



**UNIVERSIDADE FEDERAL DO CARIRI
CENTRO DE CIÊNCIAS E TECNOLOGIA
CURSO DE GRADUAÇÃO EM ENGENHARIA CIVIL**

ANTÔNIO MARCOS CRUZ DA PAZ

**POSICIONAMENTO AUTOMÁTICO DE MÓDULOS FOTOVOLTAICOS EM
TELHADOS CONSIDERANDO EFEITOS DE SOMBREAMENTO**

**JUAZEIRO DO NORTE
2022**

ANTÔNIO MARCOS CRUZ DA PAZ

**POSICIONAMENTO AUTOMÁTICO DE MÓDULOS FOTOVOLTAICOS EM TELHADOS
CONSIDERANDO EFEITOS DE SOMBREAMENTO**

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia Civil do
Centro de Ciências e Tecnologia da Universi-
dade Federal do Cariri, como requisito parcial à
obtenção do grau de bacharel em Engenharia
Civil.

Orientador: Prof. Dr. Vicente Helano
Feitosa Batista Sobrinho

JUAZEIRO DO NORTE

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Cariri
Sistema de Bibliotecas

P348p Paz, Antônio Marcos Cruz da.

Posicionamento automático de módulos fotovoltaicos em telhados considerando efeitos de sombreamento / Antônio Marcos Cruz da Paz. – 2022.

119 f.: il. color.30 cm.

Monografia (graduação) - Universidade Federal do Cariri, Curso de Engenharia Civil, Juazeiro do Norte, 2022.

Orientação: Prof. Dr. Vicente Helano Feitosa Batista Sobrinho.

1. Energia solar. 2. Python. 3. Sistemas fotovoltaicos - edificações. I.Título.

CDD 627

Bibliotecária: Glacinésia Leal Mendonça
CRB 3/ 925

ANTÔNIO MARCOS CRUZ DA PAZ

**POSICIONAMENTO AUTOMÁTICO DE PLACAS SOLARES EM TELHADOS
CONSIDERANDO EFEITOS DE SOMBREAMENTO**

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia Civil do
Centro de Ciências e Tecnologia da Universi-
dade Federal do Cariri, como requisito parcial à
obtenção do grau de bacharel em Engenharia
Civil.

Aprovada em: 22 de dezembro de 2022

BANCA EXAMINADORA

Prof. Dr. Vicente Helano Feitosa Batista Sobrinho (Orientador)
Universidade Federal do Cariri (UFCA)

Prof. Dr. Paulo Roberto Lacerda Tavares
Universidade Federal do Cariri (UFCA)

Prof. Dr. Rafael Perazzo Barbosa Mota
Universidade Federal do Cariri (UFCA)

Eng. André Wagner de Barros Silva
Universidade Federal do Cariri (UFCA)

AGRADECIMENTOS

À Deus, pelo dom da vida.

Ao meu pai Antônio e minha mãe Solange, professores, por promoverem a minha educação e a de inúmeras outras pessoas.

À minha irmã Soylle, formanda em Jornalismo, com a qual tenho muito orgulho de dividir a jornada acadêmica.

Aos amigos Davi Freitas, José Matheus e Vitória Tiffany, que conheci na graduação, pela parceria e pela inspiração.

Ao meu orientador Prof. Vicente Helano, pela dedicação em me ajudar a construir este trabalho, como também aos professores Rafael Perazzo, Maxwell Oliveira e Roberto Hugo, que conseguiram despertar em mim o gosto pela programação.

Aos meus professores de Matemática e suas tecnologias, como também de Física, do ensino fundamental e médio, por serem decisivos na escolha da minha carreira profissional.

À PROJETTA Empresa Júnior de Engenharia Civil, pela importância na minha formação profissional.

Forever in debt to your priceless advice

(Kurt Cobain)

RESUMO

A possibilidade de tornar edificações autossuficientes em energia elétrica tem se tornado uma realidade com os avanços das tecnologias relacionadas a módulos fotovoltaicos. O Brasil possui um grande potencial de produção dessa matriz renovável, mas existem inúmeros fatores que podem prejudicar a geração de energia a partir de módulos fotovoltaicos, entre eles, o sombreamento dos módulos. O levantamento das sombras que podem incidir sobre a região pretendida para a instalação dos módulos pode ser realizado de várias maneiras, a partir de visitas a campo e/ou remotamente, com o auxílio de softwares, aplicando-se modelos tridimensionais. O presente trabalho visa desenvolver uma metodologia computacional para a delimitação das regiões sombreadas em uma edificação, a partir de modelos tridimensionais, utilizando um algoritmo de traçado de raios adaptado para o trabalho, desenvolvido em Python, e a alocação automática dos módulos, a partir de uma heurística de empacotamento, de forma a tornar o arranjo dos módulos o menos sombreado possível. A metodologia computacional aplicada no presente trabalho teve sucesso em produzir imagens a partir do traçado de raios em um tempo menor do que o necessário para se obter as mesmas observações em campo, e a partir de considerações apropriadas para o problema, a heurística de empacotamento pôde produzir arranjos otimizados em tempos menores que 1 hora.

Palavras-chave: Energia solar; Colorização; Programação; Python; Traçado de raios; Empacotamento; Sistemas fotovoltaicos integrados a edificações.

ABSTRACT

The possibility of making buildings autossufficient in electricity has become a reality, due to the breakthroughs on technology regarding photo-voltaic modules. Brazil has huge potential on the production of this renewable source, but there are inumerous factors that can harm the power generation on PV modules, among many, the shading of the modules. The verification of the shadows that can reach the intended region for the installation of mudules can be done using many methods, on field observations, or remotely, aided by software and using tridimensional models. The present monography aims to develop computational methodologies for the delimitation of the shaded areas on a building, using tridimensional models and a ray tracing algorithm developed to this problem, written in Python programming language, and the automatic placing of the modules, via packing heuristics, in order to make the arrangement of the modules as little shaded as possible. The computational methodology applied had success in rendering ray traced images in a shorter time than is necessary to obtain the same observations in the field, and due to problem-specific considerations, the packing heuristic was able to return optimized module arrangements in less than an hour.

Keywords: Solar Energy; shading; programming; Python; ray tracing; packing; Building-integrated photovoltaic systems.

SUMÁRIO

| | | |
|--------------|---|-----------|
| 1 | INTRODUÇÃO | 21 |
| 1.1 | Objetivos | 22 |
| 1.1.1 | <i>Objetivo Geral</i> | 22 |
| 1.1.2 | <i>Objetivos Específicos</i> | 22 |
| 1.2 | Organização do Trabalho | 23 |
| 2 | REFERENCIAL TEÓRICO | 24 |
| 2.1 | Energia Solar e Módulos Fotovoltaicos | 24 |
| 2.2 | Efeito do Sombreamento em Módulos Fotovoltaicos | 25 |
| 2.3 | Trajetória Solar | 27 |
| 2.4 | Determinação do Sombreamento do Ambiente | 32 |
| 2.4.1 | <i>Análise do local e dos obstáculos</i> | 32 |
| 2.4.2 | <i>Análise da trajetória solar usando folha de acetato</i> | 32 |
| 2.4.3 | <i>Equipamentos e softwares para análise de sombreamento</i> | 33 |
| 2.5 | Síntese de imagens digitais | 35 |
| 2.5.1 | <i>Traçado de Raios</i> | 37 |
| 2.5.2 | <i>Colorização de pixels</i> | 41 |
| 2.6 | Algoritmos e heurísticas de otimização | 44 |
| 2.7 | Trabalhos Relacionados | 48 |
| 3 | METODOLOGIA | 51 |
| 3.1 | Construção do campo de sombreamento | 51 |
| 3.1.1 | <i>Modelo geométrico tridimensional</i> | 51 |
| 3.1.2 | <i>Azimutes e elevações solares</i> | 53 |
| 3.1.3 | <i>Funcionamento do traçado de raios</i> | 53 |
| 3.1.4 | <i>Validação do modelo de traçado de raios</i> | 61 |
| 3.2 | Alocação automática de módulos fotovoltaicos | 69 |
| 3.2.1 | <i>Considerações sobre a heurística desenvolvida</i> | 69 |
| 3.2.2 | <i>Modalidades de inserção de módulos</i> | 81 |
| 3.2.3 | <i>Validação da heurística desenvolvida</i> | 86 |
| 4 | RESULTADOS | 91 |
| 4.1 | Exemplo 1 | 91 |

| | | |
|------------|--|-----|
| 4.2 | Exemplo 2 | 101 |
| 4.3 | Sobre o coeficiente de sombreamento | 107 |
| 4.4 | Sobre a espessura do módulo | 110 |
| 4.5 | Sobre o desempenho computacional | 111 |
| 5 | CONSIDERAÇÕES FINAIS | 113 |
| | REFERÊNCIAS | 115 |

