



Daniel Filipe Almeida Bandeira

Otimização de um sistema de impressão tridimensional bimaterial

Dissertação de Mestrado em Matemática, Área de Especialização em Análise Aplicada e Computação, orientada pela Professora Doutora Marta Margarida Braz Pascoal e apresentada ao Departamento de Matemática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Junho 2017



UNIVERSIDADE DE COIMBRA

Otimização de um sistema de impressão tridimensional bimaterial

Daniel Filipe Almeida Bandeira



UNIVERSIDADE DE COIMBRA

Mestrado em Matemática

Master in Mathematics

Dissertação de Mestrado | MSc Dissertation

Junho 2017

Agradecimentos

Para a conclusão desta dissertação de mestrado foi essencial o apoio de várias pessoas, às quais agora me dirijo para expressar o meu agradecimento.

À Professora Doutora Marta Margarida Braz Pascoal, orientadora desta dissertação, pelo apoio, disponibilidade, conhecimentos transmitidos e contribuições valiosas para a realização deste trabalho.

Aos meus pais, por todo o suporte emocional e material que me deram ao longo de todos estes anos de estudos.

Aos restantes familiares, pelo carinho e encorajamento.

Aos colegas de curso, pelos cinco anos de companheirismo e vários momentos partilhados.

Aos amigos, que sempre me apoiaram nas várias etapas ultrapassadas até hoje.

Este trabalho foi desenvolvido no âmbito do Projeto PT2020-POCI-SII & DT 17963: NEXT.Parts, Next-Generation of Advanced Hybrid Parts, do programa Portugal 2020, através do COMPETE 2020 – Programa Operacional Competitividade e Internacionalização

Cofinanciado por:



Resumo

Neste trabalho estuda-se um sistema de impressão tridimensional no caso bimaterial e utilizando estereolitografia. A zona de impressão e o objeto a imprimir são representados por meio de uma discretização espacial em *voxels* e estuda-se a projeção de um feixe de luz *laser* no plano de impressão. O problema de impressão tridimensional bimaterial é dividido em outros dois. Primeiro, analisa-se o problema da localização de emissores capazes de atingir todas as partes do polímero constituinte do objeto a imprimir, com o objetivo de minimizar o número de emissores utilizados. Em seguida é feita a afetação entre os emissores selecionados e as zonas do polímero a atingir, com os objetivos de maximizar o ângulo incidência da luz *laser* no plano de impressão e minimizar o número de emissores ativados em cada camada. Para ambos os casos, são apresentadas formulações de otimização linear inteira e são desenvolvidos vários métodos, heurísticos e exatos. Por fim é analisado um caso de estudo utilizado para experiência computacional, para o qual se discutem alguns resultados.¹

¹A imagem da capa foi retirada de: Creative Tools. 2015. “3DBenchy printed on a resin printer”. *Creative Tools*. Acedido a 30 de maio de 2017. https://www.flickr.com/photos/creative_tools/17345089792/.

Conteúdo

| | |
|-------------------------------------------------------------------------------|-----------|
| Lista de Figuras | ix |
| Lista de Tabelas | xi |
| 1 Introdução | 1 |
| 1.1 Impressão tridimensional por processo aditivo | 1 |
| 1.2 Impressão tridimensional bimaterial | 2 |
| 1.3 Objetivo e organização | 2 |
| 2 Impressão tridimensional bimaterial | 5 |
| 2.1 Descrição geral do problema | 5 |
| 2.2 Matriz de cobertura dos emissores | 6 |
| 2.3 Projeção do <i>laser</i> | 8 |
| 3 Problema da localização de emissores | 15 |
| 3.1 Modelo do problema | 15 |
| 3.2 Problema de cobertura de conjuntos | 16 |
| 3.3 Métodos heurísticos para o problema de localização de emissores | 19 |
| 4 Problema da afetação de emissores | 25 |
| 4.1 Formulação do problema | 25 |
| 4.2 Modelo de soma pesada | 28 |
| 4.3 Modelo lexicográfico | 31 |
| 5 Experiência computacional | 35 |
| 5.1 Caso de estudo | 35 |
| 5.2 Problema da localização de emissores | 37 |
| 5.3 Problema da afetação de emissores | 40 |
| 5.4 Análise das soluções obtidas | 41 |
| 6 Conclusão | 47 |
| Bibliografia | 49 |

Lista de Figuras

| | | |
|-----|-----------------------------------------------------------------------------------------------------------|----|
| 2.1 | Zona de impressão | 9 |
| 2.2 | Ângulos α e θ | 9 |
| 2.3 | Distorção da luz <i>laser</i> ao atingir um <i>voxel</i> | 10 |
| 2.4 | Áreas a calcular | 10 |
| 2.5 | Projeção da luz <i>laser</i> num <i>voxel</i> | 11 |
| 2.6 | Área da secção da elipse | 14 |
| 2.7 | Área da secção da elipse dentro do <i>voxel</i> | 14 |
| 5.1 | (a) Zona de impressão e objeto a imprimir; (b) Plano horizontal intermédio do objeto a imprimir | 35 |
| 5.2 | PLE: soluções e tempos de execução dos Algoritmos 2 e 3 | 38 |
| 5.3 | PAE: soluções e tempos de execução | 41 |
| 5.4 | PAE: tempos de execução da abordagem A5 e do Algoritmo 5 | 41 |
| 5.5 | Áreas de distorção | 42 |

Lista de Tabelas

| | | |
|------|------------------------------------------------------------------------------|----|
| 5.1 | Parâmetros dos testes | 37 |
| 5.2 | PLE: soluções e tempos de execução (segundos) com CPLEX | 37 |
| 5.3 | PLE: soluções e tempos de execução (segundos) dos Algoritmos 2 e 3 | 38 |
| 5.4 | PLE: soluções e tempos de execução (segundos) da heurística H1 | 39 |
| 5.5 | PLE: soluções e tempos de execução (segundos) da heurística H2 | 39 |
| 5.6 | PAE: soluções | 43 |
| 5.7 | PAE: tempos de execução (segundos) do modelo lexicográfico | 44 |
| 5.8 | PAE: tempos de execução (segundos) das várias abordagens | 44 |
| 5.9 | PAE: tempos de execução (segundos) do Algoritmo 5 | 44 |
| 5.10 | Áreas de distorção do objeto | 45 |

Capítulo 1

Introdução

1.1 Impressão tridimensional por processo aditivo

Os processos de prototipagem rápida são divididos em processos aditivos, em que o objeto é construído por adição de matéria, e processos subtrativos, onde o objeto final é obtido por desbaste da matéria prima. A impressão tridimensional, é um processo aditivo de prototipagem rápida, onde o objeto final é criado pela adição de sucessivas camadas de material. Cada camada corresponde a uma “fatia transversal” do objeto que se pretende construir, sendo que a impressora desenha cada camada como se de uma impressão bidimensional se tratasse. Após a adição de cada camada, o sólido desce exatamente na mesma medida da espessura da camada, para a próxima camada ser adicionada, de forma a que todas as camadas sejam desenhadas sobre o mesmo plano. Podem ser encontrados mais detalhes sobre estes processos em [5, 12].

A tecnologia e os materiais necessários para este tipo de processos começaram a ser desenvolvidos no final do século passado, na década de 80. As primeiras impressões tridimensionais surgiram em 1984, apesar da dificuldade, nessa altura, de acesso aos equipamentos necessários para a impressão, devido aos seus elevados custos. Desde a sua criação que têm sido feitas várias melhorias nos métodos de impressão, e já é possível adquirir impressoras que fazem este tipo de impressão por menos de 1000 euros.

Um modelo do objeto que se pretende imprimir é desenvolvido por computador e, por essa razão, as impressoras devem possuir um *software* próprio para o efeito. O *software* precisa de definir as várias camadas a imprimir do objeto a partir do modelo. As tecnologias utilizadas para este tipo de impressão ainda se encontram sob forte desenvolvimento. Por exemplo, algumas impressoras utilizam filamentos de plástico que são fundidos por aquecimento e depositados na plataforma onde é criado o objeto. Outras impressoras utilizam fusão a *laser*, sendo os materiais mais utilizados o metal e o plástico.

A tecnologia para impressão tridimensional em que estamos interessados denomina-se estereolitografia. Neste caso a adição das camadas é conseguida a partir de um polímero líquido, que solidifica quando exposto a uma luz *laser* ultravioleta. Apenas a zona do polímero onde a luz do *laser* incide passa a sólida, permanecendo o restante polímero no estado líquido. A plataforma que serve de base ao modelo construído move-se então para a impressão de uma nova cada camada. Para mais informações relativamente a esta tecnologia pode consultar-se [14].

1.2 Impressão tridimensional bimaterial

Usando o processo de impressão descrito no ponto anterior, pretende-se imprimir um objeto em que o polímero é solidificado em torno de uma grelha de metal. Como este é um processo não tratado, torna-se essencial certificar as condições necessárias para fazer a impressão pretendida.

A utilização de um segundo material na impressão, em particular a utilização de uma grelha metálica construída *a priori* e que serve de suporte ao polímero, levanta dificuldades adicionais à impressão. Com efeito, através de um processo tradicional, em que apenas é utilizado um *laser* fixo que orienta o seu feixe em qualquer direção do espaço de impressão, ao incidir na estrutura metálica a luz proveniente do *laser*, pode deixar de incidir sobre algumas zonas da superfície polimérica, que ficarão “sombreadas”.

É de referir que tem existido uma procura crescente de componentes complexos, que combinem dois ou mais materiais, e possuam características acrescidas, em termos físicos e de funcionalidade. Estes materiais estão a evoluir em termos de aplicabilidade e funcionalidade. Como tal, procuram-se novos métodos de fabrico que as possam produzir de forma viável em termos de processo e sustentada em termos económicos. Uma tecnologia que tem sido bastante usada para a sua produção é a moldagem por injeção [16]. A técnica aqui estudada é particularmente relevante para a produção de peças complexas em pequena série, como por exemplo, ortóteses personalizadas, componentes inteligentes (com sensores embebidos, por exemplo), ou peças complexas ou frágeis em que utilizar injeção não é viável.

1.3 Objetivo e organização

A solução para a impressão tridimensional bimaterial estudada neste trabalho consiste em, adicionalmente ao *laser* utilizado no processo usual e colocado na parte superior da impressora, distribuir sistemas de espelhos galvanométricos adicionais pelas paredes laterais da impressora, por forma a utilizá-los para refletir a luz proveniente do *laser* com orientações bem escolhidas. Pretende-se, deste modo, atingir as partes da camada polimérica que é necessário solidificar e que não podem ser atingidas diretamente pela luz *laser*. Considera-se que os sistemas de espelhos são estáticos, e portanto que se mantêm no mesmo sítio durante toda a impressão, sendo apenas possível orientá-los, por forma a que estes reflitam a luz *laser* com os ângulos pretendidos.

As primeiras questões abordadas neste trabalho estão relacionadas com a representação do espaço onde a impressão é realizada. Após algumas considerações iniciais é apresentada uma representação simplificada do espaço de impressão e modela-se a projeção de um feixe de luz sobre a camada de impressão como sendo a projeção de um ponto sobre um plano. Esta informação, combinada com o conhecimento dos obstáculos que a luz pode encontrar, é utilizada para construir uma matriz de cobertura do espaço de impressão. A matriz de cobertura é parte dos modelos apresentados para o problema de impressão tridimensional bimaterial. É igualmente estudada a projeção de um círculo num determinado plano, com o propósito de avaliar a distorção da luz *laser* ao incidir na camada de impressão e obter uma medida da qualidade do objeto impresso.

O sistema para impressão tridimensional com dois materiais proposto é tratado em duas fases. Na primeira fase pretende-se determinar quantos sistemas de espelhos são necessários, e qual a sua

melhor disposição, para permitir uma impressão completa. Este problema é denominado por problema da localização de emissores. Na segunda fase, posterior à determinação dos sistemas de espelhos a utilizar, é necessário estabelecer que zonas da superfície polimérica devem ser atingidas por cada sistema de espelhos. Este problema designa-se de problema da afetação de emissores.

O problema da localização de emissores é formulado como um programa linear inteiro, em particular um problema de cobertura de conjuntos, cujas restrições são definidas a partir da matriz de cobertura referida acima. O problema de cobertura de conjuntos é um problema NP-completo, e portanto a sua resolução exata é condicionada pela dimensão dos dados. São estudados resultados que permitem reduzir, e simplificar, o problema. Estudam-se também procedimentos gerais para lidar com o problema de cobertura em termos heurísticos e, em seguida, apresentam-se dois métodos heurísticos de implementação simples, que permitem calcular soluções admissíveis deste problema.

O problema da afetação de emissores é modelado como um programa linear inteiro com duas funções objetivo, uma relacionada com o ângulo de incidência da luz *laser* na camada de impressão (e implicitamente a distorção do *laser*) e outra representando o número de emissores que tem que ser ativado para cada camada a imprimir. Discutem-se duas formas de abordar o problema linear inteiro bicritério: uma tendo por base uma abordagem em termos de soma pesada dos dois objetivos, e outra em que opta por um tratamento lexicográfico dos mesmos objetivos. Em geral é necessária a utilização de um *solver* de programação matemática para determinar soluções eficientes de qualquer dos modelos propostos. Para um dos casos particulares destes modelos é apresentado um algoritmo, do tipo guloso e exato.

Por fim, os modelos desenvolvidos para as duas fases do problema de impressão tridimensional com dois materiais são aplicados a um caso de estudo, para o qual as metodologias propostas são comparadas em termos das soluções que produzem, em particular considerando a distorção do objeto final impresso associada a cada resolução. É igualmente avaliada a eficiência destes métodos em termos dos tempos de execução. No entanto, é de referir que este aspeto não é determinante para o problema em causa, uma vez que numa implementação deste sistema os cálculos não precisam de ser feitos em tempo real, podendo ser realizados *a priori*.

O resto do texto está organizado do seguinte modo. No Capítulo 2 são introduzidos os aspetos mais relevantes do sistema de impressão tridimensional. Descreve-se como determinar uma matriz de cobertura e é analisada a projeção de um círculo de luz *laser* no plano onde é feita a impressão. No Capítulo 3 é abordado o problema da localização de emissores, que é modelado como um problema de cobertura de conjuntos com cardinalidade mínima. Apresentam-se uma revisão da literatura sobre este assunto e duas heurísticas para a resolução do problema da localização dos emissores. Em seguida, no Capítulo 4, trata-se o problema da afetação de emissores. O problema é formulado como um programa linear inteiro com dois objetivos, para o qual se considera uma abordagem baseada numa soma pesada dos objetivos e outra baseada numa ordem lexicográfica dos mesmos. O Capítulo 5 é dedicado ao caso de estudo, para o qual se apresentam e discutem os resultados computacionais obtidos pelas metodologias propostas. No último capítulo apresentam-se algumas conclusões e considerações sobre trabalho futuro.

Capítulo 2

Impressão tridimensional bimaterial

Neste capítulo estabelecem-se os pressupostos necessários para a formulação matemática do problema de impressão tridimensional bimaterial e estudam-se alguns aspetos deste problema que serão utilizados no resto do trabalho. Em particular, calcula-se a projeção de um ponto no espaço num plano a partir de um dado emissor. Esta informação é utilizada para descrever um algoritmo para calcular uma matriz de cobertura de emissores, que indique as zonas do plano em que a luz *laser* pode incidir a partir de um determinado emissor. Além disso, analisa-se a distorção da projeção da luz *laser* ao atingir a camada onde se imprime o objeto, dependendo da posição do emissor e da posição a atingir.

2.1 Descrição geral do problema

Pretende-se criar condições para, através do processo de impressão tridimensional denominado estereolitografia, se poder construir um objeto composto por uma grelha de metal, que serve de suporte ao polímero que é solidificado neste processo de impressão. Como se pretende solidificar o polímero em torno da grelha, considera-se que a mesma é colocada no espaço de impressão durante todo o processo, e a impressão tem de ser pensada tendo em conta esse pressuposto.

Há pormenores relativos à impressora que é preciso levar em conta. Supõe-se que o *laser* que emite a luz para solidificar o polímero se encontra fixo na face superior da zona de impressão e que o feixe que emite pode ser orientado em qualquer direção. As camadas do objeto são construídas na face inferior da zona de impressão, pelo que existe uma plataforma móvel, sobre a qual é colocada previamente a estrutura de metal, e que desce à medida que cada camada é acrescentada, tal como descrito na introdução. No que se segue sistemas de espelhos galvanométricos e *laser* são, por vezes, designados emissores.

A hipótese em estudo é a colocação de sistemas de espelhos nas paredes laterais da impressora, para que estes, em funcionamento com o *laser*, possibilitem a construção completa do objeto. Por uma questão de estabilidade do sistema, considera-se que estes sistemas de espelhos se encontram fixos durante todo o processo de impressão, mas os mesmos podem ser orientados para refletir a luz *laser* com os ângulos necessários para atingir uma determinada posição da camada.

Como já explicado, o objeto é dividido em várias camadas que vão sendo sucessivamente adicionadas ao longo da sua impressão. Por sua vez, para definir cada camada e moldar a sua forma é necessário indicar que regiões do polímero líquido precisam ser atingidas por luz *laser*. Para esse

efeito, o espaço onde a camada é construída é dividido em quadrados mais pequenos, denominados *voxels*, de forma a permitir especificar quais devem ser atingidos por luz *laser*. O termo *voxel* é um neologismo, resultante da combinação das palavras *pixel* e volume, muito utilizado em visualização computacional em três dimensões. Em termos gerais *voxel* designa um volume de uma grelha regular num espaço tridimensional. Para mais informações, consultar [13]. A utilização de *voxels* corresponde à discretização do espaço de impressão. Quanto mais, e mais pequenos, forem os *voxels*, mais precisa será a impressão. Em contrapartida, uma discretização mais fina também aumenta a dimensão dos problemas associados, e portanto é necessário um compromisso entre estes dois aspetos.

