fato que $e^x \leq 1 + 2x$ quando $0 \leq x \leq 1$, temos:

$$\begin{aligned}&\leq (1 + 2(\varepsilon/4) + \varepsilon/4) \omega(S^*) \\ &\leq (1 + \varepsilon) \omega(S^*),\end{aligned}$$

□

Com a demonstração desses lemas, pode-se enunciar o teorema

Teorema 3.2.1. *O Algoritmo 4 é um FPTAS.*

Demonstração. Segue dos Lemas 3.2.3 e 3.2.4.

□

No próximo capítulo serão apresentados algoritmos para a cobertura de objetos geométricos descritos por um sistema de inequações polinomiais.

Capítulo 4

Cobertura de Objetos Geométricos

Uma versão do problema de cobertura por discos é o problema de cobertura de objetos geométricos. Muitos problemas práticos podem ser modelados como uma instância deste problema, como o de cobertura de uma cidade pelo sinal de antenas. Nesse caso, a cidade poderia ser modelada como um objeto geométrico no plano e o sinal das antenas modelados como discos centrados na posição destas, cujos raios seriam o alcance dos sinais das antenas. Com essa modelagem, uma cobertura do objeto geométrico por discos forneceria a posição e o alcance das antenas de forma que toda a cidade seja provida pelo sinal das antenas.

Mais formalmente, seja \mathcal{L} um conjunto de objetos geométricos descritos por um sistema de inequações polinomiais, C um conjunto de n pontos que definem os centros c_i dos discos e $\omega(r) = r^\alpha$ a função que atribui o custo a um disco de raio r , onde α é uma constante. O objetivo é determinar uma coleção de números não negativos $S = (r_1, \dots, r_n)$ que definem uma lista de discos $\mathcal{D}(D(c_1, r_1), \dots, D(c_n, r_n))$ de forma que \mathcal{D} cubra \mathcal{L} e o custo total desses discos, $\omega(S) = \sum_{r \in S} \omega(r)$, seja minimizado. Este problema é denominado *Minimum Geometric Disk Cover Problem*, MGDCP.

Dado um objeto geométrico g , denota-se por $\mathcal{P}(g)$ o conjunto de pontos de g . De maneira similar, dado um conjunto de objetos geométricos \mathcal{L} , $\mathcal{P}(\mathcal{L}) = \bigcup_{g \in \mathcal{L}} \mathcal{P}(g)$. Além disso, diz-se que uma coleção de objetos geométricos G é coberta por outra coleção de objetos geométricos \mathcal{L}' se $\mathcal{P}(G) \subseteq \mathcal{P}(\mathcal{L}')$.

4.1 Um PTAS para cobertura de objetos geométricos

Nesta seção descrevemos um PTAS que desenvolvemos para o problema da cobertura por discos de objetos geométricos bem comportados, isto é, que podem ser descritos por um sistema de inequações algébricas de grau limitado por uma constante. Tais objetos geométricos compreendem polígonos e objetos mais complexos. Nesta dissertação, tais ob-

jetos serão referenciados simplesmente como objetos geométricos, omitindo-se a expressão “bem comportados”.

A idéia geral é utilizar uma partição do espaço em um número polinomial de regiões. Para cada região que contém pontos de um objeto geométrico é obtido um ponto representante. Uma cobertura dos pontos representantes é, então, construída, da qual conseguimos uma cobertura para os objetos geométricos originais.

Nas próximas seções, serão apresentados o algoritmo elaborado e a sua análise.

4.1.1 Descrição do algoritmo proposto

Seja uma instância $I = (C, \mathcal{L}, \alpha)$ do problema, onde C é o conjunto dos centros dos discos, \mathcal{L} é um sistema de γ inequações polinomiais de grau no máximo δ que descrevem os objetos a serem cobertos, e $\omega(r) = r^\alpha$ é a função que determina o custo de um disco. Seguem o algoritmo proposto e a descrição dos seus passos.

Algoritmo 5: PTAS para cobertura de objetos geométricos

Entrada: Conjunto de objetos geométricos \mathcal{L} , um conjunto de centros C e inteiros $\alpha \geq 1$ e $\varepsilon \geq 0$

Saída: Lista S de valores para os raios de cada centro $c_i \in C$

- 1 $\xi \leftarrow \varepsilon / (8\alpha)$;
- 2 $r_{\min} \leftarrow \xi \cdot \max_{O \in \mathcal{L}} \{ \max_{p, q \in O} \{ \|p, q\|_2 \} \} / 2n$;
- 3 $r_{\max} \leftarrow \max_{c \in C, O \in \mathcal{L}} \{ \max_{p \in O} \{ \|c, p\|_1 \} \}$;
- 4 $m \leftarrow$ o menor inteiro tal que $r_{\min}(1 + \xi)^{m-1} \geq r_{\max}$;
- 5 $\mathcal{P} \leftarrow \emptyset$;
- 6 $S \leftarrow \emptyset$;
- 7 $R \leftarrow \{r_{\min}, r_{\min}(1 + \xi)^1, r_{\min}(1 + \xi)^2, \dots, r_{\min}(1 + \xi)^{m-1}\}$;
- 8 $\mathcal{R} \leftarrow$ arranjo dado por círculos centrados em $c_i \in C$ e raios $r_j \in R$;
- 9 **para** toda face h' do arranjo \mathcal{R} **faça**

//	Obtenção dos pontos representantes
10	se $h' \cap O \neq \emptyset, O \in \mathcal{L}$ então
11	$\mathcal{P} \leftarrow \mathcal{P} \cup \{p(h')\}$, onde $p(h') \in h' \cap O$;
- 12 $\mathcal{X} \leftarrow \mathcal{P} \cup C$;
- 13 **para** todo $p_i \in \mathcal{X}$ **faça**