LISTA DE ILUSTRAÇÕES

| | |
|---|----|
| Figura 1 – Hierarquização do agrupamento de células fotovoltaicas em módulos e arranjos. | 25 |
| Figura 2 – Ligação mista de módulos em um arranjo. | 26 |
| Figura 3 – Esquema de ligação dos diodos de fileira e de by-pass. | 27 |
| Figura 4 – Declinação solar ao longo das estações do ano. | 28 |
| Figura 5 – Representação do azimute θ_A , da elevação solar θ_{el} e do zênite solar θ_z . | 28 |
| Figura 6 – Diferenças entre o sistema de azimute e o círculo trigonométrico. | 30 |
| Figura 7 – Norte com orientação diferente ao plano de trabalho. | 31 |
| Figura 8 – Ângulo de desvio do norte em relação ao plano de trabalho $\theta_{A_{off}}$. | 31 |
| Figura 9 – Determinação dos ângulos de azimute e de elevação de um elemento gerador de sombreamento incidente em um sistema fotovoltaico. | 32 |
| Figura 10 – Demonstração do uso da folha de acetato e do dispositivo “SunEye”, que mostra automaticamente a trajetória solar. | 33 |
| Figura 11 – Solar Pathfinder | 34 |
| Figura 12 – Exemplo de modelagem no PVsyst. | 35 |
| Figura 13 – Reprodução de sombras em centro comercial de Cingapura a partir do <i>ECT Analysis</i> . | 35 |
| Figura 14 – Imagem na forma vetorial e na forma matricial. | 37 |
| Figura 15 – Reta convertida em uma imagem matricial por meio de um algoritmo de rasterização. | 38 |
| Figura 16 – Diferenças entre a projeção paralela e a projeção perspectiva. | 39 |
| Figura 17 – Possíveis caminhos dos raios de luz em um algoritmo básico de traçado de raios: (1) parte do observador, (2) atinge um ponto visível, não sombreado, (3) e (4) atinge pontos sombreados, (5) não intercepta objeto. | 40 |
| Figura 18 – Esfera representada por uma malha triangular, em quatro níveis de refinamento. | 41 |
| Figura 19 – Diferença na reflexão de um raio de luz em um metal, considerando uma superfície suave e uma superfície rugosa. | 42 |
| Figura 20 – Representação de um raio luminoso I refletido pela superfície em R, com N sendo a normal da superfície, V a direção de visualização do observador, e β o ângulo entre o observador e o raio refletido. | 43 |

| | |
|--|----|
| Figura 21 – Representação de um raio luminoso l refletido pela superfície na direção da região amarela, com v sendo a direção de visualização do observador, n sendo a normal da superfície, h o vetor na bissetriz de l e v , e α o ângulo entre n e h . | 44 |
| Figura 22 – Exemplificação visual de um problema de <i>strip packing</i> bidimensional. | 46 |
| Figura 23 – Considerações sobre a ordem de inserção em um algoritmo de empacotamento. | 46 |
| Figura 24 – Exemplo prático de um problema MPLP. | 47 |
| Figura 25 – Modelagem do escritório verde da UTFPR no software <i>SketchUp</i> . | 48 |
| Figura 26 – Resultados de inserção de módulos obtido por Marinosky <i>et al.</i> (2004). | 49 |
| Figura 27 – Malha de sombreamento em Cingapura, obtida a partir de dados do <i>ECOTECT</i> . | 50 |
| Figura 28 – Estrutura de um arquivo .obj de um cubo. | 52 |
| Figura 29 – Configuração necessária para exportação do arquivo OBJ no Blender com a orientação correta dos eixos coordenados. | 55 |
| Figura 30 – Exemplo de imagens de um modelo produzidas com diferentes valores de γ_p . | 56 |
| Figura 31 – Fluxograma do algoritmo implementado para produzir o campo de sombreamento. | 61 |
| Figura 32 – Primeiro modelo para validação do traçado de raios. | 62 |
| Figura 33 – Segundo modelo para validação do traçado de raios. | 63 |
| Figura 34 – Resultados do primeiro modelo de validação. | 64 |
| Figura 35 – Resultados do segundo modelo de validação. | 64 |
| Figura 36 – Terceiro modelo para validação do traçado de raios. | 66 |
| Figura 37 – Fotos tiradas durante a observação do terceiro modelo. | 66 |
| Figura 38 – Resultados visuais do terceiro modelo de validação. | 68 |
| Figura 39 – Hipótese de Dür e Nowak (2008) sobre a quantidade de subdivisões de uma placa | 70 |
| Figura 40 – Pixel base para a inserção de uma placa na heurística proposta. | 70 |
| Figura 41 – Diferença causada pela obrigatoriedade de dimensões ímpares de espaços. | 71 |
| Figura 42 – Estratégia de planificação do problema. | 72 |
| Figura 43 – Ilustração da dificuldade no tratamento de telhados com duas águas. | 73 |
| Figura 44 – Mapa de sombreamento do exemplo. | 74 |
| Figura 45 – Estado inicial dos mapa de sombreamento e locação. | 75 |
| Figura 46 – Pixel atual na verificação da imagem para locação da placa. | 75 |
| Figura 47 – Novo estado das matrizes após a inserção da placa da Figura 46. | 76 |

| | |
|---|----|
| Figura 48 – Segunda posição para inserção de placa. | 76 |
| Figura 49 – Recorte da região indicada na Figura 48. | 77 |
| Figura 50 – Terceira posição para inserção de placa. | 77 |
| Figura 51 – Recorte da região indicada na Figura 50. | 78 |
| Figura 52 – Exemplificação do cálculo do coeficiente de sombreamento da placa da Figura 50. | 79 |
| Figura 53 – Heurística clássica de bottom-left. | 79 |
| Figura 54 – Casos possíveis de travessia da região viável. | 80 |
| Figura 55 – Dois arranjos de placas com o mesmo coeficiente de sombreamento, mas partindo de extremidades diferentes. | 81 |
| Figura 56 – Formação da grade na modalidade alinhada. | 83 |
| Figura 57 – Grade de pixels viáveis, numerados pela ordem estabelecida no mapeamento. | 83 |
| Figura 58 – Formação de novas grades a partir do deslocamento da origem. | 85 |
| Figura 59 – Grades idênticas com origens distintas, indicadas em rosa. | 85 |
| Figura 60 – Número de resultados possíveis para formação do grid. | 86 |
| Figura 61 – Medidas estimadas para os Exemplos 3 e 4 de Dür e Nowak (2008) com o auxílio do Autodesk AutoCAD e do Autodesk Revit. | 87 |
| Figura 62 – Resultados obtidos para o Exemplo 3 segundo Dür e Nowak (2008) e a heurística proposta neste trabalho. | 88 |
| Figura 63 – Resultados obtidos para o Exemplo 4 segundo Dür e Nowak (2008) e a heurística proposta neste trabalho. | 89 |
| Figura 64 – Planta do modelo do Exemplo 1. | 92 |
| Figura 65 – Corte AA do modelo do Exemplo 1. | 92 |
| Figura 66 – Vista 3D do modelo do Exemplo 1. | 93 |
| Figura 67 – Mapa de sombreamento combinado das águas esquerda e direita do telhado do Exemplo 1. | 94 |
| Figura 68 – Teste na água direita do Exemplo 1 para as duas orientações da placa. | 95 |
| Figura 69 – Mapa de locação combinado das águas esquerda e direita do telhado do Exemplo 1. | 97 |
| Figura 70 – Ilustração tridimensional da locação de placas solares obtida para o Exemplo 1. | 97 |
| Figura 71 – Mapa de locação de placas para uma eventual expansão do sistema do Exemplo 1. | 98 |

| | |
|--|-----|
| Figura 72 – Representação da trajetória solar no Exemplo 1. | 99 |
| Figura 73 – Trajetória solar para Juazeiro do Norte/CE. | 100 |
| Figura 74 – Planta do modelo do Exemplo 2. | 101 |
| Figura 75 – Corte AA do modelo do Exemplo 2. | 102 |
| Figura 76 – Vista 3D do modelo do Exemplo 2. | 102 |
| Figura 77 – Mapa de sombreamento da região viável do Exemplo 2. | 103 |
| Figura 78 – Resultado da alocação de placas do Exemplo 2. | 104 |
| Figura 79 – Imagens geradas para a primeira alocação do Exemplo 2, considerando as 10h do dia 21/06/2022. | 105 |
| Figura 80 – Sombreamento devido a oclusões ocasionadas por placas próximas umas das outras. | 105 |
| Figura 81 – Resultado da alocação de placas do Exemplo 2, com espaçamento adicional de 12 pixels. | 106 |
| Figura 82 – Imagens geradas da alocação com espaçamento adicional para o Exemplo 2, para as 10h do dia 21/06/2022. | 106 |
| Figura 83 – Sobreposição das placas locadas ao mapa de sombreamento da alocação do Exemplo 2 com espaçamento adicional. | 107 |
| Figura 84 – Exemplo de módulos com o mesmo coeficiente de sombreamento em períodos diferentes. | 108 |
| Figura 85 – Exemplo de placas com o mesmo coeficiente de sombreamento em diferentes manchas de sombreamento. | 109 |
| Figura 86 – Majoração do sombreamento ao se desconsiderar a espessura <i>esp</i> da placa. . | 110 |

LISTA DE ALGORITMOS

| | |
|---|----|
| Algoritmo 1 – Determinação das coordenadas da tela de projeção. | 56 |
| Algoritmo 2 – Delimitação da região viável. | 58 |
| Algoritmo 3 – Produção de imagens a partir do traçado de raios. | 59 |
| Algoritmo 4 – Determinação das áreas sombreadas. | 60 |
| Algoritmo 5 – Pseudo-código para a modalidade livre, inserindo apenas nas áreas não sombreadas. | 82 |
| Algoritmo 6 – Heurística de locação de placas solares na modalidade alinhada. | 84 |

LISTA DE QUADROS

Quadro 1 – Comparativo entre as execuções do traçado de raios dos Exemplos 1 e 2 . . 111