Uma vez definidos os *voxels* que precisam ser atingidos por luz *laser* e estipuladas as possíveis localizações dos emissores, procede-se à resolução de dois problemas distintos:

- Problema de localização de emissores (PLE): No primeiro problema são selecionadas, de entre as posições admitidas para colocar os emissores, aquelas que permitem minimizar o número de emissores necessários para a impressão completa.
- Problema de afetação de emissores (PAE): Tendo em conta os emissores disponíveis, que incluem o *laser* e os emissores selecionadas no primeiro problema, no segundo problema é necessário determinar quais os *voxels* de cada camada que cada emissor deve atingir.

As condições do problema de impressão tridimensional têm semelhanças com alguns estudos desenvolvidos no contexto de tratamentos por radioterapia, uma técnica utilizada em diversos tipos de cancro. Nestes tratamentos uma zona do corpo do paciente é exposta a um feixe de radiação ionizante que danifica o ADN das células. A terapia por radiação de intensidade modulada é uma forma de radioterapia que permite variar a intensidade da radiação aplicada através de um feixe. Recentemente têm sido investigadas formas de tornar os tratamentos mais eficientes. A este propósito referem-se os trabalhos de revisão de Ehrgott et al. [9, 10]. Apesar de serem inúmeras as diferenças entre o problema da impressão tridimensional e os problemas de terapia por radiação de intensidade modulada, em ambos existe um feixe e um conjunto de *voxels* que devem ser atingidos, pelo que os métodos empregues para este último problema podem servir de inspiração para o primeiro.

2.2 Matriz de cobertura dos emissores

Nesta secção descreve-se um algoritmo para determinar os *voxels* de uma camada que são atingidos por cada possível emissor.

Considera-se um conjunto de m *voxels* em que a luz *laser* precisa de incidir. Considera-se também um conjunto de n possíveis emissores. Para resolver o problema de impressão tridimensional é necessário conhecer que *voxels* são atingidos por cada emissor. Para tal é definida a matriz $A = [a_{ij}]_{i=1,\dots,m; j=1,\dots,n}$, com entradas

$$a_{ij} = \begin{cases} 1 & \text{se o emissor na posição } j \text{ atinge o } \textit{voxel } i \\ 0 & \text{caso contrário} \end{cases}$$

para $i = 1, \dots, m$, $j = 1, \dots, n$, designada matriz de cobertura dos emissores. De seguida explica-se como obter os valores destas entradas.

Supõe-se que são conhecidas:

- as dimensões da zona de impressão e, conseqüentemente, as posições para instalação de emissores,
- a forma da estrutura metálica que irá integrar o objeto (e que se considera instalada antes do início da impressão),
- a forma da parte polimérica do objeto a imprimir.

Como se referiu, o objeto é particionado em *voxels* mais pequenos, de modo uniforme. No que se segue considera-se que cada camada é dividida em $l \times l$ *voxels*. Considera-se o sistema de coordenadas tridimensionais de forma a que a plataforma de impressão, onde é adicionada a nova camada, coincida com o plano $x0y$ e que as coordenadas dos centros dos *voxels* da camada que vai ser adicionada sejam dadas por $(r, s, 0)$, $r = 1, \dots, l$, $s = 1, \dots, l$. A terceira coordenada do sistema diz respeito à altura relativamente à plataforma de impressão, e é necessária como referência para as coordenadas dos emissores e dos pontos usados para definir a grelha metálica.

A forma da grelha de metal é conhecida no início da impressão e considera-se formada por segmentos, cada um definido pelas coordenadas dos seus pontos extremos. Para simplificar a exposição, assume-se que estes são segmentos de reta. Pretende-se verificar que posições da camada a imprimir são atingidas pela luz *laser* ou, de outro modo, que posições da camada não são “sombreadas”. Note-se que o que se passa no eixo $x0y$, e abaixo dele, não é relevante, assim como não o são todos os pontos que se encontram à mesma altura, ou acima, do emissor. Deste modo, para determinar os *voxels* $(r, s, 0)$ atingidos por um emissor na posição $D = (d_1, d_2, d_3)$, é suficiente analisar pontos (x, y, z) dos segmentos que formam a grelha tais que $0 < z < d_3$. Considera-se que a projeção de um segmento fica definida pela projeção dos seus extremos. Suponham-se dados *cont* segmentos nestas condições e denotemos os seus pontos extremos por $U(i)$ e $V(i)$, para $i = 1, \dots, cont$. Será considerado que o segmento formado pela projeção dos dois pontos extremos na camada é dado por um conjunto discreto de pontos.

Sejam $D = (d_1, d_2, d_3)$ a posição do emissor e $U = (u_1, u_2, u_3)$ a posição de um extremo de um segmento metálico. Pretende-se agora saber qual a projeção do ponto U na plataforma, a partir do emissor D . Denote-se o novo ponto por $(Proj_x(U), Proj_y(U))$. O vetor que vai de D para U é dado por $\overrightarrow{(u_1 - d_1, u_2 - d_2, u_3 - d_3)}$. Multiplicando este vetor por $\frac{d_3}{d_3 - u_3}$, obtém-se o vetor

$$\overrightarrow{\left(\frac{d_3}{d_3 - u_3} (u_1 - d_1), \frac{d_3}{d_3 - u_3} (u_2 - d_2), -d_3 \right)},$$

portanto, através da adição desse vetor ao ponto D , conclui-se que

$$Proj_x(U) = d_1 + \frac{d_3}{d_3 - u_3} (u_1 - d_1) \quad \text{e} \quad Proj_y(U) = d_2 + \frac{d_3}{d_3 - u_3} (u_2 - d_2). \quad (2.1)$$

O algoritmo abaixo resulta para os *voxels* de apenas uma camada. Suponha-se que a impressão é feita utilizando p camadas e que, para cada uma, existem m_k *voxels* a atingir, $k = 1, \dots, p$. Obviamente, tem-se $m_1 + m_2 + \dots + m_p = m$. Considere-se uma divisão em blocos da matriz A , de forma a que cada submatriz $A_k \in \{0, 1\}^{m_k \times n}$ contenha as linhas da matriz A correspondentes aos *voxels* da camada

$k, k = 1, \dots, p$. Cada coluna da matriz A_k corresponde aos *voxels* da camada k atingidos por cada um dos $j = 1, \dots, n$ emissores. O Algoritmo 1 resume os passos necessários para determinar a coluna j da matriz A_k . Este algoritmo utiliza uma matriz B como variável auxiliar. As posições da matriz B são inicializadas com o valor 1. Em seguida aquelas que correspondem a zonas “sombreadas” são marcadas com 0.

Algoritmo 1: Cálculo da coluna j da matriz A_k

```

for  $r = 1, \dots, l$  do
  for  $s = 1, \dots, l$  do  $b_{rs} \leftarrow 1$ 
for  $i = 1, \dots, cont$  do
   $(x_1, y_1) \leftarrow (Proj_x(U(i)), Proj_y(U(i)))$ 
   $(x_2, y_2) \leftarrow (Proj_x(V(i)), Proj_y(V(i)))$ 
  for  $(r, s) \in (x_1, y_1)(x_2, y_2)$  do  $b_{rs} \leftarrow 0$ 
for  $i = 1, \dots, m_k$  do
   $(r, s) \leftarrow$  coordenadas do centro do voxel  $i$  a atingir com luz laser
   $a_{ij} \leftarrow b_{rs}$ 

```

2.3 Projeção do *laser*

Nesta secção é analisada a distorção da luz *laser* quando esta atinge a camada polimérica. Em particular são calculadas a área atingida pelo *laser* além do *voxel* e a área do *voxel* que fica por atingir, que serão utilizadas mais tarde para avaliar a qualidade de uma solução.

A direção com que cada emissor atinge cada *voxel* é definida por dois ângulos, α e θ . Sejam $D = (d_1, d_2, d_3)$ a posição de um emissor e $P = (p_1, p_2, p_3)$ a posição de um *voxel* a atingir, ambas representadas na Figura 2.1. O ângulo α é medido no plano xOy , em sentido direto formado entre a semirreta paralela ao eixo do x com sentido positivo e a semirreta com origem em D e que passa em P . Já θ corresponde ao ângulo formado entre o feixe emitido em sentido descendente e o plano horizontal que passa por D . Os dois ângulos são ilustrados na Figura 2.2.

A luz *laser* utilizada no processo de impressão tem a forma de um círculo, mas ao atingir a superfície polimérica tem a forma de uma elipse. Em seguida estuda-se a forma desta elipse e a sua posição relativamente aos *voxels*, considerando-se que r é o raio do círculo da luz *laser*.

Se $d_1 = p_1$ e $d_2 = p_2$, a luz é emitida verticalmente e a elipse degenera num círculo de raio r . Este caso é óbvio e é excluído da análise que se segue.

O comprimento do semieixo menor da elipse, que se denota por b , é sempre $b = r$. Por outro lado, o semieixo maior, denotado por a – ver Figura 2.3, depende das posições de D e P , e está orientado de acordo com o ângulo α . Com base nas coordenadas de D e P , e observando a Figura 2.2, vem para este caso, supondo $d_1 < p_1$ e $d_2 \leq p_2$, que

$$\tan(\alpha) = \frac{p_2 - d_2}{p_1 - d_1} \Leftrightarrow \alpha = \arctan\left(\frac{p_2 - d_2}{p_1 - d_1}\right).$$

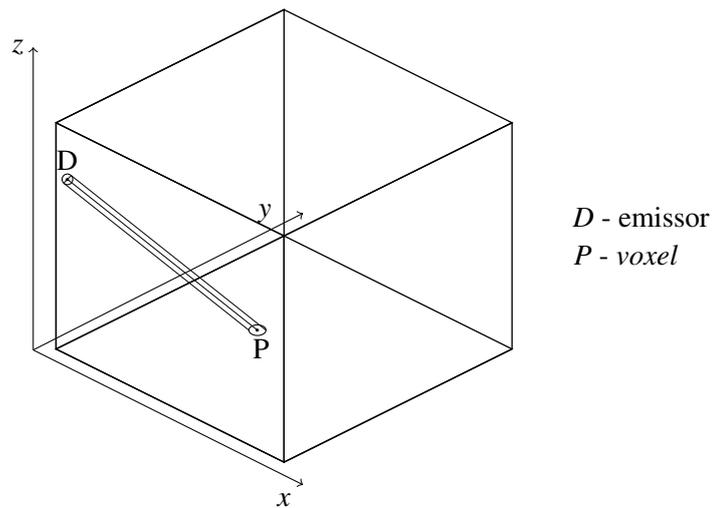
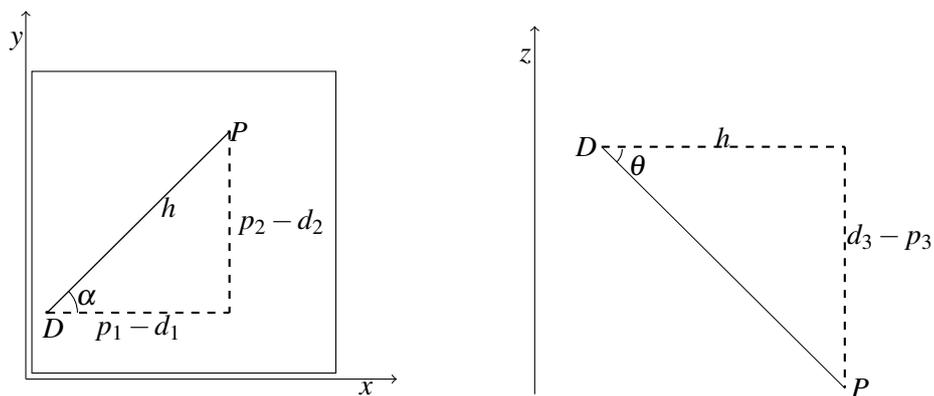


Figura 2.1 Zona de impressão

Figura 2.2 Ângulos α e θ

Assumindo que a função \arctan devolve, em radianos, valores entre $-\frac{\pi}{2}$ e $\frac{\pi}{2}$, pode generalizar-se esta fórmula do seguinte modo

$$\alpha = \begin{cases} \arctan\left(\frac{p_2-d_2}{p_1-d_1}\right) & \text{se } d_1 < p_1 \text{ e } d_2 \leq p_2 \\ 2\pi + \arctan\left(\frac{p_2-d_2}{p_1-d_1}\right) & \text{se } d_1 < p_1 \text{ e } d_2 > p_2 \\ \pi + \arctan\left(\frac{p_2-d_2}{p_1-d_1}\right) & \text{se } d_1 > p_1 \\ \frac{\pi}{2} & \text{se } d_1 = p_1 \text{ e } d_2 < p_2 \\ \frac{3\pi}{2} & \text{se } d_1 = p_1 \text{ e } d_2 > p_2 \end{cases}$$

Para determinar a é necessário começar por calcular a distância assinalada por h na Figura 2.2. Pelo Teorema de Pitágoras resulta que

$$h = \sqrt{(p_1 - d_1)^2 + (p_2 - d_2)^2}.$$

Com base nas Figuras 2.2 e 2.3, conclui-se que

$$\tan(\theta) = \frac{d_3 - p_3}{h} \Leftrightarrow \theta = \arctan\left(\frac{d_3 - p_3}{h}\right),$$

e, portanto, $a = \frac{r}{\sin(\theta)}$.

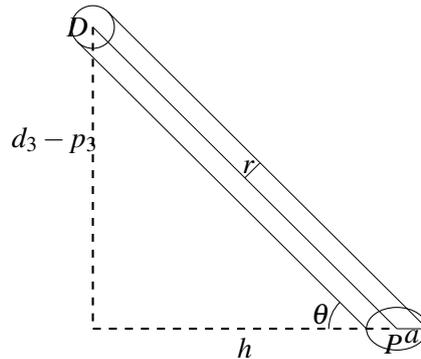


Figura 2.3 Distorção da luz *laser* ao atingir um *voxel*

Assume-se que o lado de cada *voxel* é a unidade de medida, pelo que a área do *voxel* é unitária. Após ser determinado o emissor que atinge o *voxel*, são conhecidos os ângulos α e θ referidos anteriormente, assim como a forma da elipse que o atinge. Note-se que o centro do *laser* é apontado em direção ao centro do *voxel*. Portanto, dependendo da forma da elipse que atinge esse ponto, pode ser atingida uma zona além do *voxel* alvo e parte do *voxel* alvo pode não ser atingida pela luz *laser*. Estas áreas são denotadas por A_{ext} e A_{int} , respetivamente, e são ilustradas na Figura 2.4. Esta informação será utilizada no Capítulo 5, para avaliar as soluções propostas nos Capítulos 3 e 4, pelo que em seguida se faz um estudo das áreas destas duas regiões.

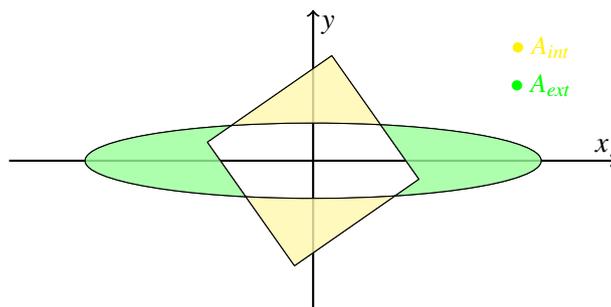


Figura 2.4 Áreas a calcular

Supõe-se que o raio do *laser* utilizado fica completamente contido num *voxel*, isto é, $2r \leq 1$. Caso contrário, o *laser* poderia não ter precisão suficiente para realizar a impressão desejada. Tal como visto acima, a elipse é tal que

$$a = \frac{r}{\sin(\theta)} \text{ e } b = r.$$

O ângulo α , que pode apresentar valores, em radianos, entre 0 e 2π , determina a orientação da elipse relativamente ao *voxel*. Para o presente estudo apenas interessa conhecer o ângulo, denotado por γ , da rotação do *voxel* relativamente ao caso em que as arestas são paralelas aos eixos da elipse. Como este ângulo pode ser medido em sentido direto ou inverso, vamos considerar os dois casos, podendo γ ter valores, em radianos, entre 0 e $\frac{\pi}{4}$. Note-se que os casos em que γ é medido em sentido inverso e em sentido direto são simétricos relativamente a um dos eixos da elipse. Então, γ pode ser obtido a partir de α como

$$\gamma = \begin{cases} \alpha & \text{se } 0 \leq \alpha \leq \frac{\pi}{4} \\ \frac{\pi}{2} - \alpha & \text{se } \frac{\pi}{4} < \alpha \leq \frac{\pi}{2} \\ \alpha - \frac{\pi}{2} & \text{se } \frac{\pi}{2} < \alpha \leq \frac{3\pi}{4} \\ \pi - \alpha & \text{se } \frac{3\pi}{4} < \alpha \leq \pi \\ \alpha - \pi & \text{se } \pi < \alpha \leq \frac{5\pi}{4} \\ \frac{3\pi}{2} - \alpha & \text{se } \frac{5\pi}{4} < \alpha \leq \frac{3\pi}{2} \\ \alpha - \frac{3\pi}{2} & \text{se } \frac{3\pi}{2} < \alpha \leq \frac{7\pi}{4} \\ 2\pi - \alpha & \text{se } \frac{7\pi}{4} < \alpha < 2\pi \end{cases}$$

Por conveniência, e sem perda de generalidade, no que se segue considera-se o sistema de coordenadas alinhado com os eixos da elipse, conforme representado na Figura 2.5. Dependendo da posição do emissor, os lados do *voxel* a atingir podem ser paralelos aos eixos, como se vê na imagem quando $\gamma = 0$, ou sofrer uma rotação igual ao ângulo γ .