//	Definição de \mathcal{F}
14	se $p_i \in C$ então
15	$F_i \leftarrow 0$;
16	senão
17	$F_i \leftarrow \infty$;
- 18 $\hat{S} \leftarrow$ solução da instância do MSCP $I = (\mathcal{X}, \mathcal{F}, 2, \alpha)$;
- // Construção da solução para os objetos geométricos
- 19 $S \leftarrow \{r_i : r_i = \max\{r_{\min}, (1 + \xi)\hat{r}_i\}\}$;

O primeiro passo do algoritmo proposto, na Linha 7, é a partição do espaço em um número polinomial de regiões. Para tanto, será construído um arranjo de círculos centrados nos pontos $c_i \in C$, cujos raios serão dados por um conjunto finito de valores. O número de raios criados deve ser polinomial no tamanho da entrada, pois dessa maneira é obtido um número polinomial de regiões. Os valores dos raios estarão limitados entre r_{\min} , o menor valor, e r_{\max} , o maior.

O valor de r_{\min} será escolhido de maneira que qualquer solução para a cobertura dos objetos geométricos seja maior que o custo de n discos de raio r_{\min} . A cobertura dos objetos geométricos terá custo pelo menos igual ao custo da cobertura mínima do diâmetro do maior objeto conexo de \mathcal{L} . A cobertura mínima do diâmetro do maior objeto, por sua vez, é limitado inferiormente pelo custo de uma cobertura em que todos

os discos são dispostos sobre o diâmetro igualmente espaçados entre si. Tal limitante foi estabelecido no Lema 3.2.2 e se deve à convexidade da função de custo dos discos. Seja ℓ o diâmetro do maior objeto e os pontos p e q os seus extremos. Assim $\|\ell\| = \|p, q\|_2$. O valor de r_{\min} será escolhido de tal forma que $r_{\min} \leq \|p, q\|_2/n$. Observa-se que a distância entre p e q na norma L_2 é maior que a metade da distância entre p e q na norma L_1 , ou seja, $\frac{\|p, q\|_1}{2} \leq \|p, q\|_2$. Assim, dado $\varepsilon > 0$, seja $\xi = \varepsilon/(8\alpha)$ e tomemos $r_{\min} = \xi \cdot \max_{O \in \mathcal{L}} \{\max_{p, q \in O} \{\|p, q\|_1\}\}/2n$. Seja $S_{r_{\min}} = n\omega(r_{\min})$, S_L^* o custo de uma cobertura mínima do diâmetro do maior objeto conexo contido em \mathcal{L} e S^* o custo de uma cobertura mínima dos objetos geométricos contidos em \mathcal{L} . Pode-se afirmar que:

$$S_{r_{\min}} = n\omega(\xi \cdot \max_{O \in \mathcal{L}} \{\max_{p, q \in O} \{\|p, q\|_1\}\}/2n) \leq \xi^\alpha S_L^* \leq \xi^\alpha S^*. \quad (4.1)$$

Por outro lado, r_{\max} será escolhido de forma que um disco de raio r_{\max} centrado em qualquer ponto $c_i \in C$ cubra os objetos geométricos descritos por \mathcal{L} . Tal disco é uma solução para o problema, ou seja, uma solução ótima terá custo menor ou igual ao custo de um disco de raio r_{\max} . Assim, toma-se $r_{\max} = \max_{c \in C, O \in \mathcal{L}} \{\max_{p \in O} \{\|c, p\|_1\}\}$, ou seja, a maior distância entre um centro $c_i \in C$ e um ponto de um objeto geométrico na métrica L_1 . A métrica L_1 é utilizada pois a distância entre dois pontos na métrica L_1 é sempre maior ou igual à distância entre os mesmos pontos na métrica Euclidiana.

O caso degenerado em que o conjunto de objetos geométricos a serem cobertos são um conjunto finito de pontos pode ser resolvido pelo algoritmo proposto em [5], o qual foi descrito na Seção 2.1. Portanto, será considerado que a instância a ser resolvida não é degenerada.

Seja $R = \{r_{\min}, r_{\min}(1 + \xi)^1, r_{\min}(1 + \xi)^2, \dots, r_{\min}(1 + \xi)^{m-1}\}$ um conjunto de valores para os raios, onde m é o menor inteiro para o qual $r_{\min}(1 + \xi)^{m-1} \geq r_{\max}$. Considere todos os círculos centrados nos centros $c_i \in C$ cujos raios são determinados pela discretização dada por R . São obtidos nm círculos, os quais definem um arranjo cuja complexidade combinatória total é quadrática no número de círculos (veja [14]) e, portanto, o número total de regiões (faces) do arranjo é $O(n^2m^2)$. Um exemplo de arranjo obtido é ilustrado na Figura 4.1

O segundo passo do algoritmo proposto, na Linha 11, é a obtenção de pontos representantes para as regiões que têm intersecção não nula com objetos geométricos de \mathcal{L} . Seja \mathcal{R} o conjunto das faces que intersectam pelo menos um objeto geométrico de \mathcal{L} . Cada uma dessas faces pode ser descrita por um sistema de inequações polinomiais. Para cada um desses sistemas, podem-se adicionar as inequações que limitam os objetos geométricos para obter uma descrição da intersecção entre uma face e os objetos geométricos. Utilizando o algoritmo apresentado em [13] para a solução de sistemas de inequações polinomiais, é possível obter, para cada região $h \in \mathcal{R}$, um ponto representante $p(h)$ que pertence a um objeto geométrico de \mathcal{L} que intersecta h . Um exemplo de possíveis pontos representantes