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 – Informações de elevação e azimute para o dia 20/09/2022. | 62 |
| Tabela 2 – Informações de elevação e azimute para o dia 21/10/2022. | 65 |
| Tabela 3 – Comparativo entre as áreas sombreadas obtidas. | 67 |
| Tabela 4 – Quantidade de placas alocadas e tempo de execução relativos às heurísticas de Dür e Nowak (2008) e deste trabalho, aplicadas aos Exemplos 3 e 4. . . . | 90 |
| Tabela 5 – Informações de elevação e azimute solar, em graus, para Juazeiro do Norte-CE | 94 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|----------------|--|
| ANEEL | Agência Nacional de Energia Elétrica |
| API | Application Programming Interface (Interface de Programação de Aplicações) |
| BIM | Building Information Modeling (Modelagem da Informação da Construção) |
| CAD | Computer Aided Design (Desenho Assistido por Computador) |
| CIE-RGB | Comissão Internacional de Iluminação - Vermelho, Verde e Azul |
| CMYK | Cian, Magenta, Yellow e black Key (ciano, magenta, amarelo e preto) |
| CPU | Central Processing Units (Unidades de processamento central) |
| CREA-SC | Conselho Regional de Engenharia, Arquitetura e Agronomia de Santa Catarina |
| dpi | dots per inch (pontos por polegada) |
| FBX | FilmBox File (Arquivo FilmBox) |
| GPS | Global Positioning System (Sistema de Posicionamento Global) |
| GPU | Graphics Processing Units (Unidades de processamento gráfico) |
| GW | Gigawatt |
| HD | High Definition (Alta definição) |
| HSV | Hue, Saturation, Value (matiz, saturação e valor) |
| id | Identificação numérica da placa |
| IFC | Industry Foundation Classes (Classes de Base da Indústria) |
| IIPP | Identical Item Packing Problem (problema de empacotamento de itens idênticos) |
| MPLP | Manufaturer's Pallet Loading Packing Problem (Problema de Carregamento de Paletes do Fabricante) |

| | |
|-------|---|
| OBJ | Object File (Arquivo de Objeto) |
| OOP | Object Oriented Programming (Programação orientada a objetos) |
| PIL | Python Imaging Library (Biblioteca de imagem do Python) |
| PNG | Portable Network Graphic (Gráfico de rede portátil) |
| ppi | pixels per inch (pixels por polegada) |
| PV | Photo-voltaic (fotovoltaico) |
| RGB | Red, Green, Blue (vermelho, verde, azul) |
| RGBA | Red, Green, Blue, Alpha (vermelho, verde, azul, alfa) |
| UFRGS | Universidade Federal do Rio Grande do Sul |
| UHD | Ultra High Definition (Ultra alta definição) |
| UTFPR | Universidade Tecnológica Federal do Paraná |

LISTA DE SÍMBOLOS

| | |
|-------------------------|---|
| θ_A | Azimute Solar |
| θ_{el} | Elevação Solar |
| θ_z | Zênite Solar |
| $\theta_{A\text{trig}}$ | Ângulo do azimute no círculo trigonométrico |
| $\theta_{A\text{off}}$ | Ângulo de desvio do norte em relação ao plano de trabalho |
| p | Ponto de destino de um raio representado como uma reta paramétrica |
| e | Origem de um raio representado como uma reta paramétrica |
| t | Distância percorrida por um raio representado como uma reta paramétrica |
| d | Direção de um raio representado como uma reta paramétrica |
| r | Ponto que direciona um raio representado como uma reta paramétrica |
| a, b, c | Vértices de um triângulo genérico |
| ω, γ | Coeficientes das coordenadas baricêntricas de um triângulo genérico |
| L_d | Cor de um pixel referente à componente difusa |
| k_d | Coeficiente de difusão ou cor difusa |
| I_d | Intensidade da luz difusa |
| θ | Ângulo que o raio de luz faz com a normal da superfície |
| n | Vetor normal unitário à superfície |
| ℓ | Vetor unitário da direção da luz |
| L_s | Cor de um pixel referente à componente especular |
| k_s | Coeficiente especular ou cor especular |
| I_s | Intensidade da luz especular |
| ρ | Expoente de Phong |
| β | Ângulo entre a direção de visualização do observador e a direção de reflexão |
| v | Vetor da direção de visualização |
| h | Vetor situado na bissetriz entre a direção de visualização e a direção da luz |
| L_a | Cor de um pixel referente à luz ambiente |

| | |
|--------------------------|---|
| L_{final} | Cor final resultante da soma das componentes difusa, especular e ambiente |
| V | Lista de vértices |
| T | Lista de todos os triângulos da cena |
| ℓ_x, ℓ_y, ℓ_z | Componentes do vetor unitário da direção da luz |
| p_a | Coeficiente de ponderação da componente da luz ambiente |
| p_d | Coeficiente de ponderação da componente difusa |
| C | Cor RGBA de um triângulo ou objeto |
| ε | “Forramento” das bordas da imagem |
| τ | Valor a partir do qual uma distância é considerada infinita no programa produzido |
| X_{\max} | Maior coordenada x da tela |
| X_{\min} | Menor coordenada x da tela |
| Y_{\max} | Maior coordenada y da tela |
| Y_{\min} | Menor coordenada y da tela |
| Z_{\max} | Maior coordenada z da tela |
| γ_p | Densidade de pixels |
| Q_h | Quantidade de pixels no sentido horizontal da tela |
| Q_v | Quantidade de pixels no sentido vertical da tela |
| M_{coord} | Matriz das coordenadas dos pixels da tela |
| RV | Matriz da região viável |
| T_{rv} | Lista de triângulos dentro da região viável |
| MS | Matriz com os valores para uma imagem do mapa de sombreamento |
| H | Matriz do mapa de sombreamento |
| P_t | Intervalos do dia considerados no campo escalar de sombreamento |
| dim_{proj} | Dimensão da placa projetada na imagem |
| dim_{incl} | Dimensão da placa a ser inclinada na projeção da imagem |
| α | Inclinação da placa na área de interesse |
| ID | Identificação numérica da placa |

| | |
|----------|---|
| s | Coeficiente de sombreamento da placa |
| i_0 | Menor índice vertical da placa na imagem |
| i_f | Maior índice vertical da placa na imagem |
| j_0 | Menor índice horizontal da placa na imagem |
| j_f | Maior índice horizontal da placa na imagem |
| S_{ij} | Valor no mapa de sombreamento correspondente a um certo pixel |
| L | Lista das placas locadas |
| ML | Matriz do mapa de locação |
| $count$ | Contador de índices na execução de algoritmos |
| GV | Grade regular auxiliar |
| N | Número de resultados possíveis na modalidade “alinhado” |
| n_x | Número de pixels em x da placa |
| n_y | Número de pixels em y da placa |
| n_p | Número de placas pré-determinado pelo usuário |
| esp | Espessura da placa |

1 INTRODUÇÃO

A utilização da matriz energética solar, no Brasil e no mundo, tem aumentado vertiginosamente, impulsionada pelo esgotamento de recursos minerais não renováveis, a conscientização ambiental e, principalmente, pelo aumento da tecnologia no setor, que passou a proporcionar equipamentos mais baratos, acessíveis e eficientes. Em maio de 2022, o Brasil superou a marca de 5 GW de capacidade instalada em usinas fotovoltaicas com geração centralizada, e 10,74 GW de capacidade instalada em unidades de micro e minigeração distribuída, que é aquela gerada pelos próprios consumidores (ANEEL, 2022a). A rápida ascenção desta matriz é evidenciada também por outros dados: estão sendo construídas usinas fotovoltaicas com o potencial de produzir 4,7 GW, com previsão de início de funcionamento entre 2022 e 2023 (ANEEL, 2022a), praticamente dobrando a capacidade instalada das usinas atualmente em funcionamento. Quanto à geração pelos próprios consumidores, basta se considerar que em 2019, três anos antes, a capacidade instalada em unidades de micro e minigeração distribuída era de apenas 1 GW, dez vezes menor que a atual (ANEEL, 2022b).

O estado do Ceará possui atualmente uma potência instalada de aproximadamente 380 MW, incluindo as seguintes modalidades de geração: autoconsumo remoto, geração compartilhada e geração na própria unidade consumidora, (ANEEL, 2022c). Como afirma Almeida e Almeida (2022), pela regularidade e disponibilidade da irradiação no nordeste, a escolha da energia solar é justificada, tendo reduzido custo de instalação e menor degradação ambiental.

A produção de energia a partir de módulos fotovoltaicos, porém, possui algumas limitações. Para Azêvedo (2020), vários fatores técnicos, climatológicos, ambientais e de localização afetam o desempenho energético do sistema, dentre os principais: azimute, ângulo de incidência, localização dos módulos no sistema, sombreamento, degradação, sujeira, entre outros, principalmente por influenciar a forma com que a irradiação solar é aproveitada pelo módulo.

Como afirma Ilse *et al.* (2019), o sombreamento parcial de um módulo fotovoltaico pode diminuir a produção de energia significativamente. Em usinas de maior porte, geralmente instaladas em grandes áreas planas, afastadas de grandes centros urbanos, o sombreamento dos módulos é menos recorrente. Porém, muitas situações podem causar o sombreamento total ou parcial de um módulo em usinas solares de geração distribuída, como a presença de árvores, estruturas da própria edificação (platibandas, castelos d'água, barriletes, entre outros), e em localidades com elevada densidade de construções, edificações adjacentes. A verticalização das

grandes cidades brasileiras, como também a ocupação desordenada, característica de grandes centros comerciais, pode apresentar um desafio à produção fotovoltaica na unidade consumidora, visto que estas unidades podem não receber radiação solar suficiente para suprir o seu consumo, sendo necessária a complementação da produção, por meio de usinas, fotovoltaicas ou não, afastadas da unidade consumidora, aumentando o custo econômico e ambiental da operação.

Como a grande maioria das edificações existentes não foi projetada de forma a aproveitar com eficiência a irradiação solar, a alocação de módulos fotovoltaicos nestas circunstâncias passa a ser uma questão de adaptação do espaço disponível à quantidade de módulos necessárias para suprir o consumo, balizando-se nas possíveis interferências que o sistema possa enfrentar.

1.1 Objetivos

1.1.1 Objetivo Geral

Baseando-se no que foi apresentado até então, o objetivo deste trabalho será o de utilizar metodologias que possam fornecer uma previsão da ocorrência de sombras em uma edificação ou região qualquer, por meio da modelagem da mesma e de informações da posição do sol, para auxiliar na determinação da localização ideal de módulos fotovoltaicos. Mais especificamente, serão utilizados conceitos de computação gráfica para produzir sombras a partir de um modelo tridimensional computadorizado, e a alocação dos módulos se dará por meio de uma heurística de otimização.