Recordando a Figura 2.4, atendendo à simetria, tem-se que A_{ext} é composta por duas áreas iguais que atingem a superfície polimérica em lados opostos do *voxel*. Para calcular A_{int} , determina-se a área da elipse interior ao *voxel* e subtrai-se este valor à área do *voxel*. Portanto, para determinar A_{ext} e A_{int} basta calcular uma das áreas que é atingida fora do *voxel* de um dos lados. Sem perda de generalidade, calcula-se a área do lado direito.

É importante notar que a elipse pode estar totalmente contida no *voxel*. Nesse caso $A_{ext} = 0$ e apenas será determinada A_{int} , subtraindo a área total da elipse à área do *voxel*. Tanto para o cálculo das áreas pretendidas, como para verificar se a elipse se encontra circunscrita no *voxel*, é necessário verificar se a elipse intersesta as arestas assinaladas a verde e a vermelho na Figura 2.5. No caso de existirem, os pontos de interseção serão denotados por $P_1 = (x_1, y_1)$ e $P_2 = (x_2, y_2)$.

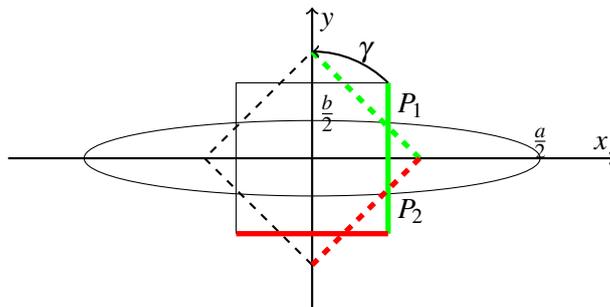


Figura 2.5 Projeção da luz *laser* num *voxel*

A equação reduzida da elipse é dada por

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \quad (2.2)$$

e a equação reduzida da reta que contém a aresta verde, para $\gamma > 0$, é dada por

$$y = -\frac{1}{\tan(\gamma)}x + \frac{0.5}{\sin(\gamma)}. \quad (2.3)$$

A equação reduzida da reta que contém a aresta vermelha, perpendicular à anterior, é dada por

$$y = \tan(\gamma)x - \frac{0.5}{\cos(\gamma)}. \quad (2.4)$$

Se $\gamma = 0$, a elipse apenas pode intersestar a aresta a verde, e conclui-se facilmente que tal só acontece se o comprimento do eixo maior da elipse exceder o comprimento do lado do *voxel*, isto é, se $2a \geq 1$. Neste caso, tem-se $x = 0.5$, donde, isolando y na equação (2.2),

$$P_1 = \left(0.5, \frac{b}{a}\sqrt{a^2 - 0.5^2}\right)$$

e

$$P_2 = \left(0.5, -\frac{b}{a}\sqrt{a^2 - 0.5^2}\right).$$

Quando $\gamma > 0$, se o *voxel* e a elipse se intersestam, pelo menos uma das interseções será com a aresta verde, pelo que se começa por verificar esta situação. Combinando as equações (2.2) e (2.3), conclui-se que tal apenas acontece se

$$\frac{1}{\tan^2(\gamma)\sin^2(\gamma)} - 4\left(\frac{1}{\tan^2(\gamma)} + \frac{a^2}{b^2}\right)\left(\frac{0.25}{\sin^2(\gamma)} - b^2\right) > 0.$$

Se esta condição não for satisfeita, a elipse está completamente inscrita no *voxel*. Caso contrário, o ponto $P_1 = (x_1, y_1)$ corresponde à solução desta com a menor abcissa, pelo que

$$x_1 = \frac{\frac{1}{\tan(\gamma)\sin(\gamma)} - \sqrt{\frac{1}{\tan^2(\gamma)\sin^2(\gamma)} - 4\left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2}\right)\left(\frac{0.25}{\sin^2(\gamma)} - b^2\right)}}{2\left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2}\right)}$$

e

$$y_1 = -\frac{1}{\tan(\gamma)}x_1 + \frac{0.5}{\sin(\gamma)}.$$

Se a elipse intersesta o *voxel* é necessário calcular também o ponto P_2 . Consideram-se dois casos:

- a elipse intersesta a aresta a verde,
- ou a elipse intersesta a aresta a vermelho.

Uma forma de determinar com qual das duas arestas há interseção é verificar se a maior abcissa dos pontos da interseção das curvas (2.2) e (2.3) é superior à abcissa do vértice formado pelas arestas

verde e vermelha, $V = (x_3, y_3)$. Igualando as equações (2.3) e (2.4), obtém-se

$$x_3 = \frac{1 + \tan(\gamma)}{2 \cos(\gamma)(\tan^2(\gamma) + 1)}. \quad (2.5)$$

Portanto, se

$$\frac{\frac{1}{\tan(\gamma) \sin(\gamma)} - \sqrt{\frac{1}{\tan^2(\gamma) \sin^2(\gamma)} - 4 \left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2} \right) \left(\frac{0.25}{\sin^2(\gamma)} - b^2 \right)}}{2 \left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2} \right)} \leq \frac{1 + \tan(\gamma)}{2 \cos(\gamma)(\tan^2(\gamma) + 1)},$$

então P_2 é ponto de interseção da elipse com a aresta a verde. Caso contrário, P_2 é comum à elipse e à aresta a vermelho. No primeiro caso vem que $P_2 = (x_2, y_2)$ é tal que

$$x_2 = \frac{\frac{1}{\tan(\gamma) \sin(\gamma)} + \sqrt{\frac{1}{\tan^2(\gamma) \sin^2(\gamma)} - 4 \left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2} \right) \left(\frac{0.25}{\sin^2(\gamma)} - b^2 \right)}}{2 \left(\frac{1}{\tan^2(\gamma)} + \frac{b^2}{a^2} \right)}$$

e

$$y_2 = -\frac{1}{\tan(\gamma)} x_2 + \frac{0.5}{\sin(\gamma)}.$$

No segundo caso, vem que

$$x_2 = \frac{\frac{\tan(\gamma)}{\cos(\gamma)} - \sqrt{\frac{\tan^2(\gamma)}{\cos^2(\gamma)} - 4 \left(\tan^2(\gamma) + \frac{b^2}{a^2} \right) \left(\frac{0.25}{\cos^2(\gamma)} - b^2 \right)}}{2 \left(\tan^2(\gamma) + \frac{b^2}{a^2} \right)}$$

e

$$y_2 = \tan(\gamma) x_2 - \frac{0.5}{\cos(\gamma)}.$$

Após a análise das interseções, e sabendo que estas existem, ou seja, que a elipse ultrapassa o quadrado, começa por se calcular a área da secção da elipse indicada a azul na Figura 2.6, denotada por A_1 . Os ângulos β_1 e β_2 assinalados na imagem são obtidos a partir das coordenadas de P_1 e P_2 da seguinte forma

$$\beta_1 = \arctan\left(\frac{y_1}{x_1}\right) \text{ e } \beta_2 = \arctan\left(\frac{y_2}{x_2}\right).$$

Posto isto, vem que

$$A_1 = \frac{ab}{2} \left(\arctan\left(\frac{a}{b} \tan(\beta_1)\right) - \arctan\left(\frac{a}{b} \tan(\beta_2)\right) \right).$$

À área A_1 é agora preciso subtrair aquela que faz parte do quadrado, denotada por A_2 . Para o cálculo desta área analisa-se duas situações: a elipse intersesta apenas uma aresta do *voxel* – Figura 2.7a, ou a elipse intersesta duas arestas distintas – Figura 2.7b. Nas duas figuras a área a calcular encontra-se representada a vermelho. No primeiro caso

$$A_2 = \frac{|x_2 y_1 - x_1 y_2|}{2},$$

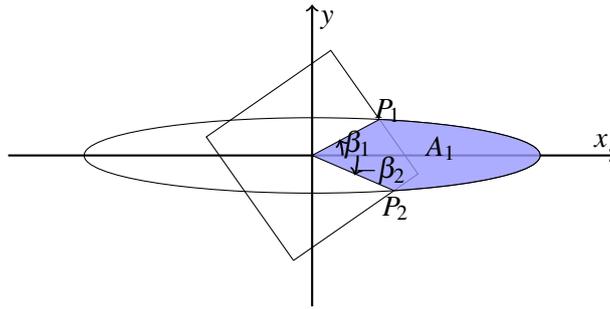


Figura 2.6 Área da secção da elipse

e no segundo

$$A_2 = \frac{|x_3y_1 - x_1y_3|}{2} + \frac{|x_2y_3 - x_3y_2|}{2}.$$

Substituindo (2.5) na equação de uma das arestas, por exemplo a aresta com equação (2.4), vem

$$y_3 = \tan(\gamma)x_3 - \frac{0.5}{\cos(\gamma)}.$$

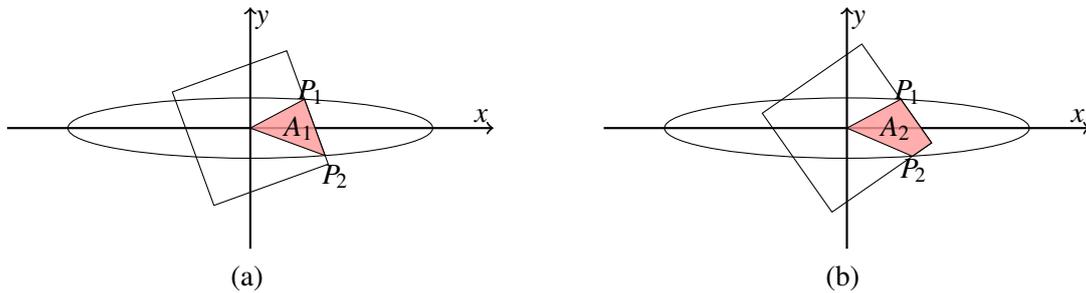


Figura 2.7 Área da secção da elipse dentro do voxel

Concluída a análise de todas as situações, resta referir que a área da elipse é dada por $A = \pi ab$. Juntando todos os dados, conclui-se que:

1. se a elipse está circunscrita no voxel,

$$A_{ext} = 0 \quad \text{e} \quad A_{int} = 1 - A,$$

2. se a elipse não se limita ao voxel,

$$A_{ext} = 2(A_1 - A_2) \quad \text{e} \quad A_{int} = 1 - (A - A_{ext}) = 1 - \pi ab + 2(A_1 - A_2).$$

Capítulo 3

Problema da localização de emissores

Neste capítulo é apresentado um modelo para resolver o problema da localização de emissores. Como se verá, o modelo é um programa linear com variáveis binárias, que decorre da interpretação do problema como um problema de cobertura de conjuntos. Desta forma, é descrito o problema de cobertura de conjuntos geral, e é apresentada uma revisão dos principais métodos conhecidos para a sua resolução. Tendo por base esses métodos, são apresentados dois métodos heurísticos gulosos para o PLE, tendo em conta as suas especificidades.

3.1 Modelo do problema

Supõe-se que existem n possíveis emissores de luz, sendo uma delas o *laser*, suposto fixo, e as restantes $n - 1$ sistemas de espelhos galvanométricos, cujas posições se pretende selecionar. Considera-se também que existem m *voxels* do objeto a atingir.

Tal como em [11, 17], este problema pode ser formulado como um problema de cobertura. Dizemos que o emissor j cobre o *voxel* i se é capaz de o atingir com luz *laser*, $j = 1, \dots, n$, $i = 1, \dots, m$. Para que o polímero num dado *voxel* solidifique, é necessário que pelo menos um emissor o cubra. Para a resolução deste problema utiliza-se a matriz de cobertura dos emissores, $A \in \{0, 1\}^{m \times n}$, definida no Capítulo 2.

Pretende-se minimizar o número de emissores necessários para os m *voxels* do objeto serem atingidos. Então, definam-se as variáveis de decisão, x_j , como sendo binárias e tais que

$$x_j = \begin{cases} 1 & \text{se o emissor na posição } j \text{ deve ser instalado} \\ 0 & \text{caso contrário} \end{cases}$$

para $j = 1, \dots, n$. Nestas condições, o número total de emissores a utilizar, e que se quer minimizar, é dado por

$$\sum_{j=2}^n x_j$$

Considera-se que o *laser* é sempre utilizado diretamente como fonte de emissão, pelo que $x_1 = 1$ é uma restrição do problema. Além disso, uma solução é admissível se todos os *voxels* $i = 1, \dots, m$

puderem ser atingidos por algum emissor, isto é, se

$$\sum_{j=1}^n a_{ij}x_j \geq 1, \quad i = 1, \dots, m,$$

pelo que estas são igualmente restrições a incluir no problema. Então, o problema linear binário a resolver é

$$\begin{aligned} \min \quad & \sum_{j=2}^n x_j \\ \text{sujeito a} \quad & Ax \geq 1 \\ & x_1 = 1 \\ & x \in \{0, 1\}^n \end{aligned} \quad (3.1)$$

O valor ótimo do problema (3.1) é o número de sistemas de espelhos necessários para completar a impressão. A solução ótima permite conhecer as posições em que esses espelhos devem ser instalados na impressora, uma vez que estes são indicados pelas variáveis x_j tais que $x_j = 1$.

Como mencionado acima, este é um problema de cobertura de conjuntos, um problema clássico de otimização combinatória, com o objetivo de encontrar o subconjunto de colunas com custo mínimo que cobre todas as linhas de uma matriz 0-1. Por essa razão a próxima secção é dedicada ao problema de cobertura de conjuntos no caso geral.

3.2 Problema de cobertura de conjuntos

O problema de cobertura de conjuntos, ou simplesmente problema de cobertura, é uma questão combinatória clássica que pode ser formulada como um programa linear com variáveis inteiras (e binárias). Dado um universo U e uma coleção de subconjuntos de U , $S \subseteq \mathcal{P}(U)$, pretende-se determinar conjuntos $S_1, S_2, \dots, S_k \in S$, de modo a que todos os elementos de U pertençam a pelo menos um desses conjuntos, isto é, de modo a que

$$U = \cup_{j=1}^k S_j.$$

Neste caso os conjuntos S_1, S_2, \dots, S_k dizem-se uma cobertura de U . Adicionalmente pode associar-se um custo c_j a cada conjunto $S_j \in S$, procurando-se então minimizar o custo total dos conjuntos escolhidos. Resumindo, pretende-se

$$\begin{aligned} \min \quad & \sum_{j=1}^k c_j \\ \text{sujeito a} \quad & \cup_{j=1}^k S_j = U \\ & S_j \in S, \quad j = 1, \dots, k \end{aligned}$$

Quando $c_j = 1$ para todo $j = 1, \dots, k$, o problema diz-se problema de cobertura com custos unitários, ou problema de cobertura com cardinalidade mínima.

Por exemplo, considerando o universo $U = \{1, 2, 3\}$, a coleção $S = \mathcal{P}(U) - U$ e custos unitários,

$$C_1 = \{\{1\}, \{2\}, \{3\}\},$$

$$C_2 = \{\{1\}, \{1, 2\}, \{3\}\},$$

$$C_3 = \{\{1,2\}, \{3\}\},$$

$$C_4 = \{\{1,2\}, \{2,3\}\},$$

são coberturas de U , sendo C_3 e C_4 coberturas com cardinalidade mínima.

O problema de cobertura de conjuntos pode ser formulado como um programa linear com variáveis inteiras. Seja $A = [a_{ij}] \in \{0, 1\}^{m \times n}$, designada matriz de cobertura. Diz-se que a coluna j de A cobre a linha i de A se e só se $a_{ij} = 1$, $i = 1, \dots, m$, $j = 1, \dots, n$. Caso contrário, $a_{ij} = 0$. A utilização da coluna j acarreta um custo adicional $c_j \in \mathbb{R}^+$, $j = 1, \dots, n$. O objetivo do problema de cobertura de conjuntos é encontrar o subconjunto de colunas de A , com custo mínimo, de forma a que cada linha seja coberta por pelo menos uma coluna. Considerando as variáveis de decisão

$$x_j = \begin{cases} 1 & \text{se a coluna } j \text{ pertence à solução} \\ 0 & \text{caso contrário} \end{cases}$$

o problema de cobertura de conjuntos pode ser formulado como

$$\begin{aligned} \min & \quad \sum_{j=1}^n c_j x_j \\ \text{sujeito a} & \quad \sum_{j=1}^n a_{ij} x_j \geq 1, i = 1, \dots, m \\ & \quad x_j \in \{0, 1\}, j = 1, \dots, n \end{aligned} \quad (3.2)$$

Naturalmente o problema (3.2) só é admissível se cada linha puder ser coberta por pelo menos uma coluna.

Para o exemplo anterior a matriz A , tem dimensão 3×7 e elementos

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

onde as soluções correspondentes a C_3 e C_4 são

$$x' = (0, 0, 0, 1, 1, 0, 0) \text{ e}$$

$$x'' = (0, 0, 0, 0, 1, 0, 1),$$

respetivamente. Como referido acima, x' e x'' são soluções ótimas do problema (3.2) se os custos forem unitários.

Os problemas de cobertura de conjuntos têm inúmeras aplicações, tais como planeamento estratégico, planeamento de produção, problemas de localização de instalações, bem como muitos outros problemas de agendamento e de roteamento.

Em 1972 Karp [15] mostrou que o problema de cobertura é NP-completo. Isto significa que, apesar de existirem algoritmos para resolvê-lo de forma exata, estes têm aplicabilidade e eficiência reduzidas. Por este motivo têm igualmente sido propostos vários métodos para encontrar soluções próximas da ótima para problemas de cobertura de conjuntos de grandes dimensões em tempo razoável. Em 2000 Caprara, Toth e Fischetti [6] publicaram um trabalho de revisão deste problema. De seguida listam-se alguns dos métodos, heurísticos ou exatos, propostos para o resolver.