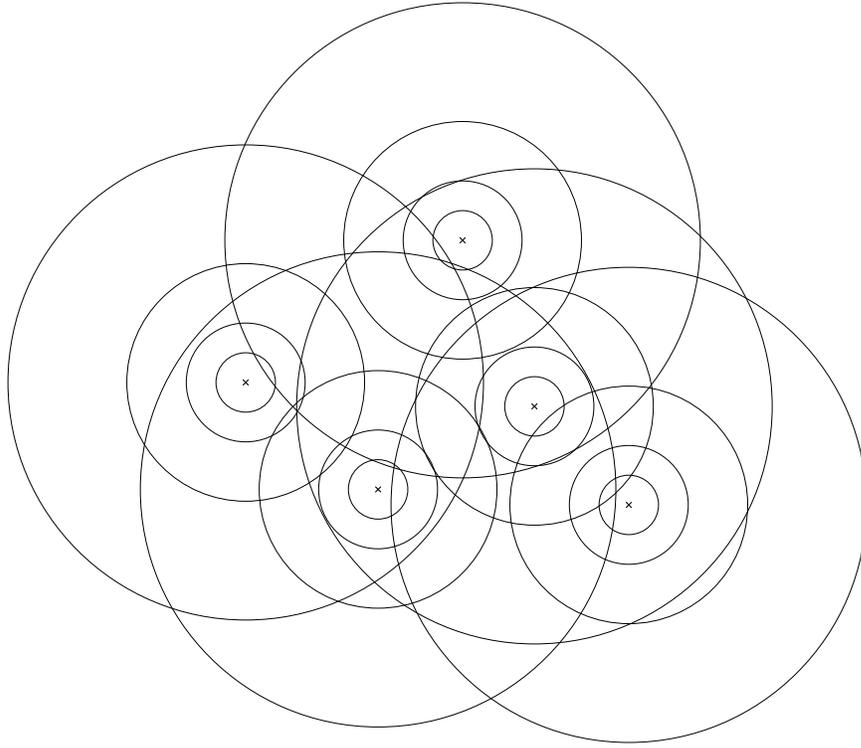


Figura 4.1: Exemplo de arranjo gerado a partir da discretização dada por R .

para um polígono é apresentado na Figura 4.2

O terceiro passo do algoritmo proposto, na Linha 18, consiste na obtenção de uma cobertura para os pontos representantes. Para tanto, é criada uma instância do problema MSCP. Seja $\mathcal{P} = \{p(h) : h \in \mathcal{R}\}$ o conjunto de pontos representantes a serem cobertos. Define-se um conjunto de pontos $\mathcal{X} = C \cup \mathcal{P}$. Para cada ponto $p_i \in \mathcal{X}$, define-se um custo fixo F_i , sendo $F_i = 0$, caso $p_i \in C$, e $F_i = \infty$ (ou um valor suficientemente grande), caso $p_i \in \mathcal{P}$. Tais custos fixos definem a coleção \mathcal{F} de maneira que discos centrados em pontos $p_i \in \mathcal{P}$ não sejam utilizados na solução. Um dos algoritmos aproximados conhecido para encontrar uma solução para a instância criada é o proposto em [5], apresentado na Seção 2.1.

O quarto e último passo do algoritmo, na Linha 18, é a transformação da solução assim obtida numa solução para a cobertura dos objetos geométricos. Observa-se que existe um mapeamento de cada ponto de \mathcal{P} para uma região de \mathcal{R} , as quais foram obtidas pela intersecção de círculos concêntricos com raios crescentes por um fator de $(1 + \xi)$. Além disso, os discos que compõem a cobertura dos pontos representantes estão também centrados em pontos $c_i \in C$. Dessa forma, cada disco de uma cobertura de \mathcal{P} terá raio intermediário entre dois raios da discretização dada por R . Seja $r_{\mathcal{P}}$ o raio de um disco

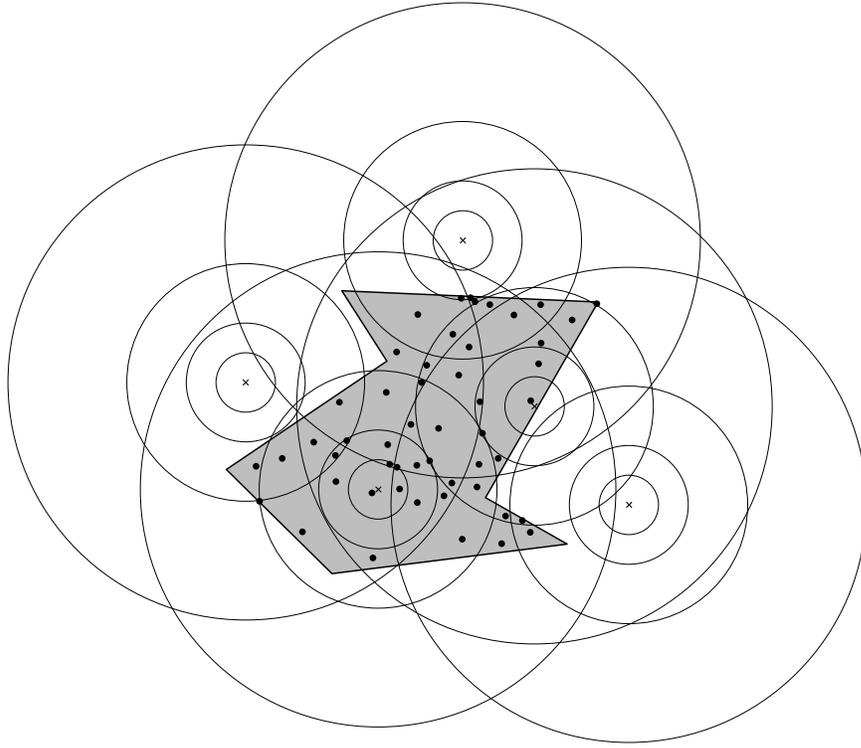


Figura 4.2: Exemplo de pontos representantes para um polígono.