1.1.2 Objetivos Específicos

Para cumprir com o objetivo geral deste trabalho, foram estabelecidos os seguintes objetivos específicos:

- Implementar a técnica de traçado de raios para a construção de mapas de sombreamento a partir de modelos geométricos tridimensionais;
- Validar a implementação do traçado de raios utilizando um modelo em escala;
- Implementar uma heurística de otimização apropriada para a consideração do sombreamento, que possa utilizar os dados obtidos do traçado de raios;
- Validar a implementação da heurística de otimização adotada;
- Apresentar a aplicação da metodologia na forma de exemplos de execução.

1.2 Organização do Trabalho

O restante deste trabalho está organizado em cinco capítulos. O capítulo 2, intitulado de Referencial Teórico, aborda os estudos preliminares à aplicação dos métodos e as principais referências literárias que apoiam o trabalho. O capítulo 3, Metodologia, descreve de que maneira o problema levantado pelo trabalho será solucionado, considerando as especificidades do mesmo. O capítulo 4, Resultados e Discussões, aborda exemplos que servem para demonstrar a aplicação do software produzido. As Considerações Finais, capítulo 5, dizem respeito às problemáticas enfrentadas e das futuras aplicabilidades dos resultados deste trabalho.

2 REFERENCIAL TEÓRICO

Nesta seção, serão apresentados os principais conceitos que norteiam o trabalho, partindo de informações mais gerais sobre o uso de energia solar, até temas mais específicos que justificam a premissa apresentada, descrevendo metodologias utilizadas por outros autores para a caracterização do sombreamento em um empreendimento e introduzindo conceitos que serão importantes para a metodologia computacional aqui proposta, tais como traçado de raios e algoritmos de otimização.

2.1 Energia Solar e Módulos Fotovoltaicos

Como afirma Gómez *et al.* (2018), a energia solar é parte dos requisitos necessários para a existência de vida na Terra, dada a sua importância em processos químicos, físicos e biológicos. A interação com o sol tem influenciado inclusive a evolução dos seres que povoam a Terra. Há espécies que procuram o calor fornecido pela radiação para se manterem aquecidas e aquelas dotadas de visão, uma característica que eleva a possibilidade de reconhecimento do ambiente e, consequentemente, as chances de sobrevivência.

O uso da radiação solar como fonte de energia, na natureza, é principalmente representado pelo efeito da fotossíntese em plantas e algas. Como levantado por Kluge *et al.* (2015), a radiação liberada pelo sol desencadeia a ocorrência de reações fotossintéticas nas plantas, produzindo carboidratos importantes ao processo respiratório.

O efeito fotovoltaico foi observado pela primeira vez por Edmond Bequerel em 1839 (NASCIMENTO, 2004). Em um experimento, ele observou que havia uma diferença de potencial entre eletrodos imersos em uma solução ácida quando o sistema era iluminado (BÜHLER, 2016). Porém, a primeira célula solar fotovoltaica só seria produzida em 1954, por cientistas da Bell Labs, cuja eficiência era de apenas 6% (MACHADO; MIRANDA, 2015). Daí por diante, vários fatores aceleraram a evolução tecnológica no setor, como a corrida espacial e a crise do petróleo em 1973 (BÜHLER, 2016).

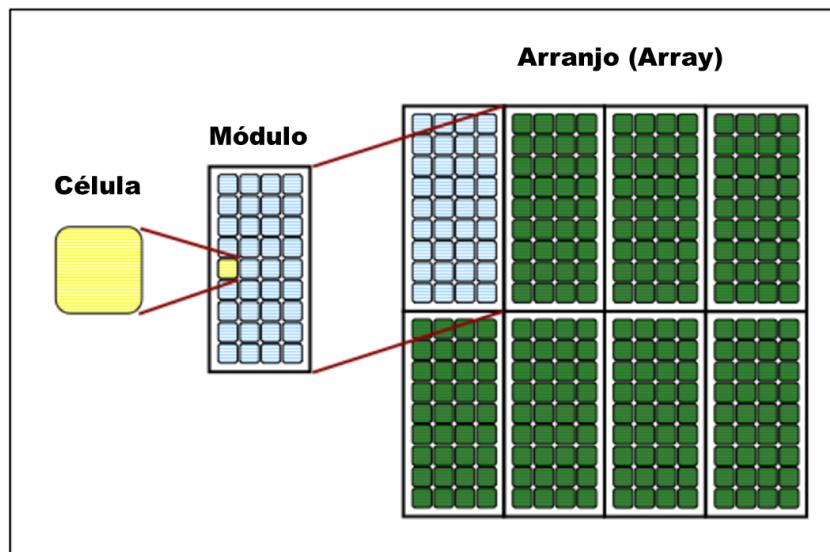
Atualmente, células fotovoltaicas são fabricadas com material semicondutor, ou seja, material com características intermediárias entre condutores e isolantes (NASCIMENTO, 2004). Quando uma superfície semicondutora é iluminada numa certa faixa de frequência, elétrons ligados aos átomos são promovidos a elétrons livres, capazes de conduzir corrente elétrica (VALADARES; MOREIRA, 1998). O material mais utilizado para a produção de células

fotovoltaicas é o silício. Outros materiais alternativos são utilizados em menor escala, ou ainda não são viáveis na prática, como é o caso do grafeno (PEREIRA, 2019).

2.2 Efeito do Sombreamento em Módulos Fotovoltaicos

Uma *célula* fotovoltaica produz uma quantidade de energia insuficiente para suprir uma residência ou comércio. Um *módulo* fotovoltaico nada mais é do que um conjunto de pequenas células fotovoltaicas, cada uma produzindo uma pequena parte da potência do sistema. Como afirma Carneiro (2010), a associação dessas pequenas células é feita em série (o polo negativo de uma célula é ligado ao polo positivo da célula seguinte), compondo assim um módulo fotovoltaico. Neste trabalho, as expressões módulo fotovoltaico e placa solar, como é popularmente conhecido o módulo fotovoltaico, terão o mesmo significado. Já a ligação entre os módulos fotovoltaicos pode ocorrer em série, em paralelo, ou de forma mista, balanceando-se as vantagens e desvantagens das duas anteriores. O agrupamento de vários módulos recebe várias denominações, mas para os efeitos deste trabalho será chamado de *arranjo* (do inglês *array*), como mostrado na Figura 1.

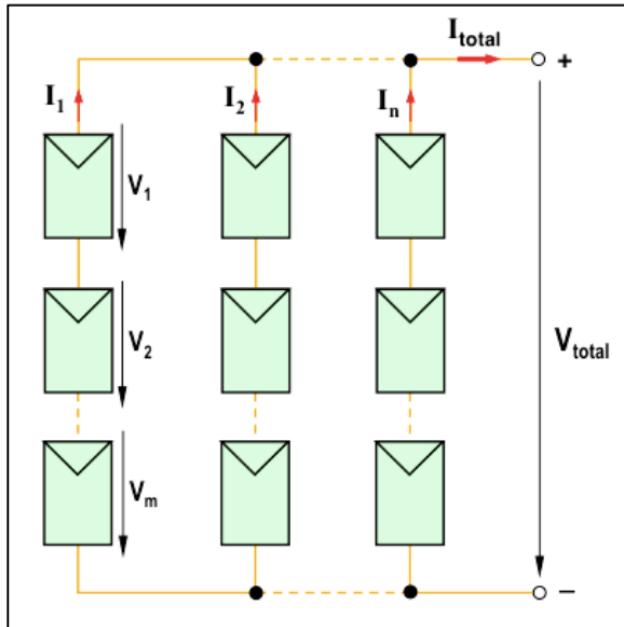
Figura 1 – Hierarquização do agrupamento de células fotovoltaicas em módulos e arranjos.



Fonte: Modificado de Carneiro (2010).

A maneira mais comum de se conectar os arranjos é de forma mista, fazendo-se com que todos os módulos em uma fileira, também chamada de *string*, sejam ligados em série, e as fileiras sejam ligadas entre si em paralelo, como mostra a Figura 2.

Figura 2 – Ligação mista de módulos em um arranjo.



Fonte: Carneiro (2010).

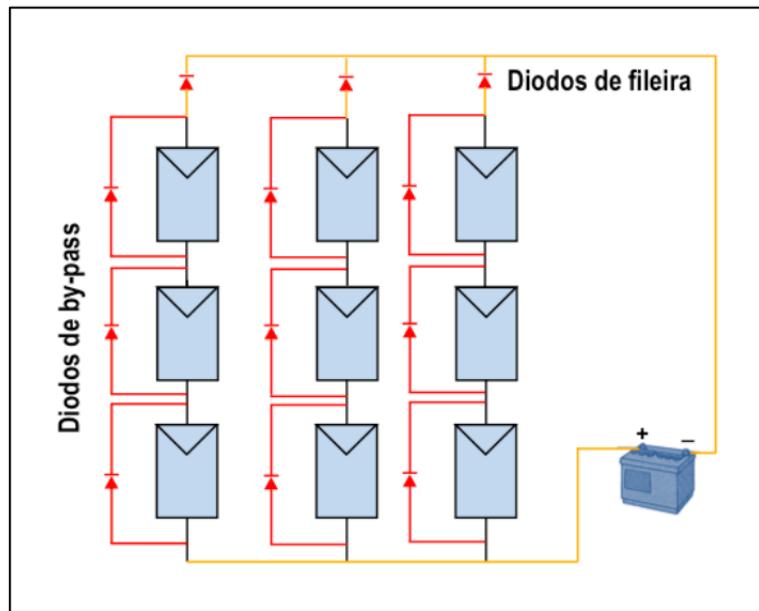
Um dos problemas dessa forma de arranjo é que se há interferência em algum dos módulos de uma fileira, a produção de toda a fileira é prejudicada, podendo até cessar, diminuindo a energia efetivamente gerada. Como afirma Chaudhary e Chaturvedi (2017), quando uma sombra cobre um módulo, ou algum resíduo fica aderido à superfície, as células afetadas (sombreadas ou sujas) produzem menor corrente de fótons em comparação com aquelas completamente iluminadas, tornando as células afetadas inversamente polarizadas, que passam a agir como resistências, dissipando a energia na forma de calor.