A maioria das heurísticas para o problema de cobertura de conjuntos segue um modelo guloso, em que cada coluna j é associada a um valor σ_j , que dá uma indicação do interesse em incluir a coluna j na solução. A escolha das colunas é feita por ordem decrescente de σ_j e o processo termina quando for encontrada uma solução admissível, isto é, uma cobertura. Este método foi introduzido por Chvátal [7], considerando $\sigma_j = c_j/|M_j|$, onde M_j representa o conjunto de linhas cobertas pela coluna j . No Teorema 3.2.1 enuncia-se um resultado apresentado naquele trabalho, que fornece um limite superior para o erro cometido na aproximação calculada por este método heurístico.

Teorema 3.2.1 ([7]). *Seja $H_n = \sum_{i=1}^n \frac{1}{i} \approx \log n$. O algoritmo guloso devolve uma cobertura de conjuntos com custo no máximo H_n vezes superior ao da cobertura de conjuntos com custo mínimo.*

Desde a introdução desta heurística foram propostas outras versões, em que variam os critérios para selecionar as colunas, normalmente dependendo do seu custo, do número de linhas que cobrem, ou por processos mais complexos utilizando os custos de multiplicadores de Lagrange, entre outros. Dos trabalhos publicados sobre este assunto podem destacar-se Vasko e Wilson [18, 20], que introduziram uma versão aleatória da heurística gulosa. Mais tarde, Beasley [3] propôs uma heurística baseada em relaxação Lagrangeana e otimização por subgradientes, que gera uma cobertura após cada iteração. Recentemente, Vasko, Lu e Zyma [19] apresentaram uma comparação computacional entre alguns destes métodos.

Os métodos exatos para o problema de cobertura com maior destaque têm sido métodos *branch and bound*. Podem obter-se limites inferiores para o problema de cobertura de conjuntos através da relaxação linear, substituindo as restrições de integralidade por

$$0 \leq x_j \leq 1, j = 1, \dots, n.$$

Além disso, qualquer solução admissível determinada por um método heurístico, como os mencionados acima, fornece um limite superior. A divisão dos problemas pode ser implementada escolhendo variáveis fracionárias que possam surgir e fixando o valor de x_j como 0 ou 1, o que corresponde a excluir ou a incluir a coluna j na cobertura, respetivamente. No entanto, os pacotes de *software* de programação matemática têm desenvolvido implementações otimizadas e muito eficientes de métodos *branch and bound*, e portanto desenvolver um algoritmo exato capaz de competir com estes pacotes está longe de ser uma tarefa fácil.

Citamos, sem detalhes, alguns trabalhos sobre métodos exatos para o problema de cobertura de conjuntos. Balas e Ho [2] desenvolveram técnicas de relaxação Lagrangeana, utilizando otimização por subgradientes. Com base nesse trabalho Beasley [4] desenvolveu um algoritmo exato, resolvendo subproblemas de programação linear e utilizando novas regras para fixar variáveis. Mais recentemente, em 1996, Balas e Carrera [1] apresentaram um algoritmo *branch and bound*, baseado num procedimento que mantém dois esquemas de divisão da árvore de enumeração baseados na informação dual associada aos multiplicadores de Lagrange. Este método mostrou-se eficiente, relativamente aos existentes.

3.3 Métodos heurísticos para o problema de localização de emissores

O PLE, (3.1), é um caso particular do problema (3.2), em que se consideram apenas custos unitários. Além disso, a primeira coluna da matriz, correspondente ao emissor *laser* no topo da impressora, é forçada a ser utilizada, motivo pelo qual a minimização é feita apenas para as restantes colunas. A resolução deste problema pode ser realizada recorrendo a *software* específico, com algumas limitações. No que se segue descrevemos alguns métodos heurísticos, que podem ser uma alternativa a esses programas.

Para obter soluções do PLE considera-se o problema dividido em três fases:

1. redução do problema,
2. aplicação do método heurístico,
3. eliminação de redundância,

que se descrevem em seguida. Para simplificar exposição, denota-se por $M = \{1, \dots, m\}$ o conjunto das linhas e por $N = \{1, \dots, n\}$ o conjunto das colunas. Além disso, $N_i = \{j \in N : a_{ij} = 1\}$ representa o conjunto das colunas que cobrem a linha $i = 1, \dots, m$, e, de forma análoga, $M_j = \{i \in M : a_{ij} = 1\}$ representa o conjunto das linhas cobertas pela coluna $j \in N$.

Redução do problema Tal como já observado, a utilização da primeira coluna é uma imposição do problema. Desta forma, é imediato que para reduzir a dimensão do problema pode ser removida a primeira coluna. Do mesmo modo, podem ser eliminadas todas as linhas que são cobertas por aquela. Isto é, o problema pode ser reduzido fazendo

$$N = N - \{1\} \text{ e } M = M - M_1.$$

Outras reduções podem ser obtidas aplicando um determinado conjunto de regras. Para exemplificar a aplicação destas regras, considere-se a matriz

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

com conjunto das linhas e das colunas dados por $M = \{1, 2, 3, 4\}$ e $N = \{1, 2, 3, 4, 5\}$, respetivamente. À matriz A vão sendo aplicadas as regras apresentadas sucessivamente, ou seja, os conjuntos M e N vão ser atualizados à medida que as regras vão sendo aplicadas. As regras consideradas são as seguintes:

- R1 Se para alguma coluna $j^* \in N$ se tem $M_{j^*} = \emptyset$, então esta pode ser eliminada visto que não cobre nenhuma linha, ou seja, faz-se $N = N - \{j^*\}$.

Por exemplo, a primeira coluna da matriz A não cobre nenhuma linha, pelo que esta pode ser removida. Atualizando o conjunto das colunas vem $N = \{2, 3, 4, 5\}$.

R2 Se existir alguma linha $i^* \in M$ tal que $i^* \in M_{j^*}$ e $i^* \notin M_k$ para todo $k \neq j^*$, então j^* deve estar em todas as soluções, e as linhas cobertas por M_{j^*} podem ser eliminadas. A redução é obtida fazendo $N = N - \{j^*\}$ e $M = M - M_{j^*}$.

Por exemplo, observando a terceira linha de A tem-se que $3 \in M$, $3 \in M_4$ e $3 \notin M_2, M_3, M_5$, pelo que a quarta coluna é a única que cobre a terceira linha. Dessa forma, a quarta coluna tem que fazer parte da solução, isto é, $x_4 = 1$, e pode se proceder à atualização $M = \{1, 2, 4\}$ e $N = \{2, 3, 5\}$.

R3 Se existirem linhas $i^*, r^* \in M$ tais que $N_{i^*} \subseteq N_{r^*}$, então a linha r^* pode ser eliminada, uma vez que ao cobrir a linha i^* está também estará garantidamente coberta. Neste caso, diz-se que a linha i^* domina a linha r^* e faz-se $M = M - \{r^*\}$.

Por exemplo, observando as duas primeiras linhas tem-se que $N_1 \subseteq N_2$, ou seja, a segunda linha deve ser eliminada, pois ao cobrir a primeira linha a segunda estará automaticamente coberta. Como consequência vem $M = \{1, 4\}$.

R4 Se existirem colunas $p^*, q^* \in N$ tais que $N_{p^*} \supseteq N_{q^*}$, então a coluna q^* deve ser eliminada visto que todo o conjunto M_{p^*} coberto pela coluna p^* também cobre o conjunto M_{q^*} . Neste caso, diz-se que a coluna p^* domina a coluna q^* e faz-se $N = N - \{q^*\}$.

Por exemplo, verifica-se que $N_2 \supseteq N_5$, pelo que a quinta coluna deve ser eliminada, pois a segunda coluna cobre as linhas que a quinta coluna cobre, portanto atualiza-se $N = \{2, 3\}$.

Após a aplicação de cada regra uma vez à matriz A dada acima, verifica-se uma redução em relação aos conjuntos M e N de três colunas e duas linhas.

É expeável que as regras R1 e R2 sejam menos exigentes, em termos de implementação, do que as regras R3 e R4. As regras R1 e R2 implicam uma análise de todos os elementos da matriz A , portanto têm um número de operações com ordem de complexidade linear $O(mn)$, respetivamente. Já a regra R3 requer que, cada linha i seja comparadas com as seguintes, $i + 1, \dots, m$, e cada comparação implica a verificação de, no máximo, n valores. Então, as operações realizadas são

$$n \sum_{i=1}^m (m-i) = n \sum_{i=0}^{m-1} i = nm \frac{m-1}{2} = \frac{n(m^2 - m)}{2}$$

e, no pior caso, o tempo tem ordem de complexidade $O(nm^2)$. Analogamente se mostra que a regra R4 pode ser implementada com tempo de $O(mn^2)$.

Método heurístico Os métodos heurísticos gulosos referidos na secção anterior aplicam-se ao caso geral de problemas de cobertura de conjuntos. Em seguida consideram-se custos unitários (como acontece no PLE) e apresentam-se dois métodos heurísticos gulosos, um baseado na informação obtida de cada coluna da matriz do problema e o outro baseado na informação obtida a partir das linhas e colunas da matriz do problema.

A ideia da primeira heurística é começar por seleccionar as colunas que cobrem mais linhas ainda por cobrir. Como variáveis auxiliares utilizam-se:

- S , um conjunto que armazena os índices das colunas que vão sendo selecionadas, inicializado como o conjunto vazio,
- R , um conjunto que armazena os índices das linhas por cobrir, inicializado como M .

As linhas cobertas por uma coluna selecionada são removidas de R e o processo termina quando já não existirem linhas por cobrir. Nessa altura S contém uma cobertura. Este método é resumido no Algoritmo 2.

Algoritmo 2: Heurística baseada em colunas

```

 $R \leftarrow M$ 
 $S \leftarrow \emptyset$ 
while  $R \neq \emptyset$  do
  for  $j \in N$  do  $K_j \leftarrow |M_j \cap R|$ 
   $j^* \leftarrow \operatorname{argmax}\{K_j\}$ 
   $R \leftarrow R \setminus M_{j^*}$ 
   $S \leftarrow S \cup \{j^*\}$ 

```

No pior dos casos o Algoritmo 2 executa m iterações do ciclo **while**, enquanto que o ciclo **for** tem n iterações, pelo que se tem tempo de $O(mn)$.

Para exemplificar, considere-se a aplicação do Algoritmo 2 às matrizes

$$B = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Para a matriz B , o conjunto das linhas e colunas são $M_B = \{1, 2, 3, 4, 5, 6\}$ e $N_B = \{1, 2, 3, 4, 5\}$. Já para a matriz C , estes conjuntos são $M_C = \{1, 2, 3, 4, 5, 6\}$ e $N_C = \{1, 2, 3, 4, 5, 6, 7\}$. Note-se que $S_B = \{3, 4\}$ é uma cobertura ótima para as linhas da matriz B e que $S_C = \{2, 3\}$ é uma cobertura ótima para as linhas da matriz C .

Comece-se por aplicar o algoritmo à matriz B . Os conjuntos iniciais são $R = \{1, 2, 3, 4, 5, 6\}$ e $S = \emptyset$. Vejamos o que acontece em cada iteração do ciclo **while**:

1. Inicialmente $K_1 = 4$, $K_2 = 3$, $K_3 = 3$, $K_4 = 3$ e $K_5 = 2$. Portanto é selecionada a primeira coluna e os conjuntos são atualizados, $R = \{4, 5\}$ e $S = \{1\}$.
2. Agora $K_2 = 1$, $K_3 = 1$, $K_4 = 1$, e $K_5 = 0$. Considere-se que é selecionada a segunda coluna, apesar de haver um empate, e então, $R = \{5\}$ e $S = \{1, 2\}$.
3. Em seguida $K_3 = 0$, $K_4 = 1$ e $K_5 = 0$. Resta escolher a quarta coluna, obtendo $R = \emptyset$ e $S = \{1, 2, 4\}$.

Como o conjunto R não tem mais linhas, o algoritmo termina e a solução encontrada é $S = \{1, 2, 4\}$, que não tem cardinalidade mínima.

Considere-se agora a matriz C . Inicialmente, tem-se $R = \{1, 2, 3, 4, 5, 6\}$ e $S = \emptyset$. Em cada iteração do ciclo **while** vem:

1. Primeiro $K_1 = 3, K_2 = 2, K_3 = 4, K_4 = 3, K_5 = 3, K_6 = 3$ e $K_7 = 3$, logo é selecionada a terceira coluna, portanto $R = \{1, 2\}$ e $S = \{3\}$.
2. Em seguida, $K_1 = 1, K_2 = 2, K_4 = 1, K_5 = 1, K_6 = 0$ e $K_7 = 1$. Então é selecionada a segunda coluna, fazendo-se $R = \emptyset$ e $S = \{2, 3\}$.

Neste caso o algoritmo termina com a solução $S = \{2, 3\}$, que é ótima.

O critério utilizado pela segunda heurística privilegia as linhas. A ideia agora é identificar a linha coberta por menos colunas, uma vez que esta é a mais difícil de cobrir. De entre todas as colunas que cobrem a linha escolhida, prefere-se, tal como no Algoritmo 2, aquela que cobre mais linhas. Este método utiliza os mesmos conjuntos R e S definidos acima e é resumido no Algoritmo 3.

Algoritmo 3: Heurística baseada em linhas e colunas

```

R ← M
S ← ∅
while R ≠ ∅ do
  for i ∈ R do Li ← ∑j ∈ Li aij
  i* ← argmin{Ni}
  for j ∈ Ni* do Kj ← |Mj ∩ R|
  j* ← argmax{Kj}
  R ← R \ Mj*
  S ← S ∪ {j*}

```

Considerando novamente o pior dos casos, o Algoritmo 3 executa m iterações do ciclo **while**. Para cada uma são executadas m iterações do primeiro ciclo **for** e, no máximo, n iterações do segundo. Então, o método tem de $O(m(m+n))$.

Aplique-se também o Algoritmo 3 às matrizes B e C . Para a matriz B , vem $R = \{1, 2, 3, 4, 5, 6\}$ e $S = \emptyset$. Em cada iteração do ciclo **while** tem-se:

1. Inicialmente $L_1 = 3, L_2 = 3, L_3 = 3, L_4 = 2, L_5 = 1$ e $L_6 = 3$, donde é escolhida a quinta linha e a única coluna que pode ser escolhida é a quarta. Após a atualização, $R = \{2, 4, 6\}$ e $S = \{4\}$.
2. A seguir $L_2 = 3, L_4 = 2$ e $L_6 = 3$. Então é escolhida a quarta linha, para a qual $K_2 = 2$ e $K_3 = 3$, donde é selecionada a terceira coluna. Agora, $R = \emptyset$ e $S = \{3, 4\}$.

O algoritmo termina e devolve a $S = \{3, 4\}$, que é solução ótima.

Para a matriz C , começa-se novamente com $R = \{1, 2, 3, 4, 5, 6\}$ e $S = \emptyset$. Então:

1. Primeiro $L_1 = 2, L_2 = 4, L_3 = 4, L_4 = 4, L_5 = 3$ e $L_6 = 4$, portanto é escolhida a primeira linha, para a qual $K_1 = 3$ e $K_2 = 2$. Então seleciona-se a primeira coluna, e faz-se $R = \{2, 5, 6\}$ e $S = \{1\}$.

2. Agora $L_2 = 4$, $L_5 = 3$ e $L_6 = 4$, donde se escolhe a quinta linha, para a qual $K_3 = 2$, $K_6 = 2$ e $K_7 = 2$. Há um empate entre colunas, no entanto a escolha não interfere no número de colunas final. Suponha-se que se escolhe a terceira coluna. Então, $R = \{2\}$ e $S = \{1, 3\}$.
3. Resta escolher a segunda linha. Então, $K_2 = 1$, $K_4 = 1$, $K_5 = 1$ e $K_7 = 1$. Novamente a coluna escolhida também não afeta o número de colunas do resultado final. Supondo que é escolhida a segunda, obtém-se $R = \emptyset$ e $S = \{1, 2, 3\}$.

O algoritmo termina, tendo-se obtido a solução $S = \{1, 2, 3\}$ que não é ótima.

Estes exemplos permitem verificar que qualquer dos dois métodos, Algoritmo 2 e Algoritmo 3, permitem obter uma solução ótima, no entanto isso não é garantido.

Eliminação de redundância O conjunto de colunas encontrado pelos Algoritmos 2 e 3, representado pela variável S , é uma cobertura, no entanto esta pode conter colunas redundantes. O método apresentado a seguir procura reduzir esta redundância, através da eliminação de algumas colunas. Para tal, procuram-se colunas $j \in S$ tais que $S \setminus \{j\}$ ainda seja uma cobertura. Se isto se verificar, a coluna j é redundante e $S \setminus \{j\}$ continua a ser uma cobertura. Neste caso S é atualizado, caso contrário mantém-se. Este método é resumido no Algoritmo 4.

Algoritmo 4: Eliminação de colunas redundantes

```

for  $j \in S$  do
  if  $S \setminus \{j\}$  é cobertura then  $S \leftarrow S \setminus \{j\}$ 

```

O ciclo **for** do Algoritmo 4 é executado $|S| \leq n$ vezes e verificar se $S \setminus \{j\}$ é uma cobertura pode ser conseguido, com no máximo, $O(m)$ operações. Conclui-se então que Algoritmo 4 tem tempo de $O(mn)$.

A solução encontrada pelo Algoritmo 2, quando aplicado à matriz B foi $S = \{1, 2, 4\}$. Este conjunto deixa de ser uma cobertura se lhe for retirada alguma coluna, portanto o Algoritmo 4 não altera S .