de uma cobertura para os pontos de \mathcal{P} . Existem raios $r, r' \in R$ tais que $r' = r(1 + \xi)$ e $r \leq r_{\mathcal{P}} \leq r'$. Assim, se o raio $r_{\mathcal{P}}$ for aumentado de $(1 + \xi)$, resultará num raio $r'_{\mathcal{P}} = r_{\mathcal{P}}(1 + \xi)$ tal que $r'_{\mathcal{P}} \geq r'$. Com isso, o aumento dos discos de uma cobertura de \mathcal{P} por um fator de $(1 + \xi)$ cobrirá os objetos geométricos de \mathcal{L} . Seja $\hat{S} = (\hat{r}_1, \dots, \hat{r}_n)$ uma solução para a cobertura dos pontos representantes. Uma cobertura para os objetos geométricos $S = (r_1, \dots, r_n)$ é obtida aumentando-se os discos de \hat{S} , sendo $r_i = (1 + \xi)\hat{r}_i$ quando $\hat{r}_i \geq r_{\min}$ e $r_i = r_{\min}$ em caso contrário.

Na Figura 4.3 são ilustrados os passos do algoritmo. Em (a) é construído o arranjo de discos (cujos centros estão marcados por “x”); em (b) são determinados os pontos representantes; em (c) é obtida uma cobertura dos pontos representantes; em (d) são aumentados os discos por um fator de $(1 + \xi)$, obtendo-se uma cobertura do objeto geométrico.

Na próxima seção, será apresentada a análise do algoritmo proposto e a demonstração de que o mesmo é um PTAS para o problema.

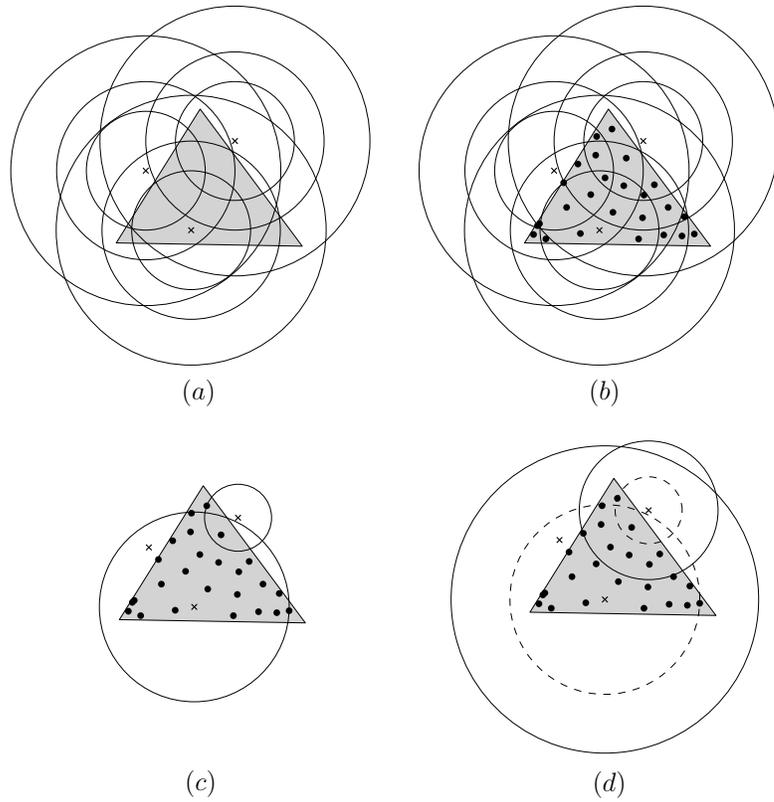


Figura 4.3: Passos do algoritmo proposto.

4.1.2 Análise do algoritmo

No primeiro passo do algoritmo, resumido na Linha 7, é criada uma discretização para a partição do espaço em faces determinadas por um arranjo de círculos, o qual é construído na Linha 8. Tal discretização é dada pelo conjunto $R = \{r_{\min}, r_{\min}(1 + \xi)^1, r_{\min}(1 + \xi)^2, \dots, r_{\min}(1 + \xi)^{m-1}\}$, que determina valores para raios de discos centrados nos pontos $c_i \in C$. É necessário que o número de faces do arranjo seja polinomial para que o algoritmo tenha complexidade de tempo polinomial. O seguinte lema garante que a discretização gerada, e conseqüentemente o número de faces do arranjo criado, é polinomial.

Lema 4.1.1. *O número $m = \|R\|$ é polinomial em n e em $1/\varepsilon$.*

Demonstração. Como m é o menor inteiro para o qual $r_{\min}(1 + \xi)^{m-1} \geq r_{\max}$, tem-se que:

$$\begin{aligned} m &= \left\lceil \log_{1+\xi}(r_{\max}/r_{\min}) \right\rceil + 1 \\ &\leq \frac{\log(r_{\max}/r_{\min})}{\log(1 + \xi)} + 2 \end{aligned}$$

Seja $c \in C$ e p um ponto de um objeto geométrico tal que $r_{\max} = \|c, p\|_1$ e $\|s, t\|_1$ o diâmetro do maior objeto conexo $O \in \mathcal{L}$ na métrica L_1 :

$$m \leq \frac{\log(2n\|c, p\|_1/(\xi\|s, t\|_1))}{\log(1 + \xi)} + 2$$

Utilizando o fato de que $\log(1 + \xi) \geq \xi = \varepsilon/8\alpha$ quando $0 \leq \xi \leq 1$:

$$\begin{aligned} m &\leq \frac{\log(2n) + \log(8\alpha/\varepsilon) + \log(\|c, p\|_1/\|s, t\|_1)}{\varepsilon/8\alpha} + 2 \\ &= O\left(\frac{\alpha}{\varepsilon} (\log(n) + \log(\alpha/\varepsilon) + \log(\|c, p\|_1/\|s, t\|_1))\right). \end{aligned}$$

Portanto, o número m de raios criados na discretização dada por R é polinomial em n e em $1/\varepsilon$. \square

A complexidade combinatória total do arranjo é quadrática no número de círculos, (veja em [14]) e já que são obtidos nm círculos, o número de faces é polinomial em n e m .