Na situação de sombreamento parcial de um módulo, o aumento da temperatura ocasionado pela conversão de energia elétrica em calor gera “pontos quentes”, do inglês, *hotspots*, que podem, em alguns casos, levar à destruição do módulo (CARNEIRO, 2010). De toda maneira, o sistema como um todo é prejudicado, já que é a célula menos iluminada que vai determinar a corrente e a potência de operação de todas as outras células ligadas em série com ela (TORTELLI, 2016).

Para mitigar estes problemas, podem ser utilizados diodos de fileira (também chamados de diodos de bloqueio) e diodos de *by-pass*. Os primeiros são dispositivos utilizados quando as fileiras são ligadas em paralelo, que garantem a passagem da corrente em apenas um sentido, já que obstruções podem ocasionar polarização inversa em uma das fileiras. Já os diodos de *by-pass* são instalados por conjunto de células dentro de um módulo (o dividindo em

sub-módulos), ou em séries de módulos em um arranjo. Em caso de defeito ou obstrução de alguma célula ou módulo no conjunto, o *by-pass* atua desviando a corrente e a fazendo “pular” o referido conjunto, desativando apenas ele. (CARNEIRO, 2010). A Figura 3 contém um esquema da utilização desses diodos.

Figura 3 – Esquema de ligação dos diodos de fileira e de by-pass.



Fonte: Carneiro (2010).

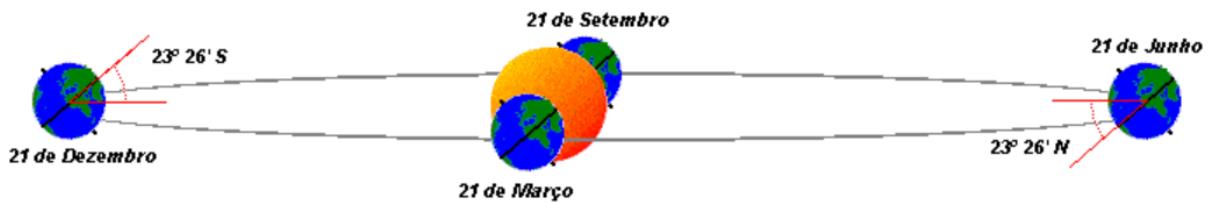
2.3 Trajetória Solar

Como afirmam Collischonn e Dornelles (2013), “O ângulo entre os raios solares e a superfície da Terra em um ponto qualquer depende da época do ano, da hora do dia e da latitude do ponto”. A rotação da terra inflige um ciclo de dia e noite, e a inclinação do eixo de rotação da terra é um importante fator na posição relativa do sol ao longo do dia. Ao ângulo que a linha do equador faz com o sol é dado o nome de declinação solar. Como é possível observar na Figura 4, a declinação solar é igual a zero nos equinócios (dias que marcam o início da primavera ou do outono), tem o seu valor máximo no solstício de verão (dia que determina o início do verão) e tem o seu valor mínimo no solstício de inverno (dia que determina o início do inverno) (BOURGES, 1985).

As datas na Figura 4, referentes aos equinócios e solstícios, serão de grande relevância ao longo deste trabalho. Como será explorado na Seção 2.7, a literatura acerca deste tema reitera a observação da trajetória solar nestas datas, já que são pontos extremos da trajetória

solar, e portanto, serão também os dias onde as extremidades da “mancha” de sombreamento ocorrerão. Em suma, a observação da trajetória e do sombreamento do ambiente nestas datas é representativa para todo o ano.

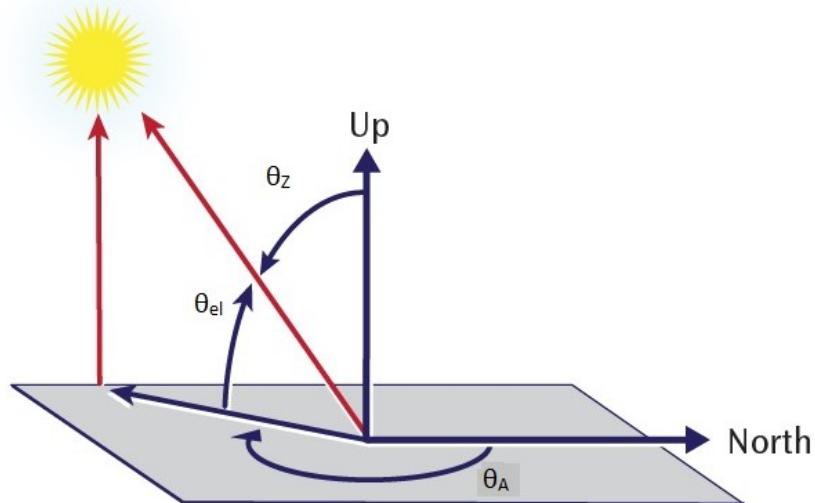
Figura 4 – Declinação solar ao longo das estações do ano.



Fonte: Associação Nacional de Cruzeiros (2022).

Para determinar a posição solar relativa ao plano de uma localidade na Terra, geralmente são utilizados os valores de azimute e elevação solar (Figura 5). O azimute solar θ_A é o ângulo que o sol faz com o norte, no sentido horário (do norte para o leste). Já a elevação solar θ_{el} é o ângulo que o sol faz com o plano do horizonte. Outro ângulo muito utilizado é o chamado zênite solar θ_z , que é o ângulo que o sol faz com a normal da superfície do local. Elevação e zênite são ângulos complementares.

Figura 5 – Representação do azimute θ_A , da elevação solar θ_{el} e do zênite solar θ_z .



θ_{el} = elevation angle,
measured up from
horizon

θ_z = zenith angle,
measured from
vertical

θ_A = azimuth angle,
measured from
North

Fonte: PVPerformance (2022).

A obtenção dos valores de azimute e elevação pode se dar de várias maneiras. São várias as equações e os algoritmos para a determinação destes valores, como por exemplo os

comentados no trabalho de Soulayman (2018). Hoje existem muitos serviços online e gratuitos que se utilizam de fórmulas e algoritmos para estabelecer os valores de azimute e elevação para qualquer dia do ano, em qualquer lugar do planeta terra, como por exemplo o *SunEarthTools*, que será utilizado mais adiante no trabalho.

Um dos principais problemas da utilização de valores de azimute em *softwares* é a diferença em relação aos valores obtidos em um círculo trigonométrico. Enquanto em um círculo trigonométrico o valor de 0° se situa no início do primeiro quadrante, e os valores aumentam no sentido anti-horário de rotação, no sistema de azimute o valor de 0° se situa onde seria o ângulo de 90° no círculo trigonométrico, no início do segundo quadrante, e os valores aumentam no sentido horário, como é mostrado na Figura 6.

Uma maneira de se resolver este problema para cálculos subsequentes que necessitam dos valores desses ângulos no círculo trigonométrico é convertê-los previamente. A equação para a conversão é dada por:

$$\theta_{\text{Atrig}} = 90^\circ - \theta_A, \quad (2.1)$$

onde θ_A indica o ângulo no sistema de azimute e θ_{Atrig} indica o ângulo no círculo trigonométrico, ambos em graus.

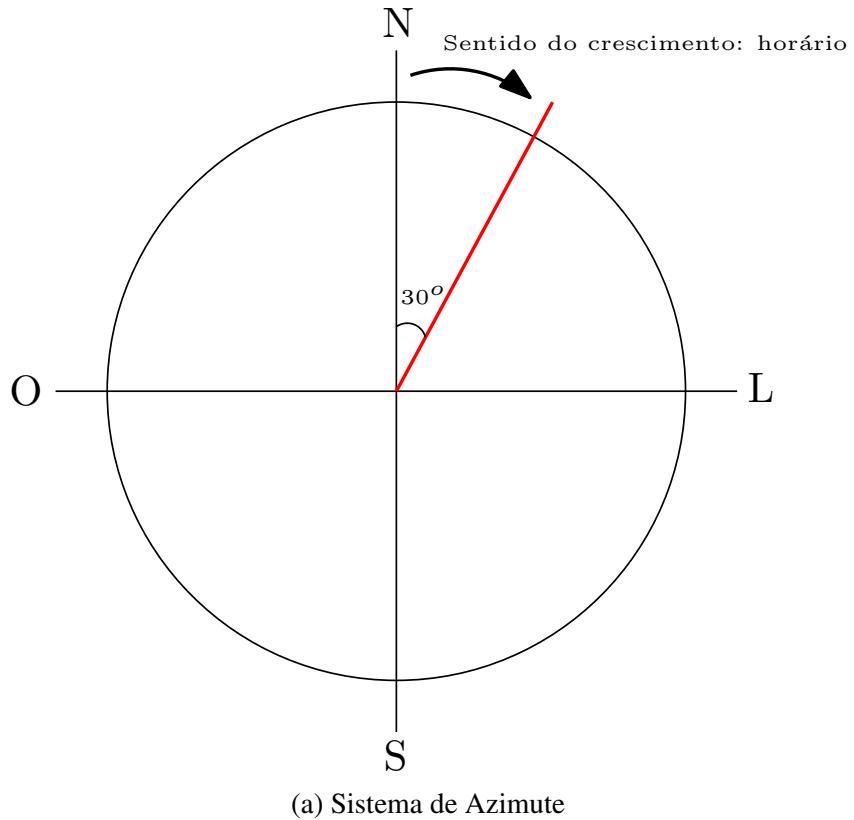
Geralmente, quando se produz uma planta ou um modelo tridimensional de uma edificação, a tendência é que o traçado seja ortogonal ao plano de trabalho e não perpendicular ao sentido do norte, ou seja, o eixo y da planta não coincide com o eixo norte-sul da realidade (Figura 7). Nesse caso, para que a direção do sol esteja coerente com a posição do modelo, mantendo-o no plano de trabalho, é necessário que o azimute seja corrigido por este desvio do norte, denominado na Figura 8 por $\theta_{A_{\text{off}}}$.