O outro exemplo em que não foi encontrada a solução ótima foi o Algoritmo 3, quando aplicado à matriz C . Neste caso a solução encontrada foi $S = \{1, 2, 3\}$. Agora, eliminando a primeira coluna deste conjunto continua a ter-se uma cobertura, portanto o Algoritmo 4 permite obter a cobertura $S = S \setminus \{1\} = \{2, 3\}$, que é uma cobertura com cardinalidade mínima das linhas de C .

Capítulo 4

Problema da afetação de emissores

O capítulo anterior foi dedicado ao estudo do PLE, que foca a localização dos emissores de luz *laser* na zona de impressão, por forma a garantir que todos os *voxels* de um determinado objeto a imprimir possam ser atingidos por algum deles. No que se segue considera-se que foi resolvido o PLE e, como tal, que um conjunto de emissores foi instalado nas posições determinadas pela solução do PLE. Nestas condições pretende-se agora definir a sequência de emissores que devem ser ativados para atingir os vários *voxels* do objeto a imprimir. De outro modo, é necessário selecionar qual o emissor que deve ser acionado para incidir em cada *voxel*. Como referido anteriormente, denomina-se este problema por problema da afetação de emissores PAE. Apresenta-se uma formulação deste problema como um programa linear com variáveis binária e dois objetivos. São propostas duas abordagens para este problema: uma baseada no método de soma pesada e outra baseada num critério lexicográfico.

4.1 Formulação do problema

Suponha-se que, após a resolução do PLE, se concluiu serem necessários n_2 emissores de luz *laser* para a impressão completa de um objeto. Um deles é um feixe *laser*, instalado no topo da impressora, e os restantes $n_2 - 1$ são sistemas de espelhos galvanométricos, com posições fixas, determinadas pela solução do problema. Considere-se ainda que a impressão é realizada em p camadas, cada uma com m_k *voxels* a atingir, $k = 1, \dots, p$.

Uma vez que a posição dos emissores é conhecida, e fixa à partida, a escolha do emissor para atingir um determinado *voxel* pressupõe a escolha do ângulo com que o *laser* o atinge. Conforme estudado no Capítulo 2, este ângulo determina a forma como a luz *laser* incide na camada de impressão, conduzindo os ângulos mais pequenos a uma maior distorção do objeto que se pretende imprimir.

Por outro lado, considera-se que o emissor pode ser ativado e desativado ao longo do processo, o que permite atingir diferentes *voxels* de forma independente. Assim, pode pensar-se na impressão de forma particionada, neste caso analisando a impressão de cada camada isoladamente. Com o objetivo de manter a estabilidade do processo, considera-se que a utilização de um emissor numa camada deve ser sequencial e, como tal, é desejável reduzir a ativação de diferentes emissores tanto quanto possível.

Suponhamos fixa uma camada a imprimir, $k = 1, \dots, p$. Uma vez que se pretende associar os emissores aos *voxels* a atingir, considerem-se as variáveis binárias y_{ij} , definidas por

$$y_{ij} = \begin{cases} 1 & \text{se o emissor } j \text{ é ativado para atingir o } \textit{voxel} i \\ 0 & \text{caso contrário} \end{cases}$$

para $i = 1, \dots, m_k$, $j = 1, \dots, n_2$.

A escolha do emissor a atingir um determinado *voxel* deve ter em conta dois aspetos: a viabilidade desta solução e a sua unicidade. Para assegurar o primeiro podem utilizar-se as condições de afetação

$$\sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k, \quad (4.1)$$

enquanto que o segundo depende das condições

$$\sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k, \quad (4.2)$$

onde $A = [a_{ij}]_{i=1, \dots, m_k; j=1, \dots, n_2}$ é uma submatriz da matriz de projeção definida no Capítulo 2, restrin-gida ao conjunto de *voxels* a solidificar e aos emissores utilizados. Como referido acima, as condições (4.1) e (4.2) não são redundantes, pois só os dois conjuntos garantem a obtenção de um único emissor a atingir cada *voxel* e que tal é possível.

O outro aspeto a considerar respeita aos emissores acionados para impressão da camada k . Assim sendo, considerem-se as variáveis x_j análogas às utilizadas no PLE, apresentado no Capítulo 3, $j = 1, \dots, n_2$. Recordar-se que a utilização direta do *laser* no topo da impressora corresponde a x_1 e as restantes variáveis correspondem aos sistemas de espelhos determinados pela solução do PLE.

Relativamente aos emissores a acionar em cada camada, podem utilizar-se as condições de cobertura introduzidas no capítulo anterior,

$$\sum_{j=1}^{n_2} a_{ij} x_j \geq 1, \quad i = 1, \dots, m_k. \quad (4.3)$$

Em alternativa, podem relacionar-se as variáveis y_{ij} e x_j , atendendo a que se o emissor j é ativado para atingir o *voxel* i , então ele é ativado. Em termos de condições isto corresponde a impor

$$y_{ij} \leq x_j, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2,$$

Note-se que $x_j = 0$ implica $y_{ij} = 0$, enquanto que se $x_j = 0$ se tem simplesmente $y_{ij} \leq 1$, o que não interfere com a definição de y_{ij} . De forma agregada estas condições podem ser escritas como

$$\sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2. \quad (4.4)$$

Acima mencionaram-se dois critérios a levar em conta ao associar os emissores e os *voxels* a solidificar: o número de sistemas de espelhos utilizados para a construção de uma camada específica, no que se segue designado por z_1 , e uma medida da distorção da forma da luz *laser* quando esta

incide na camada, designada por z_2 . O primeiro destes critérios, que se pretende minimizar, pode ser expresso por

$$z_1 = \sum_{j=2}^{n_2} x_j.$$

O segundo depende do emissor que atinge cada *voxel*, y_{ij} , e o ângulo de incidência correspondente, $\theta_{ij} \in [0, \frac{\pi}{2}]$, $i = 1, \dots, m_k$, $j = 1, \dots, n_2$. Procura-se que a distorção da luz *laser* seja o mais pequena possível, o que corresponde a fazer incidir essa luz o mais perpendicularmente possível na camada de impressão. Deste modo, deve maximizar-se

$$z_2 = \sum_{i=1}^{m_k} \sum_{j=1}^{n_2} \theta_{ij} y_{ij}.$$

Combinando toda a informação, obtém-se o seguinte problema linear

$$\begin{aligned} \min & & z_1(x, y) \\ \max & & z_2(x, y) \\ \text{sujeito a} & & \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \\ & & \sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \\ & & \sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2 \\ & & y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2 \\ & & x_j \in \{0, 1\}, \quad j = 1, \dots, n_2 \end{aligned} \quad (4.5)$$

com variáveis binárias e dois objetivos.

Espera-se que z_1 e z_2 sejam duas funções objetivo conflituosas. Isto significa que, em geral, não existe uma solução, ou seja, valores (x_j, y_{ij}) , que satisfaçam as restrições já introduzidas e que otimizem simultaneamente z_1 e z_2 . Em alternativa ao conceito habitual de optimalidade em problemas mono-objetivo, ao lidar com dois objetivos procura-se determinar soluções eficientes. Uma solução admissível de (4.5), (x', y') , diz-se uma solução eficiente se não existe outra solução admissível, (x'', y'') , que a domine, isto é, tal que

$$\begin{cases} z_1(x'', y'') \leq z_1(x', y') \\ z_2(x'', y'') \geq z_2(x', y') \end{cases} \text{ e } (z_1(x'', y''), z_2(x'', y'')) \neq (z_1(x', y'), z_2(x', y')).$$

Em [8] podem encontrar-se este e outros conceitos, resultados e métodos, relacionados com otimização envolvendo várias funções objetivo.

As abordagens a problemas multiobjetivo classificam-se em três categorias: métodos *a posteriori*, que calculam todas as soluções eficientes, com o decisor a definir depois as suas preferências; métodos interativos, para os quais há uma articulação progressiva das preferências do decisor, e métodos *a priori*, onde o decisor expressa as suas preferências antes do cálculo das soluções, por exemplo, através de uma função de utilidade.

Na literatura conhecem-se métodos para determinar todas as soluções eficientes de problemas lineares inteiros biobjetivo, como é o caso do problema (4.5), cujo comportamento depende do

comportamento da correspondente versão mono-objetivo. Atendendo à aplicação em impressão tridimensional, é suficiente (e até conveniente) determinar e selecionar uma única solução e, portanto, não é necessário calcular todas as soluções eficientes. Assim, descrevem-se agora dois métodos para este efeito. A maioria dos métodos utilizados em otimização biobjetivo baseia-se em alguma forma de reduzir das duas funções a otimizar a uma só. As abordagens descritas em seguida são adaptações de estratégias conhecidas para obter soluções eficientes de problemas de otimização biobjetivo. A primeira consiste em agregar os objetivos z_1 e z_2 na mesma função objetivo através de uma soma pesada. A segunda baseia-se numa perspectiva lexicográfica, em que se privilegia um dos objetivos, deixando o outro para segundo plano. Como se verá em seguida, qualquer destas duas abordagens permite obter soluções eficientes de (4.5).

4.2 Modelo de soma pesada

Uma das formas mais intuitivas de lidar com problemas biobjetivo consiste em agregar as duas funções objetivo por meio de uma soma. Adicionalmente, a importância relativa de cada função pode ser expressa multiplicando-a por um parâmetro bem escolhido. No caso em estudo isto significa considerar o problema

$$\begin{aligned}
 \min \quad & z(x, y) = \lambda_1 z_1(x, y) - \lambda_2 z_2(x, y) \\
 \text{sujeito a} \quad & \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \\
 & \sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \\
 & \sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2 \\
 & y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2 \\
 & x_j \in \{0, 1\}, \quad j = 1, \dots, n_2
 \end{aligned} \tag{4.6}$$

com os pesos $\lambda_1, \lambda_2 \in [0, 1]$ e tais que $\lambda_1 + \lambda_2 = 1$.

A utilização de somas pesadas para obter soluções eficientes é suportada pelo resultado seguinte.

Proposição 4.2.1 ([8]). *Toda a solução ótima de (4.6), com $\lambda_1, \lambda_2 \in]0, 1[$, é solução eficiente de (4.5).*

Demonstração. Consideremos (x^*, y^*) uma solução ótima de (4.6). Os problemas (4.5) e (4.6) têm as mesmas restrições, logo (x^*, y^*) é solução admissível do primeiro destes problemas. Por absurdo, suponhamos que (x^*, y^*) não é eficiente, ou seja, que existe outra solução (x', y') dos dois problemas, tal que

$$z_1(x', y') \leq z_1(x^*, y^*) \tag{4.7a}$$

$$z_2(x', y') \geq z_2(x^*, y^*) \tag{4.7b}$$

$$(z_1(x', y'), z_2(x', y')) \neq (z_1(x^*, y^*), z_2(x^*, y^*)) \tag{4.7c}$$

Como $0 < \lambda_1, \lambda_2 < 1$, as condições (4.7a) e (4.7b) implicam

$$\lambda_1 z_1(x', y') - \lambda_2 z_2(x', y') \leq \lambda_1 z_1(x^*, y^*) - \lambda_2 z_2(x^*, y^*).$$

Por outro lado, pela condição (4.7c), tem-se

$$z_1(x', y') < z_1(x^*, y^*) \text{ ou } z_2(x', y') > z_2(x^*, y^*).$$

Para qualquer destes dois casos vem

$$\lambda_1 z_1(x', y') - \lambda_2 z_2(x', y') < \lambda_1 z_1(x^*, y^*) - \lambda_2 z_2(x^*, y^*),$$

o que contraria a hipótese de (x^*, y^*) ser solução ótima de (4.6). \square

A Proposição 4.2.1 garante que resolvendo (4.6) para uma escolha de pesos $\lambda_1, \lambda_2 \in]0, 1[$, se encontra uma solução eficiente de (4.5). Note-se, contudo, que em geral, nem todas as soluções eficientes de um problema linear inteiro são combinação linear convexa de outras duas. Uma análise deste resultado está fora do âmbito deste trabalho, mas o leitor interessado pode consultar [8] para obter mais informação.

As duas funções objetivo, z_1 e z_2 , podem ter ordens de grandeza díspares, dado que representam medidas de natureza diferente. Para uniformizar a variação das duas funções objetivo quando agregadas, procede-se agora à normalização de z_1 e z_2 através da sua multiplicação por parâmetros $w_1, w_2 \in \mathbb{R}$.

A função z_1 é definida por um somatório com $n_2 - 1$ termos, cada um com o valor 0 ou 1, portanto $0 \leq z_1(x, y) \leq n_2 - 1$ e considera-se

$$w_1 = n_2 - 1.$$

Por outro lado, z_2 é dada por uma soma com $n_2 m_k$ termos. No entanto, qualquer que seja a solução admissível (x, y) , para cada i , existe um único j tal que $y_{ij} = 1$, e para os restantes casos tem-se $y_{ij} = 0$. Denotando por

$$\underline{\theta}_i = \min_{j=1, \dots, n_2} \{\theta_{ij}\} \text{ e } \bar{\theta}_i = \max_{j=1, \dots, n_2} \{\theta_{ij}\}, \quad i = 1, \dots, m_k,$$

tem-se $\sum_{i=1}^{m_k} \underline{\theta}_i \leq z_2 \leq \sum_{i=1}^{m_k} \bar{\theta}_i$ e, conseqüentemente, considera-se

$$w_2 = \sum_{i=1}^{m_k} (\bar{\theta}_i - \underline{\theta}_i).$$

Após a normalização, obtém-se a função objetivo a minimizar,

$$w_2 z_1 - w_1 z_2 = w_2 \sum_{j=2}^{n_2} x_j - w_1 \sum_{i=1}^{m_k} \sum_{j=1}^{n_2} \theta_{ij} y_{ij},$$

ou, de modo equivalente, pretende-se minimizar

$$\frac{w_2}{w_1} z_1 - z_2.$$

Por fim, retomando a utilização de pesos associados a $w_1 z_1$ e $w_2 z_2$, a função objetivo é

$$\lambda_1 \frac{w_2}{w_1} z_1 - \lambda_2 z_2.$$

Em conclusão, para a camada $k = 1, \dots, p$ é resolvido o problema linear binário, e mono-objetivo,

$$\begin{aligned}
\min \quad & \lambda_1 \frac{w_2}{w_1} z_1 - \lambda_2 z_2 \\
\text{sujeito a} \quad & \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& \sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& \sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2 \\
& y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2 \\
& x_j \in \{0, 1\}, \quad j = 1, \dots, n_2
\end{aligned} \tag{4.8}$$

Tal como sucede com (4.6), também as soluções de (4.8) com $\lambda_1, \lambda_2 \in]0, 1[$ são soluções eficientes de (4.5). Os argumentos para o demonstrar são semelhantes aos utilizados na Proposição 4.2.1.

Qualquer método que permita resolver um problema mono-objetivo com o conjunto de restrições de (4.5), permite resolver também (4.8). No entanto, estes são problemas lineares binários que combinam diferentes tipos de restrições, mas sem uma estrutura particular. Além disso, conforme referido no Capítulo 3, o problema de cobertura, caso particular deste, é um problema NP-completo e portanto de resolução difícil.

Se for conhecido o peso relativo de z_1 e z_2 podem escolher-se os pesos a utilizar, no entanto, estas situações são raras. Em alternativa, variar arbitrariamente esses pesos permite encontrar um conjunto de soluções eficientes.

Como já referido, os valores λ_1 e λ_2 funcionam como pesos para os dois critérios. Aumentar o primeiro significa dar maior importância ao número de emissores utilizados em cada camada, ao passo que aumentar o segundo implica dar maior importância à deformação do raio que atinge a camada. Em particular o caso em que $\lambda_1 = 0$ e $\lambda_2 = 1$ corresponde a minimizar apenas a distorção do *laser* e $\lambda_1 = 1$ e $\lambda_2 = 0$ corresponde a minimizar o número de emissores. No primeiro destes casos particulares, $\lambda_1 = 0$, $\lambda_2 = 1$, z_1 não afeta a função objetivo e portanto as variáveis x_j podem ser eliminadas. Então, o problema pode ser escrito simplesmente como

$$\begin{aligned}
\max \quad & z_2(x, y) = \sum_{i=1}^{m_k} \sum_{j=1}^{n_2} \theta_{ij} y_{ij} \\
\text{sujeito a} \quad & \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& \sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2
\end{aligned}$$

e pode ser resolvido facilmente de forma exata. Com efeito, os dois conjuntos iniciais de restrições obrigam a que, para cada *voxel* $i = 1, \dots, m_k$, apenas uma das variáveis y_{ij} tenha o valor 1, enquanto que as restantes são nulas. Uma vez que os valores θ_{ij} são conhecidos, $i = 1, \dots, m_k$, $j = 1, \dots, n_2$, z_2 pode ser maximizada, escolhendo, para cada i , o maior ângulo θ_{ij} de algum emissor que o atinja. Ou seja, é necessário determinar o emissor j que atinge i com valor máximo de θ_{ij} ,

$$j^* = \operatorname{argmax}\{\theta_{ij} : a_{ij} = 1\}$$

e fazer

$$y_{ij} = \begin{cases} 1 & \text{se } j = j^* \\ 0 & \text{caso contrário} \end{cases}$$

Portanto, para cada camada $k = 1, \dots, p$, pode ser aplicado o método resumido no Algoritmo 5. É de realçar que este algoritmo é polinomial em termos do número de operações executadas e de memória, ambos da $O(m_k n_2)$.