No segundo passo, resumido na Linha 11, são determinados pontos representantes para todas as regiões definidas pelo arranjo que têm intersecção não-nula com um objeto de \mathcal{L} . Para tanto, é necessário resolver o sistema de inequações que representa cada região de \mathcal{R} , com a adição das inequações que representam os objetos geométricos de \mathcal{L} , obtendo assim um sistema que descreve a intersecção das faces do arranjo com os objetos geométricos.

Lema 4.1.2. *A complexidade de tempo necessário para determinar um ponto representante para cada região de \mathcal{R} é polinomial em n e em $1/\varepsilon$, caso os coeficientes das funções de \mathcal{L} e ω possam ser descritos por no máximo b bits.*

Demonstração. Para cada região, é necessária a solução de um sistema de inequações com no máximo mn equações que descrevem os discos, juntamente com as γ inequações que descrevem os objetos geométricos, resultando em sistemas com no máximo $(mn + \gamma)$ inequações. O algoritmo apresentado em [13] para a solução de um sistema em tais condições é executado em tempo $b((mn + \gamma)\delta)^{d^2}$, onde b é o comprimento máximo em bits da representação dos coeficientes dos polinômios do sistema, δ é o grau máximo dos polinômios e d é a dimensão do espaço. Com b, γ, δ e d constantes, o tempo total é polinomial em mn . O algoritmo é executado para cada região, e o número deles é polinomial em mn , sendo-o também o tempo total para determinar o conjunto \mathcal{P} de pontos representantes. Pelo Lema 4.1.1, m é polinomial em n e em $1/\varepsilon$. Portanto, a complexidade de tempo necessário para determinar os pontos representantes para as faces do arranjo \mathcal{R} é polinomial em n e em $1/\varepsilon$. \square

No terceiro passo, na Linha 18, é obtida uma solução para a cobertura dos pontos representantes.

Lema 4.1.3. *A complexidade de tempo necessário para a obtenção de uma cobertura por discos dos pontos representantes é polinomial em n .*

Demonstração. A construção da instância do problema MSCP leva tempo proporcional ao número de pontos de \mathcal{X} , o qual é construído pela união dos conjuntos de centros de discos, C , e os pontos representantes \mathcal{P} . O número de pontos de \mathcal{P} é proporcional ao número de faces do arranjo, o qual é quadrático em n e m . A solução \hat{S} é obtida por meio do algoritmo proposto em [5] (veja Seção 2.1). A complexidade de tempo do algoritmo é $O\left(\|\mathcal{X}\|^{\frac{\alpha}{\varepsilon}O(d)}\right)$, ou seja, é polinomial no número de pontos a serem cobertos. Pelo Lema 4.1.1, m é polinomial em n . Portanto, a complexidade de tempo da obtenção de uma cobertura por discos dos pontos representantes é $O\left((mn)^{\frac{\alpha}{\varepsilon}O(d)}\right)$, o qual é polinomial em n . \square

O quarto passo, no laço da Linha 18, consiste no aumento dos raios da solução obtida para a cobertura dos pontos representantes por um fator de $(1 + \xi)$ e dos raios que permanecerem menores que r_{\min} para r_{\min} , levando tempo proporcional ao número de centros de discos n .

Lema 4.1.4. *A complexidade de tempo total do Algoritmo 5 é polinomial em n .*

Demonstração. A complexidade de tempo do primeiro passo, no qual é construída a partição do espaço por um arranjo de círculos, é proporcional ao número de raios dados pela discretização, sendo, portanto, $O(m)$. No segundo passo, são obtidos pontos representantes para as regiões que têm intersecção não nula com um objeto geométrico. Pelo Lema 4.1.2, a complexidade de tempo desse passo é polinomial em n . No terceiro passo, é obtida uma cobertura para os pontos representantes. Pelo Lema 4.1.3, a complexidade de tempo desse passo também é polinomial em n . O quarto passo, no qual são aumentados os valores dos raios dos discos da cobertura dos pontos representantes, tem custo de tempo proporcional ao número de centros de discos, dado por n . Portanto o tempo total do Algoritmo 5 é polinomial em n , dominado pelo custo de tempo da cobertura dos pontos representantes do terceiro passo. Assim, o custo de tempo total é $O\left((mn)^{\frac{\alpha}{\varepsilon}O(d)}\right)$, o qual é polinomial em n . \square

Lema 4.1.5. *O Algoritmo 5 fornece uma solução $(1 + \varepsilon)$ -aproximada para o MGDGP.*

Demonstração. Seja $I = (\mathcal{C}, \mathcal{L}, \alpha)$ uma instância do problema MGDGP. Seja $S^* = (r_1^*, \dots, r_n^*)$ uma solução ótima para I e $S = (r_1, \dots, r_n)$ uma solução obtida pelo Algoritmo 5. Seja \hat{S}^* uma solução ótima para o problema MSCP que cobre os pontos representantes \mathcal{P} e \hat{S} a solução obtida por meio do Algoritmo 1 para a cobertura de \mathcal{P} .