Neste caso, a Equação 2.1 passa a ter um termo a mais, como mostrado a seguir:

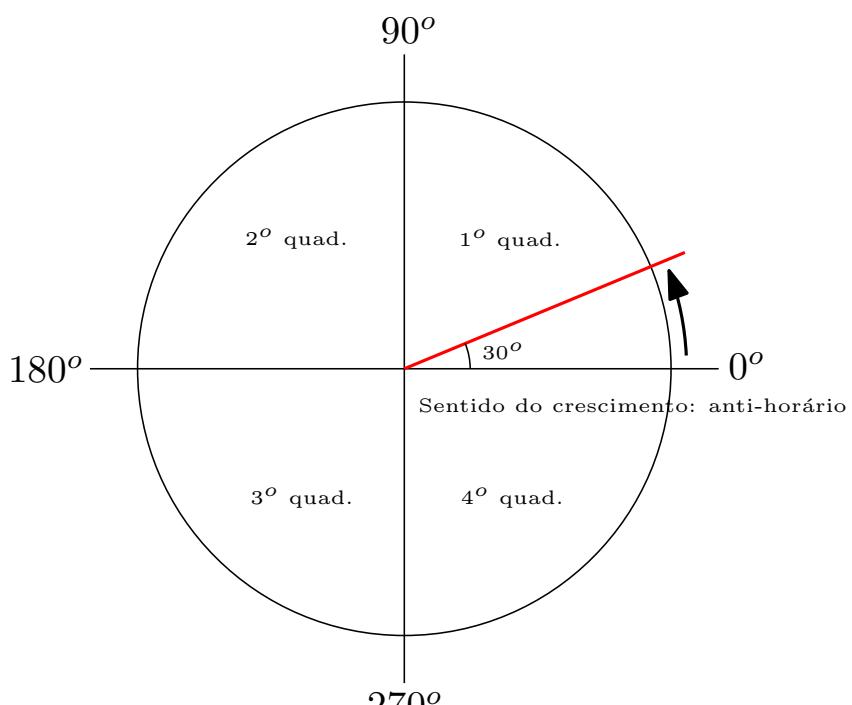
$$\theta_{\text{Atrig}} = 90^\circ - \theta_A - \theta_{A_{\text{off}}}, \quad (2.2)$$

onde θ_A indica o ângulo no sistema de azimute, $\theta_{A_{\text{off}}}$ é o ângulo de desvio do norte em relação ao plano de trabalho e θ_{Atrig} indica o ângulo no círculo trigonométrico, todos em graus. Ambos θ_A e $\theta_{A_{\text{off}}}$ são tomados em relação ao norte, aumentando no sentido horário, como já foi discutido anteriormente.

Figura 6 – Diferenças entre o sistema de azimute e o círculo trigonométrico.



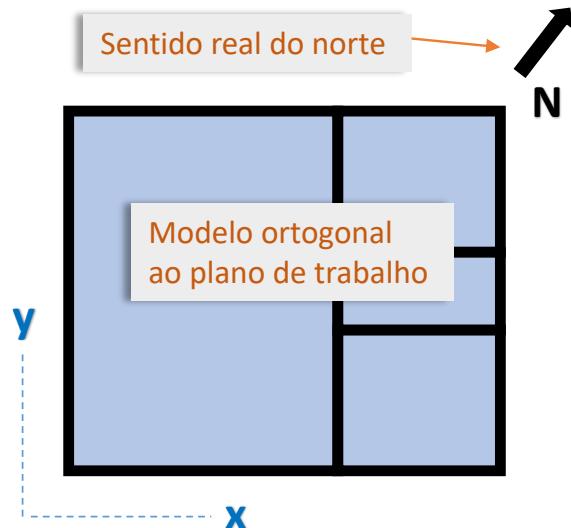
(a) Sistema de Azimute



(b) Círculo Trigonométrico

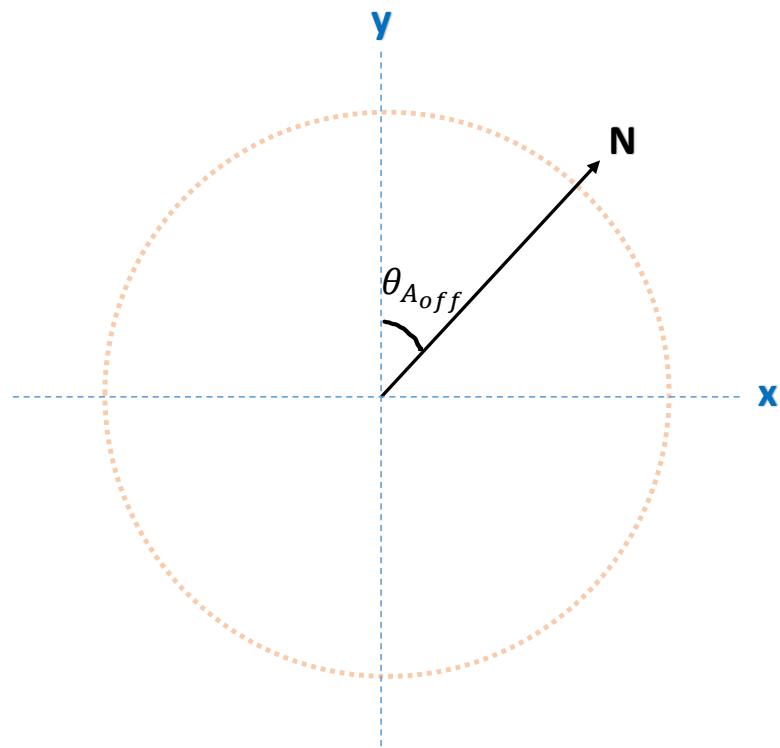
Fonte: Autoral.

Figura 7 – Norte com orientação diferente ao plano de trabalho.



Fonte: Autoral.

Figura 8 – Ângulo de desvio do norte em relação ao plano de trabalho $\theta_{A_{off}}$.



Fonte: Autoral.

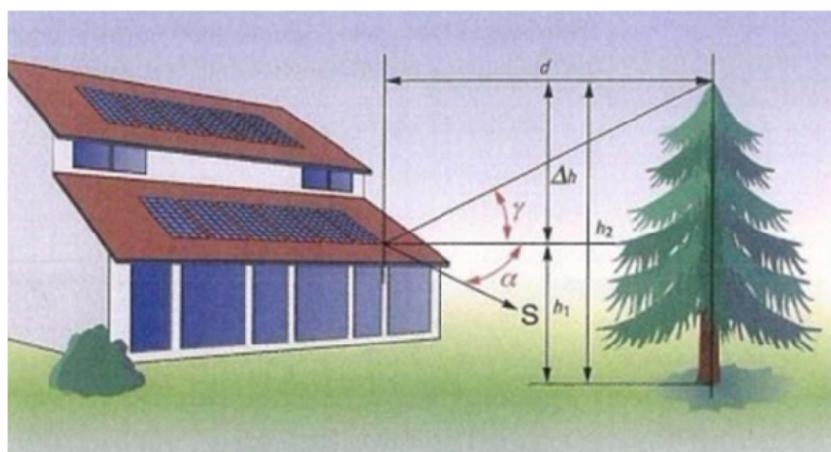
2.4 Determinação do Sombreamento do Ambiente

Para que se evite a interferência de sombras nos módulos fotovoltaicos, é importante analisar os possíveis obstáculos que a luz pode encontrar, seja na própria edificação ou em edificações próximas. Além disso, a posição do sol em relação à Terra é determinante na definição da área sombreada. Para Seratiuk (2016), podem ser usadas as seguintes metodologias para determinar o sombreamento no local de interesse.

2.4.1 Análise do local e dos obstáculos

Um método simples para determinação do sombreamento exige apenas a observação do ambiente como meio para apontar estruturas que possam causar sombras na região onde se pretende instalar as placas solares. As sombras são então determinadas a partir de proporções trigonométricas, considerando a altura da estrutura e sua distância para o local de instalação das placas solares, determinando assim o azimute e elevação solares e o sombreamento incidente, como pode ser visto na Figura 9.

Figura 9 – Determinação dos ângulos de azimute e de elevação de um elemento gerador de sombreamento incidente em um sistema fotovoltaico.



Fonte: Deutsche Gesellschaft für Sonnenenergie (2008).

2.4.2 Análise da trajetória solar usando folha de acetato

Um experimento simples para determinar sombras consiste em utilizar uma folha de acetato com marcações de elevação e azimute. Esta folha transparente é arranjada em um semicírculo, de forma que o observador pode visualizar um objeto e anotar a elevação e o azimute

na localização do mesmo (Deutsche Gesellschaft für Sonnenenergie, 2008). A vizualização pode ser aprimorada com o uso de uma lente angular, semelhante à de um “olho mágico”. Sua utilização está ilustrada na Figura 10. Quando o obstáculo é a copa de uma árvore, são utilizados fatores de transmissão, para determinar a quantidade de luz que atravessa a folhagem, a depender do tipo de árvore (Deutsche Gesellschaft für Sonnenenergie, 2008).

Atualmente, existem alternativas mais modernas ao método da folha de acetato, como aparelhos que se utilizam de GPS para determinar a trajetória solar (SERATIUK, 2016), como também mostra a Figura 10.

Figura 10 – Demonstração do uso da folha de acetato e do dispositivo “SunEye”, que mostra automaticamente a trajetória solar.



Fonte: Mertens (2014).

2.4.3 Equipamentos e softwares para análise de sombreamento

Além dos métodos tradicionais descritos anteriormente, há muitos outros que empregam softwares para determinar a ocorrência e localização de sombras. Basicamente, eles diferem quanto aos dados utilizados para analisar o ambiente: uns manipulam fotografias, enquanto que outros manipulam modelos geométricos tridimensionais que recriam a área de interesse.