Algoritmo 5: Minimização da distorção da camada k

```

for  $i = 1, \dots, m_k$  do
   $j^* \leftarrow \operatorname{argmax}\{\theta_{ij} : a_{ij} = 1\}$ 
  for  $j = 1, \dots, n_2$  do  $y_{ij} \leftarrow 0$ 
   $y_{ij^*} \leftarrow 1$  /* Emissor  $j^*$  atinge vowel  $i$  */

```

Para minimizar apenas o número de emissores, podem tomar-se os pesos $\lambda_1 = 1$ e $\lambda_2 = 0$ no problema (4.6). Este problema pode ter múltiplas soluções ótimas, o que, na prática, corresponde a várias possibilidades de afetação de emissores a alguns *voxels*. Nesse caso o problema (4.6) não especifica um critério para selecionar o emissor. O modelo proposto em seguida aborda esta questão.

4.3 Modelo lexicográfico

Por vezes pode assumir-se à partida a maior importância de um objetivo relativamente a outro, sem expressar uma dependência clara entre ambos, como acontece quando se utilizam pesos. Esta abordagem é tratada otimizando dois problemas mono-objetivo consecutivos, um considerando apenas a função objetivo de maior importância e outro, restringido às soluções do primeiro, em que se otimiza a função objetivo restante. É portanto estabelecida uma ordem entre as duas funções e estas são tratadas lexicograficamente [8].

Considere-se que z_1 e z_2 são a primeira e a segunda funções objetivo, respetivamente. O método lexicográfico aplicado ao PLE consiste em resolver dois problemas com um objetivo. Primeiro resolve-se o problema (4.5), minimizando apenas a função z_1 ,

$$\begin{aligned} \min \quad & z_1(x, y) \\ \text{sujeito a} \quad & \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \end{aligned} \quad (4.9a)$$

$$\sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \quad (4.9b)$$

$$\sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2 \quad (4.9c)$$

$$y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2$$

$$x_j \in \{0, 1\}, \quad j = 1, \dots, n_2$$

Seja z_1^* o valor ótimo de (4.9). Em seguida, maximiza-se apenas a função z_2 no conjunto de todas as soluções ótimas de (4.9), isto é,

$$\begin{aligned}
& \min && z_2(x, y) \\
& \text{sujeito a} && \sum_{j=1}^{n_2} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& && \sum_{j=1}^{n_2} a_{ij} y_{ij} = 1, \quad i = 1, \dots, m_k \\
& && \sum_{i=1}^{m_k} y_{ij} \leq m_k x_j, \quad j = 2, \dots, n_2 \\
& && \sum_{j=1}^{n_2} x_j = z_1^* \\
& && y_{ij} \in \{0, 1\}, \quad i = 1, \dots, m_k, \quad j = 1, \dots, n_2 \\
& && x_j \in \{0, 1\}, \quad j = 1, \dots, n_2
\end{aligned} \tag{4.10}$$

Proposição 4.3.1 ([8]). *Toda a solução ótima de (4.10) é solução eficiente de (4.5).*

Demonstração. Consideremos (x^*, y^*) uma solução ótima de (4.10), isto é, uma solução lexicograficamente ótima relativamente a (z_1, z_2) . A região admissível de (4.11) está contida na de (4.5), portanto (x^*, y^*) é solução dos dois problemas.

Suponhamos que (x^*, y^*) não é eficiente. Então existe outra solução de (4.5), (x', y') , que satisfaz as condições (4.7a) – (4.7c). Como (x^*, y^*) é ótima relativamente a z_1 , então (4.7a) implica

$$z_1(x', y') = z_1(x^*, y^*),$$

donde (x', y') também é solução ótima de (4.9) e solução admissível de (4.10). Atendendo ainda a (4.7b) e (4.7c), conclui-se que

$$z_2(x', y') > z_2(x^*, y^*),$$

o que contraria a hipótese de (x^*, y^*) ser uma solução lexicograficamente ótima, ou seja, ser solução ótima de (4.10). \square

A Proposição 4.2.1 garante que ao resolver (4.9), seguido de (4.10), se encontra uma solução eficiente de (4.5).

Em termos práticos, resolver (4.9) permite obter os emissores a utilizar para construção da camada k , enquanto que resolver (4.10) permite obter aquela que menos distorce a forma do *laser* ao atingir a camada de impressão. Em termos matemáticos, e algorítmicos, os problemas referidos podem ser simplificados.

É claro que a função objetivo de (4.9), z_2 , depende apenas das variáveis x_j , $j = 1, \dots, n_2$. Com efeito, as variáveis y_{ij} podem ser eliminadas daquela formulação, $i = 1, \dots, m_k$, $j = 1, \dots, n_2$. Seja (x, y) uma solução admissível desse problema. Então, por (4.9c), para cada $i = 1, \dots, m_k$,

$$\sum_{j=1}^{n_2} a_{ij} x_j \geq \sum_{j=1}^{n_2} a_{ij} \sum_{i=1}^{m_k} \frac{y_{ij}}{m_k} = \frac{1}{m_k} \sum_{i=1}^{m_k} \sum_{j=1}^{n_2} a_{ij} y_{ij}.$$

Utilizando agora (4.9b),

$$\frac{1}{m_k} \sum_{i=1}^{m_k} 1 = 1,$$

donde

$$\sum_{j=1}^{n_2} a_{ij} x_j \geq 1.$$

Portanto, para $k = 1, \dots, p$, o valor ótimo de (4.9) pode ser determinado resolvendo o problema

$$\begin{aligned} \min \quad & \sum_{j=1}^{n_2} x_j \\ \text{sujeito a} \quad & \sum_{j=1}^{n_2} a_{ij} x_j \geq 1, \quad i = 1, \dots, m_k \\ & x_j \in \{0, 1\}, \quad j = 1, \dots, n_2 \end{aligned} \quad (4.11)$$

É de salientar que (4.11) é um problema de cobertura com cardinalidade mínima análogo ao definido no Capítulo 3 para o PLE, mas restringido a uma camada específica k . A resolução deste problema permite saber quais os emissores a utilizar para construção da camada k .

Dada uma solução de (4.11), ou seja, um conjunto de emissores a utilizar, para cada *voxel*, apenas é necessário conhecer quais os que o atingem, selecionando em seguida aquele que menos distorcer a forma do *laser* ao atingir o polímero. Conforme mencionado, isto corresponde a escolher o emissor j que atinge o *voxel* i com o maior ângulo θ_{ij} . Para o efeito é feita uma pesquisa semelhante à utilizada no Algoritmo 5, mas com uma condição adicional. É necessário garantir que o emissor está a ser utilizado na impressão da camada.

O método descrito é resumido no Algoritmo 6. Tal como o algoritmo anterior, o Algoritmo 6 é da $O(m_k n_2)$ em número de operações executadas e em memória.

Algoritmo 6: Minimização da distorção da camada k utilizando os emissores na solução x

```

for  $i = 1, \dots, m_k$  do
   $j^* \leftarrow \operatorname{argmax}\{\theta_{ij} : a_{ij} = 1 \wedge x_j = 1\}$ 
  for  $j = 1, \dots, n_2$  do  $y_{ij} \leftarrow 0$ 
   $y_{ij^*} \leftarrow 1$            /* Emissor  $j^*$  atinge voxel  $i$            */

```

Capítulo 5

Experiência computacional

Neste capítulo, são testados os modelos e métodos propostos nos Capítulos 3 e 4. Para esse efeito considera-se a impressão de um cubo sem uma das faces. Mais especificamente, pretende-se revestir cada uma das cinco faces de rede metálica uniforme com polímero. Primeiro são descritos os detalhes deste caso de estudo e são definidos alguns parâmetros variáveis do mesmo, bem como apresentados os parâmetros que foram testados. São apresentados os resultados no que toca à resolução do PLE e do PAE. Por fim, é analisada a distorção da luz *laser* para as soluções obtidas. ¹

5.1 Caso de estudo

Neste caso de estudo pretende-se imprimir o cubo representado na Figura 5.1a, com apenas cinco faces (faltando a da base). A grelha visível na imagem representa a malha metálica existente e que deve ser revestida de polímero. A malha metálica na face do topo do cubo está representada a azul, da frente e da face traseira está representada a vermelho, e a das faces da esquerda e da direita está representada a verde.

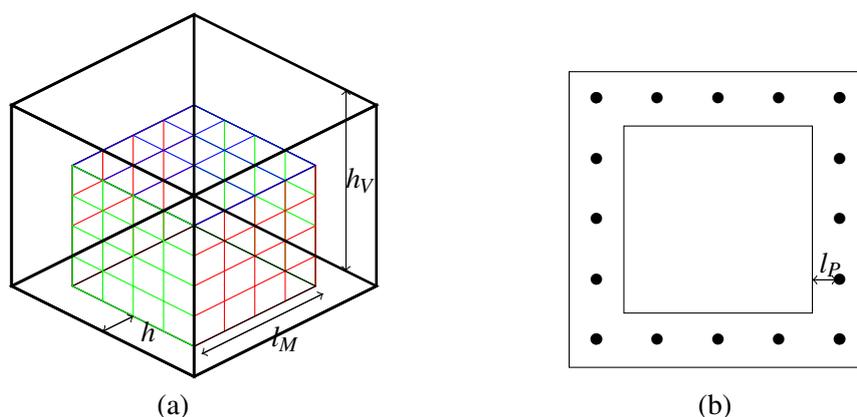


Figura 5.1 (a) Zona de impressão e objeto a imprimir; (b) Plano horizontal intermédio do objeto a imprimir

¹Os códigos usados neste capítulo encontram-se disponíveis em https://www.mat.uc.pt/~mat1229/Dissertacao-experiencia_computacional/.

Considera-se que a espessura da rede metálica é igual à espessura das camadas de polímero e este será o valor utilizado para o lado dos *voxels* considerados neste caso de estudo. A Figura 5.1b representa uma camada intermédia do cubo a ser impresso.

Os parâmetros variáveis para a impressão deste cubo são:

- O comprimento de cada segmento da rede metálica, denotado por l_M .
- A espessura do polímero adicionado em cada lado das faces de grelha metálica, denotada por l_P .
- O número de divisões da malha de metal, que se supõe dividida de forma uniforme, denotado por n_M .
- O espaço entre o cubo a imprimir e as paredes laterais da impressora, onde podem ser instalados os sistemas de espelhos galvanométricos, denotado por h .
- A altura da zona de impressão, denotada por h_V .

Note-se que, para a face superior, o comprimento da aresta do cubo final a imprimir, denotado por n_V , corresponde à adição que é feita de polímero, com comprimento l_P , de cada lado ao comprimento das arestas do cubo de metal dado por l_M , ou seja, tem-se $n_V = 2l_P + l_M$. Para as faces laterais, o comprimento da aresta do cubo final será dado por $l_P + l_M$, pois o cubo não tem face inferior, ou seja, nessa zona não é adicionado polímero. Deste modo, o objeto é formado por $l_P + l_M$ camadas. Para os testes apresentados são excluídas as duas camadas superiores. Para a sua impressão a face superior da grelha já se encontra ao nível da camada que está a ser imprimida ou abaixo desta e toda a restante estrutura de metal abaixo do plano de impressão. Portanto, pode ser utilizado diretamente o *laser* instalado na parte superior da impressora. Considera-se que o cubo é impresso centrado na plataforma de impressão.

No que se segue a unidade dos valores de comprimentos apresentados, corresponde a 0.2 milímetros, sendo este o valor do lado dos *voxels* considerados. Além disso, consideraram-se as dimensões da impressora fixas, sendo $h_V = 1250$. O valor de h varia, uma vez que depende da dimensão do objeto, no entanto é sempre escolhido o valor de forma a que a base impressora também tenha 1250 unidades de comprimento e largura. Portanto, a zona de impressão tem sempre dimensão $1250 \times 1250 \times 1250$.

Relativamente ao cubo a imprimir, recorde-se que se considera a malha metálica com espessura igual a 1 e:

- $n_M = 5, 10, 20$,
- $n_V = 200, 300, 500$,
- $l_P = 1$.

O valor de n_V determina o número de *voxels* de cada camada, sendo este dado por $n_V \times n_V$. As características dos problemas resolvidos resumem-se na Tabela 5.1.

Tabela 5.1 Parâmetros dos testes

| Teste | n_V | n_M | h | Teste | n_V | n_M | h | Teste | n_V | n_M | h |
|-------|-------|-------|-----|-------|-------|-------|-----|-------|-------|-------|-----|
| T1 | 200 | 5 | 525 | T4 | 300 | 5 | 475 | T7 | 500 | 5 | 375 |
| T2 | 200 | 10 | 525 | T5 | 300 | 10 | 475 | T8 | 500 | 10 | 375 |
| T3 | 200 | 20 | 525 | T6 | 300 | 20 | 475 | T9 | 500 | 20 | 375 |

5.2 Problema da localização de emissores

Nesta secção aborda-se a resolução do PLE por aplicação de um pacote para programação matemática à formulação (3.1) e utilizando as heurísticas apresentadas no Capítulo 3.

Para os testes apresentados nesta secção, recorreu-se ao MATLAB para calcular a matriz de cobertura dos emissores utilizada na definição do PLE e para a implementação dos métodos heurísticos. O problema linear binário (3.1) foi resolvido utilizando o CPLEX, visto ser um *software* específico para este tipo de problemas. Os tempos de execução para os problemas lineares e para as heurísticas foram obtidos através de um computador com processador Intel i7-6700 Quadcore de 3.4 GHz, 8Mb de cache, 16 Gb de RAM em openSUSE Leap 42.2. Todos os tempos de execução apresentados são valores médios obtidos para 30 repetições.

O PLE foi resolvido para os testes apresentados na Tabela 5.1. Além da utilização direta do *laser*, que corresponde a uma posição de emissão, consideraram-se 80 possíveis posições para localização de emissores, distribuídas pelas paredes laterais da impressora, resultando num total de 81 possíveis posições de emissão. Os resultados obtidos através da resolução direta do problema linear binário com recurso ao CPLEX apresentam-se na Tabela 5.2.

Tabela 5.2 PLE: soluções e tempos de execução (segundos) com CPLEX

| Teste | Posições dos sistemas de espelhos | Tempo |
|-------|------------------------------------------------------------------|---------|
| T1 | (1, 1000, 250) e (1250,1,250) | 5.03 |
| T2 | (1, 1000, 1000), (1, 251,250) e (1250, 1250, 250) | 19.13 |
| T3 | (1250, 1, 1000), (1, 1, 250), (1, 1250, 250) e (1250, 1250, 250) | 140.03 |
| T4 | (1, 1, 750) e (1250, 1250, 250) | 9.89 |
| T5 | (1, 750, 1000), (1, 251, 500) e (1250, 1000, 500) | 34.98 |
| T6 | (1, 1, 500), (1250, 1250, 500), (1, 750, 250) e (1250, 501, 250) | 1323.36 |
| T7 | (1, 1, 750) e (1250, 1250, 500) | 27.52 |
| T8 | (1, 1, 500), (1250, 1250, 500) e (1250, 1, 250) | 40.70 |
| T9 | (1, 1, 500), (1250, 1250, 500) e (1250, 1, 250) | 144.05 |

De acordo com a Tabela 5.2, foram obtidas soluções em que são necessários entre dois a quatro sistemas de espelhos para a impressão completa do objeto. Tal como seria de esperar, em geral, quando os objetos são formados por uma grelha de metal com uma malha com um maior número de divisões, são necessários mais emissores na impressão. Nota-se também que o aumento do número de divisões da malha de metal implica um aumento no tempo da resolução dos problemas lineares binários.

Aos mesmos testes foram aplicadas as heurísticas apresentadas nos Algoritmos 2 e 3. Numa primeira fase testaram-se as heurísticas sem aplicação de regras para a redução nem eliminação de

colunas redundantes. Apenas se aplicou a redução de selecionar o primeiro emissor e eliminar os *voxels* por ele atingidos, que se designará por regra R1'. Os resultados para estes testes são resumidos apresentados na Tabela 5.3. Para simplificar, agora apresentam-se apenas o número de sistemas de espelhos da solução, denotado z . São também apresentados os tempos de execução.

Tabela 5.3 PLE: soluções e tempos de execução (segundos) dos Algoritmos 2 e 3

| Teste | Alg. 2 | | Alg. 3 | |
|-------|--------|-------|--------|-------|
| | z | Tempo | z | Tempo |
| T1 | 3 | 0.10 | 3 | 0.16 |
| T2 | 3 | 0.17 | 3 | 0.37 |
| T3 | 4 | 0.26 | 4 | 0.60 |
| T4 | 3 | 0.15 | 2 | 0.22 |
| T5 | 3 | 0.27 | 3 | 0.55 |

| Teste | Alg. 2 | | Alg. 3 | |
|-------|--------|-------|--------|-------|
| | z | Tempo | z | Tempo |
| T6 | 4 | 0.40 | 4 | 0.97 |
| T7 | 3 | 0.30 | 2 | 0.39 |
| T8 | 3 | 0.48 | 3 | 0.81 |
| T9 | 3 | 0.68 | 4 | 1.40 |

A Figura 5.2 ilustra os resultados apresentados na Tabela 5.3.

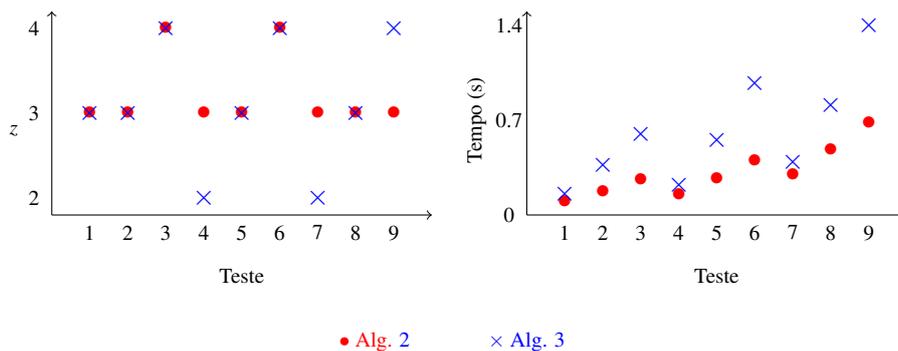


Figura 5.2 PLE: soluções e tempos de execução dos Algoritmos 2 e 3

Observando a Tabela 5.3 e a Figura 5.2 é possível fazer uma comparação entre as duas heurísticas. Para o teste T9 o Algoritmo 2 apresentou uma solução melhor do que o Algoritmo 3, enquanto que para os testes T4 e T7 se verificou o oposto. Nos restantes testes as duas heurísticas encontraram soluções com o mesmo valor. Relativamente aos tempos de execução, o Algoritmo 2 foi sempre mais eficiente do que o Algoritmo 3. Desta forma, em geral, para os testes apresentados o Algoritmo 2 apresentou-se como a melhor escolha.