A solução S é obtida a partir da solução $\hat{S} = (\hat{r}_1, \dots, \hat{r}_n)$ aumentando-se cada raio por um fator de $(1 + \xi)$ e os raios que continuarem menores que r_{\min} para r_{\min} . Seja

$B = \{i \in \{1, \dots, n\} : r_i = (1 + \xi)\hat{r}_i > r_{\min}\}$ o conjunto de (índices de) discos definidos por S que possuem raios maiores que r_{\min} e $T = \{1, \dots, n\} \setminus B$ o conjunto de discos cujos raios foram aumentados para r_{\min} . Como cada ponto de \mathcal{P} pertence a pelo menos um objeto geométrico, tem-se que $\omega(\hat{S}^*) \leq \omega(S^*)$, pois qualquer solução para a cobertura dos objetos geométricos é também uma solução para a cobertura dos pontos representantes. Portanto:

$$\begin{aligned} \omega(S) &= \sum_{i \in B} \omega((1 + \xi)\hat{r}_i) + \sum_{i \in T} \omega(r_{\min}) \\ &\leq (1 + \xi)^\alpha \omega(\hat{S}) + n\omega(r_{\min}) \end{aligned}$$

Como a solução \hat{S} é obtida por meio do Algoritmo 1, o qual é um PTAS, ou seja, $\omega(\hat{S}) \leq (1 + \xi)\omega(\hat{S}^*)$ e utilizando o limitante da Equação 4.1:

$$\begin{aligned} \omega(S) &\leq (1 + \xi)^\alpha (1 + \xi)\omega(\hat{S}^*) + \xi^\alpha \omega(S^*) \\ &\leq \left(1 + \frac{\varepsilon/8}{\alpha}\right)^{\alpha+1} \omega(S^*) + \left(\frac{\varepsilon/8}{\alpha}\right)^\alpha \omega(S^*) \end{aligned}$$

Pelo fato de que $(1 + x/\alpha)^\alpha \leq e^x$ quando $|x| \leq \alpha$ e $\alpha \geq 1$, temos:

$$\omega(S) \leq \left((e^{\varepsilon/8})(1 + \varepsilon/8) + \left(\frac{\varepsilon/8}{\alpha}\right)^\alpha \right) \omega(S^*)$$

Como $e^x \leq 1 + 2x$ quando $0 \leq x \leq 1$, obtém-se:

$$\begin{aligned} &\leq \left(\left(1 + 2\left(\frac{\varepsilon}{8}\right)\right) \left(1 + \frac{\varepsilon}{8}\right) + \frac{\varepsilon}{8} \right) \omega(S^*) \\ &\leq (1 + \varepsilon)\omega(S^*) \end{aligned}$$

□

Como o algoritmo tem complexidade de tempo polinomial e fornece uma $(1 + \varepsilon)$ -aproximação, pode-se enunciar o seguinte teorema:

Teorema 4.1.1. *O Algoritmo 5 é um PTAS para o problema MGDCP.*

Demonstração. Segue dos Lemas 4.1.4 e 4.1.5.

□

4.1.3 Alternativa para um caso mais restrito

No terceiro passo do Algoritmo 5, na Linha 18, pode-se utilizar qualquer algoritmo para encontrar a cobertura dos pontos representantes. Caso a complexidade de tempo do algoritmo utilizado para a cobertura de pontos seja polinomial, é possível obter um PTAS para

o problema. O Algoritmo 5 pode, portanto, ser considerado um framework o qual reduz o problema da cobertura de objetos geométricos para a cobertura de pontos, obtendo uma solução aproximada.

Em casos mais restritos, é possível obter resultados melhores com a utilização de algoritmos mais específicos. O caso em que os custos são dados por funções lineares ($\omega(r) = r^\alpha$, com $\alpha = 1$), é um desses casos. O Algoritmo 2 é um algoritmo polinomial exato para a solução de tal problema, caso seja possível comparar duas soluções em tempo polinomial. A suposição de que é possível comparar duas soluções em tempo polinomial é necessária devido ao fato de que a complexidade de tempo da comparação de duas somas de raízes quadradas de inteiros não é conhecida (veja [9]). Se os pontos a serem cobertos tiverem coordenadas inteiras, a comparação de duas soluções necessariamente recairá em tal problema.

Lema 4.1.6. *Seja $I = (\mathcal{C}, \mathcal{L}, 1)$ uma instância com custo linear do problema MGDCP. O custo de tempo total do Algoritmo 5 é polinomial em n e em $1/\varepsilon$, assumindo que o custo de tempo necessário para comparar duas somas de raízes quadradas de inteiros é polinomial.*

Demonstração. O número m de valores $r_i \in R$ criados na discretização é polinomial em n e em $1/\varepsilon$, como demonstrado no Lema 4.1.1. O custo de tempo para a determinação dos pontos representantes é polinomial em n e em $1/\varepsilon$, como demonstrado no Lema 4.1.2. Para a instância criada, o número de pontos a serem cobertos é proporcional ao número de faces do arranjo dos círculos cujos raios foram obtidos pela discretização dada por R . Como são obtidos nm círculos e o número de regiões criadas pelo arranjo é quadrático no número de círculos (como demonstrado em [14]), serão obtidos $O(n^2m^2)$ pontos representantes. A complexidade de tempo necessário para determinar uma cobertura ótima dos pontos representantes no terceiro passo do Algoritmo 5, nesse caso, é dada pela complexidade de tempo do Algoritmo 2, $O(\nu^{881}T(\nu))$, onde ν é o número de pontos a serem cobertos e $T(\nu)$ é o custo de tempo necessário para comparar duas soluções. Portanto, o custo de tempo necessário para obter uma cobertura ótima para os pontos representantes é $O((nm)^{1762}T((nm)^2))$, o qual é polinomial em n e $1/\varepsilon$. No quarto passo do Algoritmo 5, na Linha 18, os valores dos raios dos discos da solução obtida para a cobertura dos pontos representantes são aumentados em um fator de $(1+\varepsilon)$ e, caso o raio obtido seja menor que r_{\min} , o mesmo é aumentado para r_{\min} . Tal procedimento tem custo de tempo proporcional ao número de centros de discos, dado por n . Assim o tempo total do Algoritmo 5 é polinomial em n e em $1/\varepsilon$. \square

Observa-se que para este caso mais restrito, apenas o terceiro passo do algoritmo, na Linha 18, foi modificado em relação ao caso mais geral analisado na seção anterior. Assim a demonstração da aproximação dada pelo algoritmo é semelhante ao da seção anterior.