Como exemplo do primeiro caso, temos um equipamento chamado *solar pathfinder* (descobridor do caminho do sol, em tradução livre), produzido por uma empresa homônima. O equipamento é uma redoma convexa que permite a visualização dos elementos geradores de sombra projetados em sua superfície (SERATIUK, 2016). Fotos captadas durante a utilização

deste equipamento podem ser usadas por softwares de processamento de imagem. O kit do *solar pathfinder* é mostrado na Figura 11.

Figura 11 – Solar Pathfinder



Fonte: Solar Pathfinder (2022).

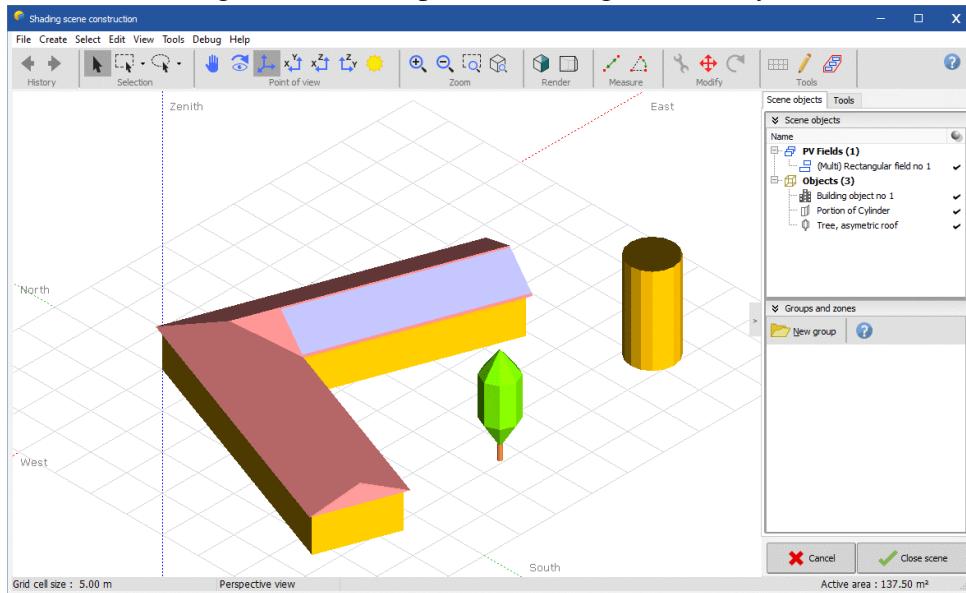
Outro equipamento é o *Horicatcher*, da *Meteonorm*, que capta fotos e permite a análise em um software proprietário, também chamado *Meteonorm*, ou de outras empresas (HORICATCHER, 2022).

Já softwares como o *PVsyst* (Figura 12) permitem a modelagem de um cenário, com formas bastante simplificadas, propondo ser mais amigável ao usuário, ou a importação de modelos tridimensionais de terceiros (*Sketchup*, *Helios3D*, *AutoCAD*, entre outros). Outras funcionalidades incluem o ponto de vista da cena em relação ao sol e o cálculo do sombreamento usando processamento paralelo. O *PVsyst* é um pacote robusto, voltado para a simulação da implantação, eficiência e longevidade do sistema. também inclui a versão 8.0 do *Meteonorm* (PVSYST, 2022).

Um software que permite uma análise mais completa de sombras utilizando modelos altamente complexos é o *ECOTECT Analysis*, da Autodesk (Figura 13). Este software foi descontinuado pela Autodesk (AUTODESK, 2016) e sua tecnologia passou a integrar outros softwares da empresa.

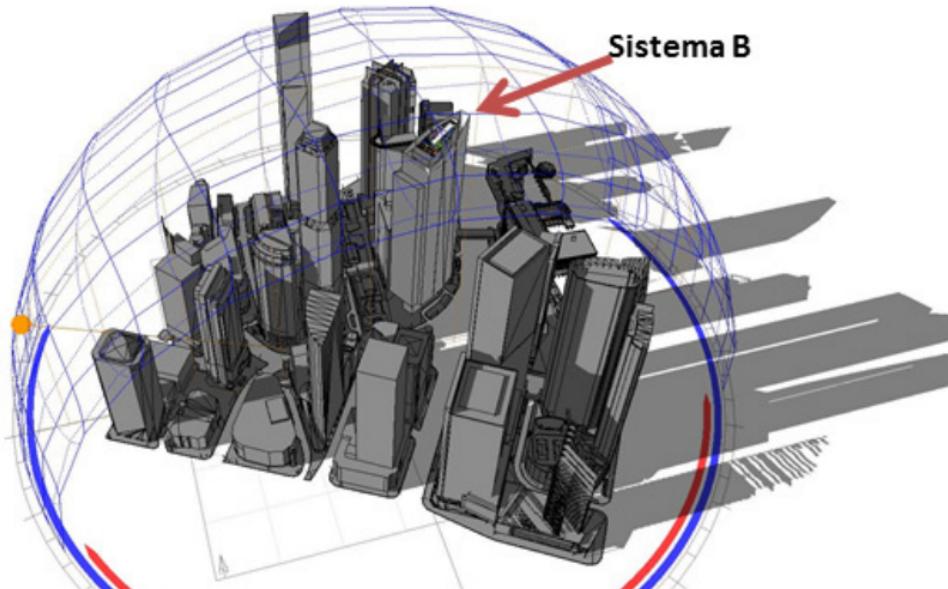
Muitos softwares voltados para modelagem arquitetônica, como é o caso do *Sketchup* e do *Autodesk Revit*, possuem ferramentas que permitem o estudo da trajetória solar e do sombreamento da edificação, mas de forma mais simplificada, visto que não são o principal atrativo destes programas.

Figura 12 – Exemplo de modelagem no PVsyst.



Fonte: PVsyst (2022).

Figura 13 – Reprodução de sombras em centro comercial de Cingapura a partir do *ECOTECT Analysis*.



Fonte: Zomer (2014).

2.5 Síntese de imagens digitais

A representação de uma cena real no computador envolve diversos fatores que impactam na qualidade final da imagem gerada. A reprodução de objetos situados em uma cena, seja ela real ou virtual, introduz uma limitação básica: a frequência de amostragem utilizada. Matematicamente, levando-se em consideração a escala de percepção visual humana, tanto as medidas dos objetos quanto o espectro visível da luz apresentam um *continuum* de

valores. Nesse caso, uma cena pode ser infinitamente subdividida, resultando em um conjunto infinito de pontos, nos quais os valores de cor são amostrados de modo exato. No entanto, devido a limitações da tecnologia de sensores, monitores, impressoras e da própria memória utilizada nesses equipamentos, as imagens representadas digitalmente passam por um processo de amostragem conhecido como *quantização* ou *discretização*, o qual limita a quantidade de subdivisões da cena e a gama de cores disponíveis.

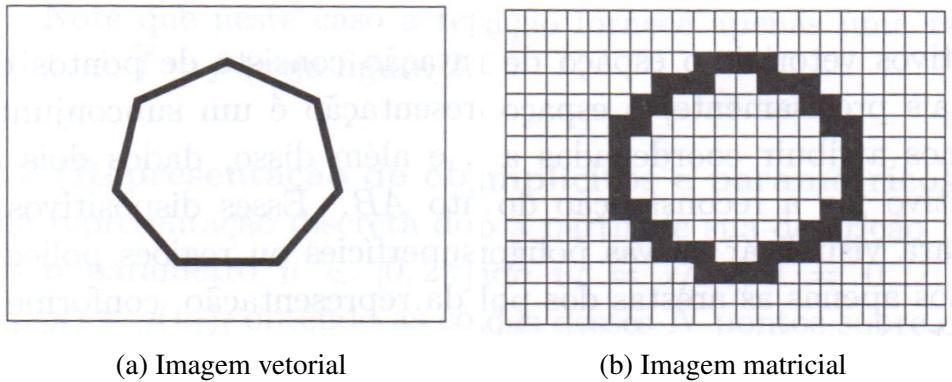
Com relação à representação do espectro visível, o sistema de cor mais difundido é o sistema padrão CIE-RGB. Ele se baseia no modelo tricromático de percepção de cor introduzido pelos físicos Thomas Young e Hermann von Helmholtz (GOMES; VELHO, 2015), os quais observaram que as células fotossensíveis do olho humano separam o espectro visível em baixas frequências (vermelho), médias frequências (verde) e altas frequências (azul), daí o acrônimo "RGB": *red*, *green*, e *blue*. Outros modelos bastante difundidos são o HSV (*hue*, *saturation* e *value*, em português, matiz, saturação e valor) e o modelo CMYK (*cian*, *magenta*, *yellow* e *black key*, em português, ciano, magenta, amarelo e preto), este último muito utilizado em equipamentos de impressão. A metodologia empregada pelo presente trabalho, como será visto mais adiante, se utilizará de uma variação do sistema RGB, o sistema RGBA, um acrônimo para *red*, *green*, *blue* e *alpha*. O valor *alpha*, também chamado de canal *alpha*, representa a quantidade de transparência presente em um ponto da imagem.

Uma cor no sistema CIE-RGBA pode ser vista como um vetor do \mathbb{R}^4 , onde as componentes dos vetores são associadas aos canais R, G, B e A, nessa ordem. Sendo assim, a quantidade total de cores presentes em uma determinada implementação desse sistema será igual ao produto das quantidades de intensidades possíveis de cada componente. No padrão *true color*, por exemplo, são utilizados 8 bits para cada canal, resultando em um total de $2^{24} = 16.777.216$ variações de cores, desconsiderando as intensidades de transparência. A quantidade total de cores permitidas em uma imagem é conhecida como sua *resolução de cor*.

Existem basicamente duas maneiras de se representar e armazenar objetos gráficos no computador: de forma matricial ou de forma vetorial (comumente chamada de vetorizada). A Figura 14 traz uma representação das duas formas. Imagens matriciais ou *bitmaps* são formadas por grades de *pixels*, uma junção dos termos *picture* e *element* (elementos de figura), que são a menor unidade de medida em uma dessas imagens. Cada *pixel* representa uma pequena região da imagem, descrita por suas cores primárias (a depender do sistema de amostragem, explicado anteriormente), e outras propriedades, como transparência. Já imagens vetoriais são

representadas por formas geométricas, linhas e curvas, de modo a conservar as suas medidas e a precisão matemática.