Após uma aplicação simples dos Algoritmos 2 e 3 foram feitos novos testes, incluindo, além da redução obtida pela escolha do primeiro emissor, as regras de redução R3 e R4 apresentadas no Capítulo 3. As regras R1 e R2 não foram aplicadas, visto que para o problema considerado não é de esperar que tenham efeito. Foi também acrescentada a eliminação de colunas redundantes do Algoritmo 4. Para simplificar a exposição, designa-se por H1 a heurística constituída pelo Algoritmo 2 e as restantes regras, e por H2 a heurística constituída pelo Algoritmo 3 e as restantes regras. Os resultados para estes testes são apresentados na Tabela 5.4, para a heurística H1, e na Tabela 5.5, para a heurística H2. Apresentam-se os tempos parciais e totais. Os resultados com tempos totais superiores a 3600 segundos são omitidos das tabelas.

Tabela 5.4 PLE: soluções e tempos de execução (segundos) da heurística H1

| Teste | z | Tempo | | | | | Total |
|-------|---|-------|--------|------|--------|--------|--------|
| | | R1' | R3 | R4 | Alg. 2 | Alg. 4 | |
| T1 | 2 | 0.03 | 83.66 | 0.11 | 0.00 | 0.03 | 83.83 |
| T2 | 3 | 0.03 | 542.39 | 0.34 | 0.01 | 0.04 | 542.81 |
| T3 | 4 | – | – | – | – | – | – |
| T4 | 2 | 0.06 | 88.33 | 0.09 | 0.00 | 0.08 | 88.57 |
| T5 | 3 | 0.06 | 544.46 | 0.41 | 0.02 | 0.13 | 545.07 |
| T6 | 4 | – | – | – | – | – | – |
| T7 | 2 | 0.17 | 66.44 | 0.04 | 0.00 | 0.28 | 66.93 |
| T8 | 3 | 0.17 | 423.83 | 0.29 | 0.01 | 0.42 | 424.72 |
| T9 | 3 | – | – | – | – | – | – |

Tabela 5.5 PLE: soluções e tempos de execução (segundos) da heurística H2

| Teste | z | Tempo | | | | | Total |
|-------|---|-------|--------|------|--------|--------|--------|
| | | R1' | R3 | R4 | Alg. 3 | Alg. 4 | |
| T1 | 2 | 0.03 | 83.21 | 0.11 | 0.01 | 0.03 | 83.39 |
| T2 | 3 | 0.03 | 529.57 | 0.33 | 0.02 | 0.05 | 530.00 |
| T3 | 4 | – | – | – | – | – | – |
| T4 | 2 | 0.06 | 85.86 | 0.09 | 0.00 | 0.09 | 86.10 |
| T5 | 3 | 0.06 | 552.67 | 0.41 | 0.03 | 0.13 | 553.30 |
| T6 | 4 | – | – | – | – | – | – |
| T7 | 2 | 0.17 | 66.37 | 0.04 | 0.00 | 0.28 | 66.86 |
| T8 | 3 | 0.16 | 423.40 | 0.29 | 0.02 | 0.43 | 424.30 |
| T9 | 4 | – | – | – | – | – | – |

Com base nas Tabelas 5.3, 5.4 e 5.5, verifica-se que são obtidas igualmente soluções com dois a quatro sistemas de espelhos. Tal como seria de esperar, estas não produzem sempre resultados ótimos. Para os testes efetuados, apenas a heurística H1 com a aplicação das regras R3, R4 e o Algoritmo 4 obteve sempre o valor ótimo do problema. Também para a heurística H2 a aplicação de regras de redução e eliminação de redundância produziu melhores resultados do que o registado na Tabela 5.3. Relativamente aos tempos obtidos, sem a aplicação de regras foram obtidos tempos bastante melhores do que os correspondes através da resolução do problema utilizando o CPLEX. Pelo contrário, quando se utilizaram regras para reduzir o problema os tempos de execução foram bastante superiores aos obtidos pelo CPLEX. Pelas Tabelas 5.4 e 5.5 percebe-se que isso é essencialmente devido à regra R3, utilizada para reduzir o número de linhas. No Capítulo 3 observou-se que esta regra de redução tem uma complexidade de $O(m^2n)$, portanto o elevado número de linhas dos problemas de teste considerados, m , justifica estes resultados.

5.3 Problema da afetação de emissores

Nesta secção resolve-se o PAE através dos dois modelos apresentados no Capítulo 4. Tal como para a resolução do PLE, foram utilizados os pacotes de *software* MATLAB e CPLEX. O primeiro foi utilizado para efetuar os cálculos da matriz de cobertura dos emissores e para implementar os Algoritmos 5 e 6 apresentados no Capítulo 4. O segundo foi aplicado para resolver os problemas lineares binários associados a este problema, (4.8) e (4.11). A máquina utilizada para medir os tempos de execução foi a utilizada na resolução do PLE. Novamente, todos os tempos de execução apresentados correspondem a uma média de 30 medições.

Como referido anteriormente, a resolução do PAE deve ser precedida pela resolução do PLE. Por este motivo, consideram-se novamente os testes resumidos na Tabela 5.1 e as soluções ótimas do PLE apresentadas na Tabela 5.2. Para cada um, aplicaram-se:

- o modelo de soma pesada, usando como pesos os pares de valores $(1 - \lambda, \lambda)$, com $\lambda = 0, 0.25, 0.5, 0.75, 1$, representados no resto do texto por A1, A2, A3, A4 e A5, respetivamente;
- o modelo lexicográfico, representado por A6 e que corresponde à resolução do problema (4.11) seguida da aplicação do Algoritmo 6.

Para cada teste realizado foram calculados os ângulos α e θ , introduzidos no Capítulo 2, a utilizar pelo emissor que atinge cada *voxel*, bem como obtidos o número de espelhos que é necessário utilizar na impressão de cada camada. A Tabela 5.6 indica os valores mínimo, médio e máximo do número de sistemas de espelhos necessários para a impressão de cada camada, denotados \underline{z}_1 , μ_1 e \overline{z}_1 , respetivamente, e de θ , denotados \underline{z}_2 , μ_2 e \overline{z}_2 .

Para as várias abordagens, também se pretendeu analisar o tempo requerido resolver o PAE. No caso do modelo de soma pesada esta é feita diretamente através do problema (4.8), enquanto que usando o modelo lexicográfico, é necessário à solução do problema (4.11) aplicar o Algoritmo 6 a todas as camadas. Por esse motivo começa por se apresentar os tempos de execução parciais para a implementação do modelo lexicográfico, na Tabela 5.7. Na Tabela 5.8 resumem-se os tempos para resolução do PAE para todas as abordagens consideradas.

O Algoritmo 5 é uma alternativa à abordagem A5. Para testar a sua eficiência foram também obtidos os seus tempos de execução quando aplicado a todas as camadas. Estes valores são registados na Tabela 5.9.

A Figura 5.3 resume os principais resultados apresentados nas Tabelas 5.6 e 5.8.

A partir da Tabela 5.6 e da Figura 5.3 pode notar-se que, em termos da solução obtida, a abordagem A2 foi melhor do que as abordagens A1 e A6 em todos os casos, visto que μ_1 é sempre o mesmo e μ_2 tem valor máximo com a abordagem A2. Os resultados obtidos pelas abordagens A2, A3, A4 e A5 diferem de acordo com os pesos dados a z_1 e a z_2 no modelo da soma pesada. Assim, a escolha da abordagem a utilizar deve depender da importância que os dois fatores considerados têm na prática. Outro dado a reter perante os resultados obtidos, é que são sempre necessários pelo menos dois sistemas de espelhos para construir uma camada. Esta informação é especialmente relevante quando se utilizam dois sistemas de espelhos para a construção completa do objeto, uma vez que nesta situação o único critério efetivo na otimização é z_2 . Tal situação é confirmada pelos testes T1, T4 e T7, em que os resultados são iguais para todas as abordagens, exceto para a abordagem A1, em que

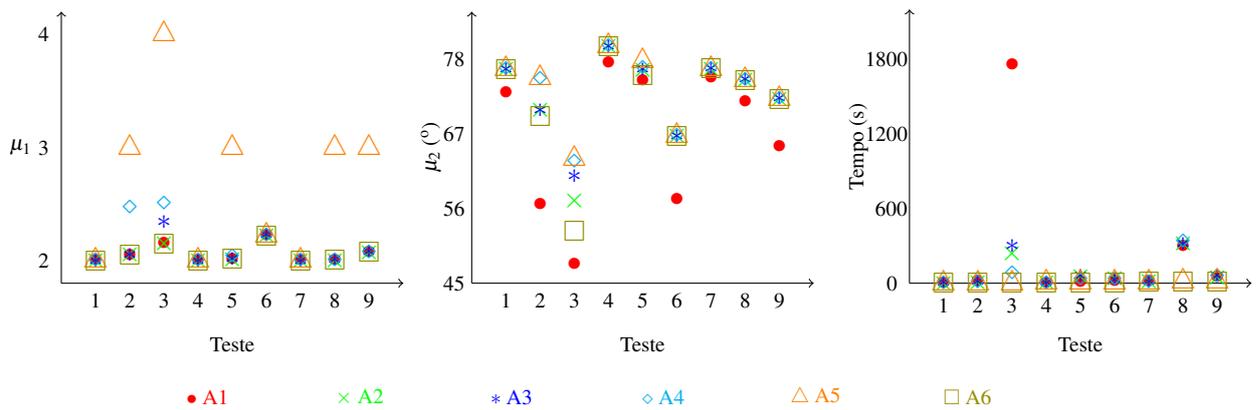


Figura 5.3 PAE: soluções e tempos de execução

não há qualquer minimização de z_2 e por isso a afetação é feita sem critério, uma vez que todas as soluções admissíveis têm o mesmo valor ótimo nesta abordagem.

Para a análise dos tempos obtidos para a resolução do problema, observe-se a Tabela 5.8, onde se destacam a sublinhado o maior, e a negrito o menor, tempo obtido, para cada teste. A abordagem A1 foi a que apresentou mais vezes os piores tempos, enquanto que a abordagem A5 foi mais vezes a mais rápida.

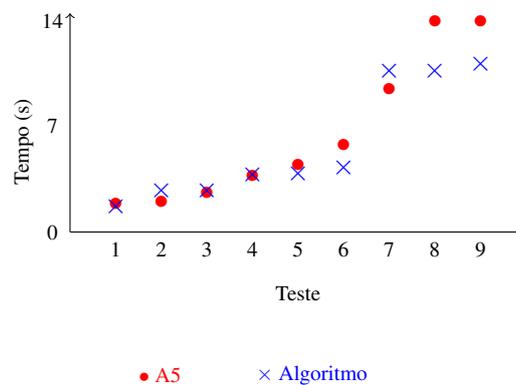


Figura 5.4 PAE: tempos de execução da abordagem A5 e do Algoritmo 5

Na Figura 5.4 são comparados os tempos de execução da abordagem A5 e do Algoritmo 5. Estes tempos também podem ser consultados nas Tabelas 5.8 e 5.9. A Figura 5.4 mostra um comportamento semelhante da abordagem A5 e o Algoritmo 5.

5.4 Análise das soluções obtidas

A presente secção é dedicada à análise dos efeitos da distorção da luz *laser* nas soluções obtidas. Conforme mostrado no Capítulo 2, conhecendo os valores dos ângulos α e θ com que a luz *laser* incide em cada *voxel*, é possível calcular a área do *voxel* que fica por solidificar, A_{int} , bem como a área fora do *voxel* atingida indevidamente, A_{ext} .

O *laser* usualmente utilizado para este processo de impressão tem um raio de 0.05 milímetros, pelo que, tendo em conta a unidade de medida considerada, a unidade como o lado do *voxel*, o *laser* tem um raio de 0.25 unidades.

Para cada teste realizado, calcularam-se os valores de A_{int} e A_{ext} para todos os *voxels*. A Tabela 5.10 mostra os valores mínimos, médio e máximo da percentagem relativamente à área do *voxel* de A_{int} , $\overline{A_{int}}$, μ_{int} e $\overline{A_{int}}$, e de A_{ext} , $\overline{A_{ext}}$, μ_{ext} e $\overline{A_{ext}}$, respetivamente. A Figura 5.5 apresenta a comparação entre os valores de μ_{int} e μ_{ext} .

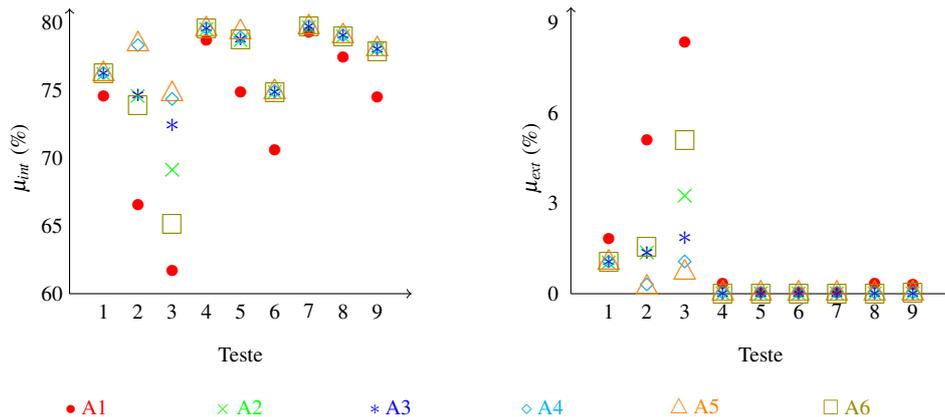


Figura 5.5 Áreas de distorção

Com base na Tabela 5.10 e na Figura 5.5, observa-se que em todos os casos há área dos *voxels* por solidificar e que na maioria dos casos há área também atingida fora dos *voxels*. Relativamente a A_{int} , o valor médio foi superior a 60% em todos os testes. Os principais motivos são a simplificação ao considerar que o *laser* atinge apenas o centro dos *voxels* e considerar-se *voxels* cujos lados são o dobro do diâmetro do raio do *laser*. Trabalhar com *voxels* mais pequenos, resultaria numa redução desta área, no entanto aumentaria o valor de A_{ext} .

A área A_{ext} é nula no caso T5 e praticamente desprezável no caso T7. Nestes casos os ângulos θ mínimos com que o *laser* atinge os *voxels* são superiores aos restantes, o que também está relacionado com a resolução do PLE em que foram obtidas posições para os emissores com alturas mais baixas, com valores de 500 unidades. Para os outros casos obtiveram-se posições de emissores com alturas de 250 unidades. É ainda de referir que para os restantes testes, apesar de o valor máximo da percentagem de área A_{ext} ser considerável (o valor mais baixo encontrado foi 17.28%), o valor médio em alguns deles é desprezável, chegando mesmo a 0.00%. Supõe-se que tal aconteça por haver poucos *voxels* em que é necessário recorrer aos emissores colocados em posições mais baixas. Por fim, é de referir que o caso com a maior percentagem média da área A_{ext} corresponde ao teste T3, abordagem A1, com um valor de 8.31%.