Lema 4.1.7. *Seja $I = (\mathcal{C}, \mathcal{L}, 1)$ uma instância com custo linear do problema MGDCP. O Algoritmo 5 fornece uma solução $(1 + \varepsilon)$ -aproximada.*

Demonstração. Seja $I = (\mathcal{C}, \mathcal{L}, 1)$ uma instância do problema MGDCP. Seja $S^* = (r_1^*, \dots, r_n^*)$ uma solução ótima para I e $S = (r_1, \dots, r_n)$ uma solução obtida pelo Algoritmo 5. Seja \hat{S}^* uma solução ótima para o problema MSCP que cobre os pontos representantes \mathcal{P} , obtida por meio do Algoritmo 2.

A solução S é obtida a partir da solução $\hat{S}^* = (\hat{r}_1^*, \dots, \hat{r}_n^*)$ aumentando cada raio por um fator de $(1 + \xi)$ e os raios que continuarem menores que r_{\min} para r_{\min} . Seja $B = \{i \in \{1, \dots, n\} : r_i = (1 + \xi)\hat{r}_i^* > r_{\min}\}$ o conjunto de (índices de) discos definidos por S que possuem raios maiores que r_{\min} e $T = \{1, \dots, n\} \setminus B$ o conjunto de discos cujos raios foram aumentados para r_{\min} . Como cada ponto de \mathcal{P} pertence a pelo menos um objeto geométrico, tem-se que $\omega(\hat{S}^*) \leq \omega(S^*)$, pois qualquer solução para a cobertura dos objetos geométricos é também uma solução para a cobertura dos pontos representantes. Portanto:

$$\begin{aligned} \omega(S) &= \sum_{i \in B} \omega((1 + \xi)\hat{r}_i^*) + \sum_{i \in T} \omega(r_{\min}) \\ &\leq (1 + \xi)\omega(\hat{S}^*) + n\omega(r_{\min}) \end{aligned}$$

Utilizando o limitante da Equação 4.1:

$$\begin{aligned} \omega(S) &\leq (1 + \xi)\omega(\hat{S}^*) + \xi\omega(S^*) \\ &\leq \left(1 + \frac{\varepsilon}{8}\right)\omega(S^*) + \left(\frac{\varepsilon}{8}\right)\omega(S^*) \\ &\leq \left(1 + \frac{\varepsilon}{4}\right)\omega(S^*) \\ &\leq (1 + \varepsilon)\omega(S^*) \end{aligned}$$

□

Com a demonstração dos Lemas 4.1.6 e 4.1.7, obtém-se o seguinte teorema.

Teorema 4.1.2. *O Algoritmo 5 fornece um FPTAS quando $\alpha = 1$ e assumindo que é possível comparar duas soluções em tempo polinomial, o que permite a utilização do Algoritmo 2 para encontrar uma cobertura ótima dos pontos representantes no terceiro passo do algoritmo.*

No próximo capítulo serão apresentadas as conclusões.

Capítulo 5

Conclusões

O algoritmo exato proposto por Li et al. em [16] para a cobertura de um segmento por discos centrados sobre o segmento cujos raios devem ser escolhidos dentre os valores dados por um conjunto finito foi apresentado no Capítulo 3. Esta versão do problema é denominada *discreta* devido à restrição sobre os possíveis raios dos discos. Estendemos esse algoritmo para a obtenção de soluções aproximadas para uma versão contínua do problema. Pode-se dizer que a versão que consideramos é mais geral pois os discos não são necessariamente centrados sobre o segmento. Apresentamos uma redução, por meio de uma discretização dos possíveis valores dos raios dos discos, para transformar uma instância do problema que consideramos numa instância do problema discreto tratado em [16], tornando possível utilizar o algoritmo exato ali proposto como uma subrotina para obtenção de uma solução aproximada. Demonstramos que o algoritmo que propusemos é um FPTAS, encontrando soluções $(1 + \varepsilon)$ -aproximadas em tempo polinomial no tamanho da entrada e em $1/\varepsilon$. A relevância desse resultado se deve à inexistência na literatura de um PTAS para o problema que consideramos.

No Capítulo 2, apresentamos algoritmos existentes na literatura para a cobertura de pontos, especificamente os propostos por Bilò et al. [5] e Gibson et al. [12]. Estendemos tais algoritmos no Capítulo 4 para encontrar soluções aproximadas para o problema de cobertura de objetos geométricos “bem comportados”, i.e., dados por um sistema de inequações polinomiais. De maneira semelhante ao caso da cobertura de um segmento, apresentamos uma redução que torna possível, por meio de uma discretização, a utilização de um algoritmo para a cobertura de pontos como subrotina para encontrar uma cobertura para objetos geométricos mais complexos.

Demonstramos que utilizando o algoritmo de [5] como subrotina, o algoritmo que propusemos é um PTAS para a cobertura de objetos geométricos, fornecendo soluções $(1 + \varepsilon)$ -aproximadas em tempo polinomial no tamanho da entrada. Tal resultado se aplica para instâncias em que os custos dos discos são dados por polinômios de grau $\alpha > 1$ e

com dimensão d finita. Não havia ainda nenhum resultado semelhante na literatura para tratar de um caso tão geral quanto o que consideramos.

Na Seção 4.1.3, demonstramos que, caso $\alpha = 1$ e considerando que seja possível comparar duas soluções em tempo polinomial, pode-se utilizar o algoritmo proposto em [12] como subrotina para a cobertura de pontos. Neste caso, demonstramos que o algoritmo que desenvolvemos fornece um FPTAS para o problema de cobertura de objetos geométricos.

A complexidade de tempo do algoritmo que apresentamos é dominado principalmente pela complexidade do algoritmo que for utilizado como subrotina para a cobertura de pontos. Assim, caso surjam algoritmos de complexidade de tempo menor para a cobertura de pontos, é possível reduzir a complexidade de tempo para a cobertura de objetos geométricos. Um possível trabalho futuro seria o de investigar casos mais restritos do problema em que se possa aplicar um algoritmo mais eficiente para a cobertura de pontos, resultando em um algoritmo aproximado de complexidade de tempo menor.

Os principais resultados apresentados nesta dissertação aparecem no artigo [8] publicados na revista *Operations Research Letters*.

Referências Bibliográficas

- [1] A. K. Abu-Affash, P. Carmi, M. J. Katz, and G. Morgenstern. Multi cover of a polygon minimizing the sum of areas. In *Proceedings of the 5th international conference on WALCOM: algorithms and computation*, pages 134–145, Berlin, 2011. Springer-Verlag.
- [2] A. Agnetis, E. Grande, P. Mirchandani, and A. Pacifici. Covering a line segment with variable radius discs. *Comp. Oper. Res.*, 36(5):1423–1436, May 2009.
- [3] H. Alt, E. Arkin, H. Brönnimann, J. Erickson, S. Fekete, C. Knauer, J. Lenchner, J. Mitchell, and K. Whittlesey. Minimum-cost coverage of point sets by disks. In *Proc. of the 22nd Ann. Symp. on Comp. Geom.*, SoCG, pages 449–458, New York, 2006. ACM.
- [4] R. Bar-Yehuda and D. Rawitz. A note on multicovering with disks. *Comput. Geom. Theory Appl.*, 46(3):394–399, Apr. 2013.
- [5] V. Bilò, I. Caragiannis, C. Kaklamanis, and P. Kanellopoulos. Geometric clustering to minimize the sum of cluster sizes. In G. Brodal and S. Leonardi, editors, *Algorithms – ESA*, volume 3669 of *LNCS*, pages 460–471. Springer-Verlag, 2005.
- [6] G. Das, R. Fraser, A. Lòpez-Ortiz, and B. Nickerson. On the discrete unit disk cover problem. In N. Katoh and A. Kumar, editors, *WALCOM: Algorithms and Computation*, volume 6552 of *LNCS*, pages 146–157. Springer-Verlag, 2011.
- [7] D. Dash, A. Bishnu, A. Gupta, and S. Nandy. Approximation algorithm for line segment coverage for wireless sensor network. *CoRR*, abs/1006.2955, 2010.
- [8] P. J. de Rezende, F. K. Miyazawa, and A. T. Sasaki. A PTAS for the disk cover problem of geometric objects. *Operations Research Letters*, 41(5):552–555, 2013.
- [9] E. D. Demaine, J. S. B. Mitchell, and J. O’Rourke. Problem 33: Sum of square roots. <http://maven.smith.edu/orourke/TOPP/P33.html>, Aug. 2012.

- [10] T. Erlebach, K. Jansen, and E. Seidel. Polynomial-time approximation schemes for geometric graphs. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, SODA '01, pages 671–679, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.
- [11] A. Freund and D. Rawitz. Combinatorial interpretations of dual fitting and primal fitting. In R. Solis-Oba and K. Jansen, editors, *Approximation and Online Algorithms*, volume 2909 of *LNCS*, pages 317–318. Springer-Verlag, 2004.
- [12] M. Gibson, G. Kanade, E. Krohn, I. Pirwani, and K. Varadarajan. On clustering to minimize the sum of radii. *SIAM Journal on Computing*, 41(1):47–60, 2012.
- [13] D. Y. Grigor'ev and N. N. Vorobjov Jr. Solving systems of polynomial inequalities in subexponential time. *Journal of Symbolic Computation*, 5(1–2):37 – 64, 1988.
- [14] D. Halperin. Arrangements. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 24, pages 529–562. CRC Press LLC, Boca Raton, FL, 2004.
- [15] N. Lev-Tov and D. Peleg. Polynomial time approximation schemes for base station coverage with minimum total radii. *Comp. Net.*, 47(4):489–501, 2005.
- [16] M. Li, X. Sun, and Y. Zhao. Minimum-cost linear coverage by sensors with adjustable ranges. In Cheng, Yu et al., editor, *Wireless Algorithms, Systems, and Applications*, volume 6843 of *LNCS*, pages 25–35. Springer-Verlag, 2011.
- [17] N. Mustafa and S. Ray. Improved results on geometric hitting set problems. *Discrete Computational Geometry*, 44:883–895, 2010.

Apêndice A

Lista de acrônimos

- BSPP: *Base Station Placement Problem*
- DAG: *Directed Acyclic Graph*
- DUDC: *Discrete Unit Disk Cover*
- FPTAS: *Fully Polynomial Time Approximation Scheme*
- MCLC: *Min-cost Linear Coverage*
- MDC: *Minimum Disk Cover*
- MDDC: *Minimum Discrete Disk Cover Problem*
- MGDCP: *Minimum Geometric Disk Cover Problem*
- MLSDCP: *Minimum Line Segment Disk Cover Problem*
- MSC: *Minimum Set Cover*
- MSCP: *Min-size k-clustering problem*
- PTAS: *Polynomial Time Approximation Scheme*
- RMLSDCP: *Versão relaxada do problema Minimum Line Segment Disk Cover Problem*