Tabela 5.6 PAE: soluções

| Teste | Abordagem | \underline{z}_1 | μ_1 | \bar{z}_1 | \underline{z}_2 (°) | μ_2 (°) | \bar{z}_2 (°) |
|-------|-----------|-------------------|---------|-------------|-----------------------|-------------|-----------------|
| T1 | A1 | 2 | 2 | 2 | 13.730 | 73.062 | 89.968 |
| | A2 | 2 | 2 | 2 | 14.166 | 76.570 | 89.968 |
| | A3 | 2 | 2 | 2 | 14.166 | 76.570 | 89.968 |
| | A4 | 2 | 2 | 2 | 14.166 | 76.570 | 89.968 |
| | A5 | 2 | 2 | 2 | 14.166 | 76.570 | 89.968 |
| | A6 | 2 | 2 | 2 | 14.166 | 76.570 | 89.968 |
| T2 | A1 | 2 | 2.051 | 3 | 13.730 | 56.589 | 89.968 |
| | A2 | 2 | 2.051 | 3 | 13.740 | 70.514 | 89.968 |
| | A3 | 2 | 2.051 | 3 | 13.740 | 70.514 | 89.968 |
| | A4 | 2 | 2.472 | 3 | 13.740 | 75.122 | 89.968 |
| | A5 | 3 | 3 | 3 | 13.952 | 75.311 | 89.968 |
| | A6 | 2 | 2.051 | 3 | 13.740 | 69.596 | 89.968 |
| T3 | A1 | 2 | 2.152 | 4 | 13.730 | 47.783 | 89.968 |
| | A2 | 2 | 2.152 | 4 | 13.740 | 57.216 | 89.968 |
| | A3 | 2 | 2.340 | 4 | 13.740 | 60.801 | 89.968 |
| | A4 | 2 | 2.508 | 4 | 13.740 | 62.865 | 89.968 |
| | A5 | 4 | 4 | 4 | 14.213 | 63.328 | 89.968 |
| | A6 | 2 | 2.152 | 4 | 13.740 | 52.774 | 89.968 |
| T4 | A1 | 2 | 2 | 2 | 12.873 | 77.529 | 89.968 |
| | A2 | 2 | 2 | 2 | 13.365 | 79.914 | 89.968 |
| | A3 | 2 | 2 | 2 | 13.365 | 79.914 | 89.968 |
| | A4 | 2 | 2 | 2 | 13.365 | 79.914 | 89.968 |
| | A5 | 2 | 2 | 2 | 13.365 | 79.914 | 89.968 |
| | A6 | 2 | 2 | 2 | 13.365 | 79.914 | 89.968 |
| T5 | A1 | 2 | 2.014 | 3 | 28.155 | 74.814 | 89.968 |
| | A2 | 2 | 2.014 | 3 | 28.183 | 76.430 | 89.968 |
| | A3 | 2 | 2.014 | 3 | 28.183 | 76.430 | 89.968 |
| | A4 | 2 | 2.040 | 3 | 28.183 | 76.683 | 89.968 |
| | A5 | 3 | 3 | 3 | 28.198 | 77.732 | 89.968 |
| | A6 | 2 | 2.014 | 3 | 28.198 | 75.605 | 89.968 |
| T6 | A1 | 2 | 2.222 | 4 | 16.932 | 57.315 | 89.968 |
| | A2 | 2 | 2.222 | 4 | 17.052 | 66.719 | 89.968 |
| | A3 | 2 | 2.226 | 4 | 17.052 | 66.719 | 89.968 |
| | A4 | 2 | 2.226 | 4 | 17.052 | 66.719 | 89.968 |
| | A5 | 4 | 4 | 4 | 17.756 | 66.721 | 89.968 |
| | A6 | 2 | 2.222 | 4 | 17.052 | 66.711 | 89.968 |
| T7 | A1 | 2 | 2 | 2 | 22.650 | 75.315 | 89.968 |
| | A2 | 2 | 2 | 2 | 22.882 | 76.649 | 89.968 |
| | A3 | 2 | 2 | 2 | 22.882 | 76.649 | 89.968 |
| | A4 | 2 | 2 | 2 | 22.882 | 76.649 | 89.968 |
| | A5 | 2 | 2 | 2 | 22.882 | 76.649 | 89.968 |
| | A6 | 2 | 2 | 2 | 22.882 | 76.649 | 89.968 |
| T8 | A1 | 2 | 2.006 | 3 | 11.441 | 71.803 | 89.968 |
| | A2 | 2 | 2.006 | 3 | 11.615 | 75.008 | 89.968 |
| | A3 | 2 | 2.006 | 3 | 11.615 | 75.008 | 89.968 |
| | A4 | 2 | 2.006 | 3 | 11.615 | 75.008 | 89.968 |
| | A5 | 3 | 3 | 3 | 11.615 | 75.008 | 89.968 |
| | A6 | 2 | 2.006 | 3 | 11.441 | 74.940 | 89.968 |
| T9 | A1 | 2 | 2.077 | 3 | 11.441 | 65.058 | 89.968 |
| | A2 | 2 | 2.077 | 3 | 11.441 | 72.169 | 89.968 |
| | A3 | 2 | 2.077 | 3 | 11.441 | 72.169 | 89.968 |
| | A4 | 2 | 2.077 | 3 | 11.441 | 72.169 | 89.968 |
| | A5 | 3 | 3 | 3 | 11.524 | 72.178 | 89.968 |
| | A6 | 2 | 2.077 | 3 | 11.441 | 72.037 | 89.968 |

Tabela 5.7 PAE: tempos de execução (segundos) do modelo lexicográfico

| Teste | Tempo | | | Teste | Tempo | | |
|-------|-------|--------|-------|-------|-------|--------|-------|
| | CPLEX | Alg. 6 | Total | | CPLEX | Alg. 6 | Total |
| T1 | 0.10 | 2.11 | 2.21 | T6 | 0.45 | 4.36 | 4.81 |
| T2 | 0.17 | 2.70 | 2.87 | T7 | 0.46 | 10.81 | 11.27 |
| T3 | 0.38 | 2.77 | 3.15 | T8 | 1.51 | 10.57 | 12.08 |
| T4 | 0.20 | 3.84 | 4.04 | T9 | 1.34 | 11.06 | 12.40 |
| T5 | 0.26 | 4.61 | 4.87 | | | | |

Tabela 5.8 PAE: tempos de execução (segundos) das várias abordagens

| Teste | Abordagem | | | | | |
|-------|----------------|---------------|--------------|--------|-------------|--------------|
| | A1 | A2 | A3 | A4 | A5 | A6 |
| T1 | 1.45 | 1.94 | 1.85 | 1.83 | 1.81 | <u>2.21</u> |
| T2 | <u>13.65</u> | 9.93 | 9.86 | 7.05 | 1.94 | 2.87 |
| T3 | <u>1757.20</u> | 243.02 | 302.51 | 76.31 | 2.55 | 3.15 |
| T4 | <u>3.86</u> | 3.80 | 3.84 | 3.93 | 3.71 | 4.04 |
| T5 | 10.46 | 56.07 | <u>57.25</u> | 30.12 | 4.39 | 4.87 |
| T6 | 18.44 | <u>39.03</u> | 26.71 | 31.81 | 5.76 | 4.81 |
| T7 | <u>13.03</u> | 12.20 | 12.49 | 9.66 | 9.45 | 11.27 |
| T8 | 300.46 | <u>318.73</u> | 312.84 | 334.93 | 13.96 | 12.08 |
| T9 | 47.49 | 48.87 | <u>55.30</u> | 54.15 | 13.97 | 12.40 |

Tabela 5.9 PAE: tempos de execução (segundos) do Algoritmo 5

| Teste | Tempo | Teste | Tempo | Teste | Tempo |
|-------|-------|-------|-------|-------|-------|
| T1 | 1.73 | T4 | 3.82 | T7 | 10.72 |
| T2 | 2.74 | T5 | 3.91 | T8 | 10.69 |
| T3 | 2.74 | T6 | 4.28 | T9 | 11.14 |

Tabela 5.10 Áreas de distorção do objeto

| Teste | Abordagem | $A_{int}(\%)$ | $\mu_{int}(\%)$ | $\bar{A}_{int}(\%)$ | $A_{ext}(\%)$ | $\mu_{ext}(\%)$ | $\bar{A}_{ext}(\%)$ |
|-------|-----------|---------------|-----------------|---------------------|---------------|-----------------|---------------------|
| T1 | A1 | 43.41 | 74.51 | 80.37 | 0 | 1.81 | 26.13 |
| | A2 | 43.55 | 76.19 | 80.37 | 0 | 1.05 | 23.78 |
| | A3 | 43.55 | 76.19 | 80.37 | 0 | 1.05 | 23.78 |
| | A4 | 43.55 | 76.19 | 80.37 | 0 | 1.05 | 23.78 |
| | A5 | 43.55 | 76.19 | 80.37 | 0 | 1.05 | 23.78 |
| | A6 | 43.55 | 76.19 | 80.37 | 0 | 1.05 | 23.78 |
| T2 | A1 | 43.41 | 66.49 | 80.37 | 0 | 5.07 | 26.13 |
| | A2 | 43.41 | 74.57 | 80.37 | 0 | 1.35 | 26.08 |
| | A3 | 43.41 | 74.57 | 80.37 | 0 | 1.35 | 26.08 |
| | A4 | 43.41 | 78.22 | 80.37 | 0 | 0.28 | 26.08 |
| | A5 | 43.47 | 78.37 | 80.37 | 0 | 0.21 | 24.91 |
| | A6 | 43.41 | 73.85 | 80.37 | 0 | 1.56 | 26.08 |
| T3 | A1 | 43.41 | 61.62 | 80.37 | 0 | 8.31 | 26.13 |
| | A2 | 43.41 | 69.16 | 80.37 | 0 | 3.24 | 26.08 |
| | A3 | 43.41 | 72.43 | 80.37 | 0 | 1.83 | 26.08 |
| | A4 | 43.41 | 74.30 | 80.37 | 0 | 1.04 | 26.08 |
| | A5 | 43.57 | 74.69 | 80.37 | 0 | 0.70 | 23.54 |
| | A6 | 43.41 | 65.12 | 80.37 | 0 | 5.09 | 26.08 |
| T4 | A1 | 43.19 | 78.61 | 80.37 | 0 | 0.31 | 31.32 |
| | A2 | 43.35 | 79.52 | 80.37 | 0 | 0.01 | 28.30 |
| | A3 | 43.35 | 79.52 | 80.37 | 0 | 0.01 | 28.30 |
| | A4 | 43.35 | 79.52 | 80.37 | 0 | 0.01 | 28.30 |
| | A5 | 43.35 | 79.52 | 80.37 | 0 | 0.01 | 28.30 |
| | A6 | 43.35 | 79.52 | 80.37 | 0 | 0.01 | 28.30 |
| T5 | A1 | 58.39 | 74.76 | 80.37 | 0 | 0 | 0 |
| | A2 | 58.43 | 78.69 | 80.37 | 0 | 0 | 0 |
| | A3 | 58.43 | 78.69 | 80.37 | 0 | 0 | 0 |
| | A4 | 58.43 | 78.82 | 80.37 | 0 | 0 | 0 |
| | A5 | 58.45 | 79.25 | 80.37 | 0 | 0 | 0 |
| | A6 | 58.43 | 78.69 | 80.37 | 0 | 0 | 0 |
| T6 | A1 | 50.77 | 70.56 | 80.37 | 0 | 0.02 | 18.19 |
| | A2 | 50.96 | 74.85 | 80.37 | 0 | 0 | 18.02 |
| | A3 | 50.96 | 74.85 | 80.37 | 0 | 0 | 18.02 |
| | A4 | 50.96 | 74.85 | 80.37 | 0 | 0 | 18.02 |
| | A5 | 51.84 | 74.85 | 80.37 | 0 | 0 | 17.28 |
| | A6 | 50.96 | 74.84 | 80.37 | 0 | 0 | 18.02 |
| T7 | A1 | 47.70 | 79.17 | 80.37 | 0 | 0 | 00.03 |
| | A2 | 49.52 | 79.65 | 80.37 | 0 | 0 | 00.02 |
| | A3 | 49.52 | 79.65 | 80.37 | 0 | 0 | 00.02 |
| | A4 | 49.52 | 79.65 | 80.37 | 0 | 0 | 00.02 |
| | A5 | 49.52 | 79.65 | 80.37 | 0 | 0 | 00.02 |
| | A6 | 49.52 | 79.65 | 80.37 | 0 | 0 | 00.02 |
| T8 | A1 | 42.88 | 77.35 | 80.37 | 0 | 0.30 | 41.86 |
| | A2 | 42.92 | 79.01 | 80.37 | 0 | 0 | 40.44 |
| | A3 | 42.92 | 79.01 | 80.37 | 0 | 0 | 40.44 |
| | A4 | 42.92 | 79.01 | 80.37 | 0 | 0 | 40.44 |
| | A5 | 42.92 | 79.01 | 80.37 | 0 | 0 | 40.44 |
| | A6 | 42.88 | 78.93 | 80.37 | 0 | 0.01 | 41.81 |
| T9 | A1 | 42.88 | 74.45 | 80.37 | 0 | 0.29 | 41.86 |
| | A2 | 42.88 | 78.00 | 80.37 | 0 | 0 | 41.86 |
| | A3 | 42.88 | 78.00 | 80.37 | 0 | 0 | 41.86 |
| | A4 | 42.88 | 78.00 | 80.37 | 0 | 0 | 41.86 |
| | A5 | 42.89 | 78.01 | 80.37 | 0 | 0 | 41.86 |
| | A6 | 42.88 | 77.85 | 80.37 | 0 | 0.02 | 41.86 |

Capítulo 6

Conclusão

Neste trabalho estudou-se a possibilidade de resolver o problema de impressão tridimensional bimaterial com recurso à colocação de sistemas de espelhos galvanométricos nas paredes laterais de uma impressora. Primeiro foram aprofundados alguns aspetos importantes a ter em conta para a resolução do problema, destacando-se a discretização do espaço de impressão e o cálculo dos *voxels* que são atingidos a partir de uma possível localização de um emissor. Estabelecidos estes aspetos, foi feita uma divisão do problema em dois: o problema de localização de emissores e o problema de afetação de emissores. O primeiro foi modelado como um problema de cobertura. Para o segundo apresentou-se uma formulação como um programa linear inteiro considerando-se dois critérios, o que levou à introdução de duas abordagens para o resolver. A primeira foi baseada no método de soma pesada e a segunda num critério lexicográfico. Por fim, foi descrito um caso de estudo e apresentados os resultados computacionais correspondentes em termos da solução encontrada e em termos de tempos de execução. Para avaliação das soluções calculou-se a área atingida com luz *laser*.

De acordo com a experiência computacional efetuada, a utilização de sistemas de espelhos adicionais é uma hipótese viável para o processo de impressão tridimensional bimaterial. Para todos os testes foram obtidas soluções teóricas que satisfazem as restrições impostas. Como medidas de avaliação da qualidade das soluções calcularam-se a área polimerizada indevidamente fora dos *voxels* previstos e a área dos *voxels* previstos que fica por polimerizar. Ambas dependem do ângulo com que a luz *laser* incide no *voxel*. Os valores da área externa aos *voxels* foram relativamente pequenos. Em termos matemáticos este problema poderá ser resolvido através da inclusão de novas restrições no modelo apresentado, limitando a altura a que os sistemas de espelhos podem estar colocados. Em termos de implementação do sistema é de referir que os processos atuais de impressão tridimensional incluem uma fase de acabamento pós-impressão, que pode eventualmente permitir um aperfeiçoamento da peça produzida. Por outro lado, os valores da área que fica por cobrir nos *voxels* rondaram os 70%, o que parece ser elevado. Melhorá-los implicaria prescindir da simplificação de considerar que o *laser* apenas atinge o centro dos *voxels*.

Para os testes considerados foi possível encontrar soluções exatas resolvendo os problemas lineares binários com recurso ao *software* CPLEX. No entanto, para que isto não fosse uma limitação e prevenir casos em que tal não aconteça ou quando não haja acesso a *software* específico para programação matemática, também foram propostas e testadas alternativas. Para o PLE, apresentaram-se duas heurísticas para resolver de forma aproximada o problema (3.1). No caso do PAE, para o modelo de

soma pesada, apresentou-se um algoritmo para resolver de forma exata o modelo (4.8), quando $\lambda_1 = 0$ e $\lambda_2 = 1$. Além disso, para o modelo lexicográfico, podem ser aplicadas as heurísticas apresentadas para o PLE, aplicando-as individualmente a cada camada, seguido do Algoritmo 6.

Além do estudo dos aspetos referidos acima, no trabalho futuro pretende-se incluir testes para casos de estudo de outros objetos.

Bibliografia

- [1] Balas, E. and Carrera, M. C. (1996). A dynamic subgradient-based branch-and-bound procedure for set covering. *Operations Research*, 44(6):875–890.
- [2] Balas, E. and Ho, A. (1980). Set covering algorithms using cutting planes, heuristics, and subgradient optimization: a computational study. *Combinatorial Optimization*, pages 37–60.
- [3] Beasley, J. (1990). A lagrangian heuristic for set-covering problems. *Naval Research Logistics*, 37(1):151–164.
- [4] Beasley, J. E. (1987). An algorithm for set covering problem. *European Journal of Operational Research*, 31(1):85–93.
- [5] Burns, M. (1993). *Automated fabrication: improving productivity in manufacturing*. Prentice-Hall, Inc.
- [6] Caprara, A., Toth, P., and Fischetti, M. (2000). Algorithms for the set covering problem. *Annals of Operations Research*, 98(1):353–371.
- [7] Chvátal, V. (1979). A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4(3):233–235.
- [8] Ehrgott, M. (2006). *Multicriteria optimization*. Springer Science & Business Media, Berlin, Heidelberg.
- [9] Ehrgott, M., Güler, Ç., Hamacher, H. W., and Shao, L. (2008a). Mathematical optimization in intensity modulated radiation therapy. *4OR*, 6(3):199–262.
- [10] Ehrgott, M., Güler, Ç., W. Hamacher, H., and Shao, L. (2010). Mathematical optimization in intensity modulated radiation therapy. *Annals of Operations Research*, 175(1):309–365.
- [11] Ehrgott, M., Holder, A., and Reese, J. (2008b). Beam selection in radiotherapy design. *Linear Algebra and its Applications*, 428(5-6):1272–1312.
- [12] Gibson, I., Rosen, D., and Stucker, B. (2015). *Additive manufacturing technologies*. Springer, New York.
- [13] Gobbetti, E. and Marton, F. (2005). Far voxels: a multiresolution framework for interactive rendering of huge complex 3d models on commodity graphics platforms. *ACM Transactions on Graphics (TOG)*, 24(3):878–885.
- [14] Jacobs, P. F. (1992). *Rapid prototyping & manufacturing: fundamentals of stereolithography*. Society of Manufacturing Engineers.
- [15] Karp, R. M. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer.
- [16] King, D. C., Lillis, R. M., and White, G. (1993). Process for the injection moulding of multi-layered articles. US Patent 5,188,787.

-
- [17] Sonderman, D. and Abrahamson, P. G. (1985). Radiotherapy treatment design using mathematical programming models. *Operations Research*, 33(4):705–725.
- [18] Vasko, F. J. (1984). An efficient heuristic for large set covering problems. *Naval Research Logistics*, 31(1):163–171.
- [19] Vasko, F. J., Lu, Y., and Zyma, K. (2016). What is the best greedy-like heuristic for the weighted set covering problem? *Operations Research Letters*, 44(3):366–369.
- [20] Vasko, F. J. and Wilson, G. R. (1986). Hybrid heuristics for minimum cardinality set covering problems. *Naval Research Logistics Quarterly*, 33(2):241–249.