Figura 14 – Imagem na forma vetorial e na forma matricial.



Fonte: Velho e Gomes (2007).

Comumente o termo “imagem digital” diz respeito apenas a imagens matriciais, já que são a forma mais comum de representar imagens do mundo real, captadas por meio de sensores, como acontece nas câmeras digitais e nos digitalizadores (*scanners*). Em uma câmera, por exemplo, o sensor é um dispositivo semicondutor com uma grade de *pixels* sensíveis à luz. A lente da câmera projeta uma imagem no sensor, e é medida a energia luminosa que chega em cada *pixel* dele, resultando em um valor que irá representar a imagem (SHIRLEY; MARSCHNER, 2015). A *resolução espacial* de uma imagem digital é igual à quantidade de pixels nela presentes.

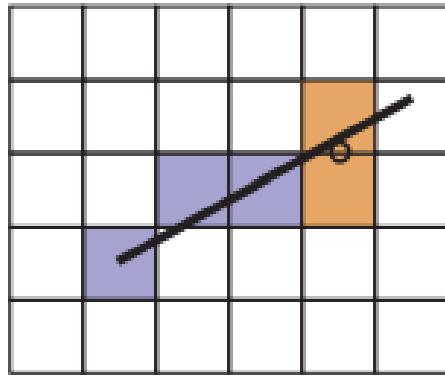
2.5.1 Traçado de Raios

Para representar uma cena digital tridimensional na tela do computador, que é uma superfície bidimensional, são utilizados algoritmos de sintetização de imagens (do inglês, *rendering*), muitas vezes chamados de algoritmos de renderização, que irão discretizar os elementos da cena. As técnicas de renderização podem se basear em rasterização, que é o escaneamento simples dos objetos em uma cena digital, ou em técnicas que simulam os efeitos da ótica, como é o caso do traçado de raios (do inglês, *ray tracing*), que produz imagens mais fotorrealistas e detalhadas do que aquelas rasterizadas (MELO, 2012).

A diferença entre rasterização e traçado de raios que é mais determinante para o presente trabalho é a organização do mapeamento dos objetos da cena. Como afirmam Shirley e Marschner (2015), a renderização envolve a consideração de como cada objeto contribui para cada *pixel*. Os algoritmos de rasterização tradicionais (Figura 15) usam uma renderização

ordenada por objeto (*object-order rendering*), ou seja, o algoritmo analisa cada um dos objetos da cena e indica quais *pixels* se situam no local daquele objeto. Já o algoritmo tradicional de traçado de raios usa uma renderização ordenada pela imagem (*image-order rendering*), ou seja, de forma contrária à rasterização, o algoritmo passa por cada *pixel* e determina qual dos objetos aquele *pixel* vai representar. Utilizando termos comuns em programação, diz-se que na rasterização o “laço” está nos objetos, enquanto que no traçado de raios o “laço” está nos *pixels* da imagem. Mesmo com estas definições, existem algoritmos de traçado de raios baseados em renderização ordenada por objeto, como também algoritmos de rasterização baseados em renderização ordenada pela imagem, mas estas são metodologias não tradicionais.

Figura 15 – Reta convertida em uma imagem matricial por meio de um algoritmo de rasterização.



Fonte: Shirley e Marschner (2015).

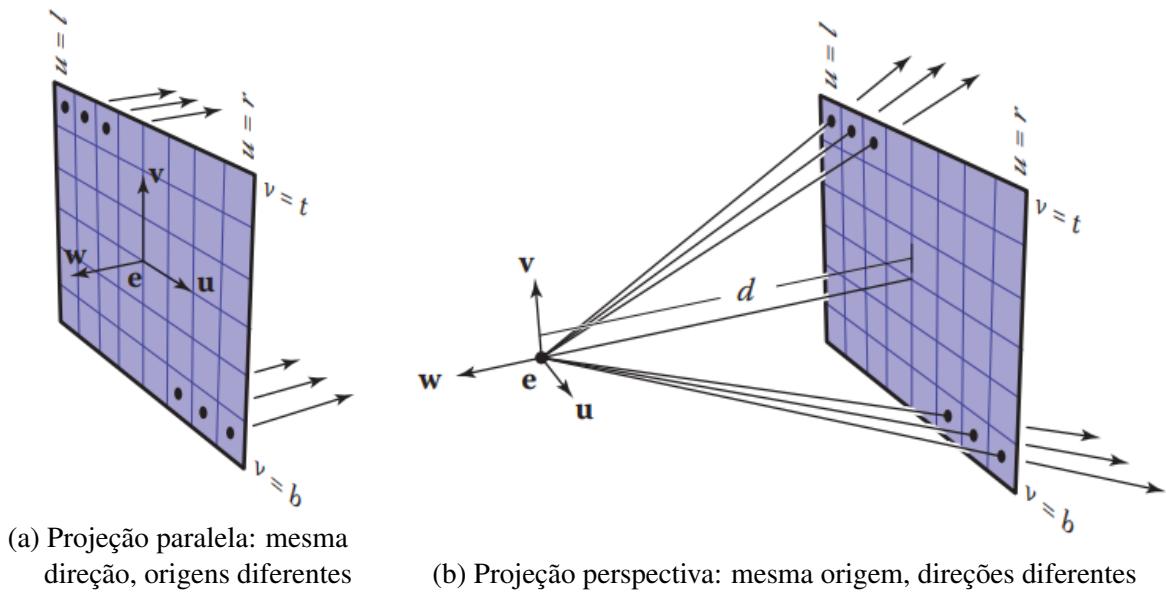
A técnica do traçado de raios consiste em simular a trajetória que os raios de luz percorreriam em uma cena, baseando-se em conceitos da física ótica. Matematicamente, uma cena fica definida por um conjunto de objetos, fontes luminosas e um observador ou câmera. Cada sensor da câmera corresponde a um *pixel* da imagem produzida.

Com o intuito de se evitar a análise da trajetória de raios que não seriam captados pela câmera, que são infinitos, emite-se o raio de luz no sentido contrário ao que ele perfaz na realidade. Isto é, ao invés de emitir um raio partindo da fonte luminosa até atingir a câmera, o raio é traçado saindo de cada sensor/*pixel*, seguindo em direção à fonte luminosa. Desta forma, somente são emitidos raios que contribuirão para a formação da imagem final.

Um ponto importante a se considerar na definição de uma cena é o tipo de projeção utilizada: paralela ou perspectiva. Nesse contexto, a imagem pode ser vista como uma tela de projeção. Na projeção perspectiva, os raios partem de um ponto relativamente próximo à tela. Por este motivo, os raios responsáveis por gerar a imagem não são paralelos entre si, causando

distorções das medidas reais dos objetos projetados, o que permite conferir maior realismo às imagens produzidas. No caso da projeção paralela, todos os raios que passam pelos *pixels* da imagem seguem paralelos entre si. Ela pode ser vista como um caso particular da projeção perspectiva, com a origem dos raios situada no infinito. Por causa disso, as medidas reais dos objetos são preservadas após projetados, o que a torna ideal para desenhos técnicos. A Figura 16 contém ilustrações desses tipos de projeção.

Figura 16 – Diferenças entre a projeção paralela e a projeção perspectiva.



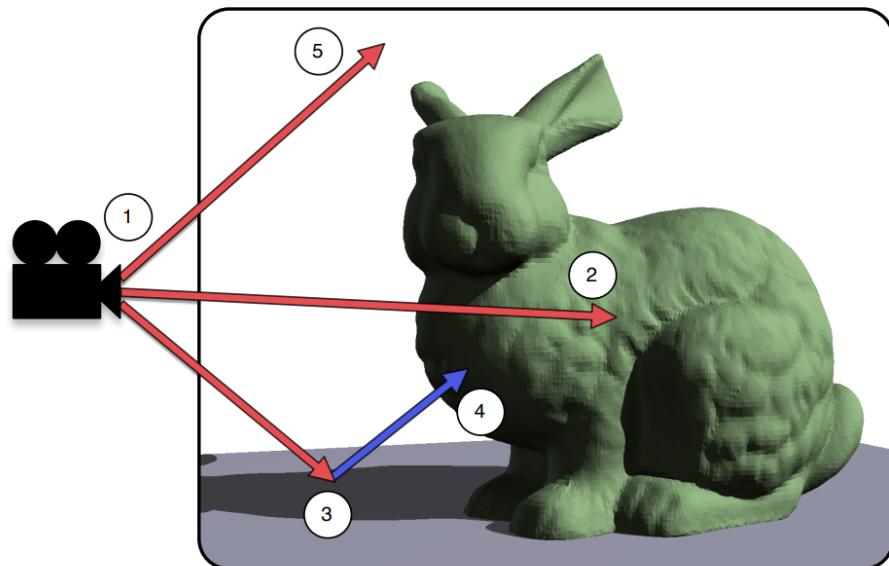
Fonte: Shirley e Marschner (2015).

O algoritmo básico de traçado de raios consiste em percorrer cada *pixel* da imagem e, para cada um deles, a tarefa básica é encontrar o objeto que será interceptado pelo raio que passa por este *pixel* (SHIRLEY; MARSCHNER, 2015). Vários fenômenos da ótica podem ser simulados em um algoritmo de traçado de raios, como sombreamento, difusão, refração e reflexão, aumentando tanto o realismo da imagem quanto o custo computacional total.

A Figura 17 ilustra o ciclo básico de funcionamento de um algoritmo de traçado de raios. O observador, indicado em (1), é o emissor dos raios. O raio em (5) não atinge nenhum objeto, portanto, o pixel correspondente recebe a coloração definida pelo fundo da cena. Já o raio em (3) atinge um objeto e, por isso, é emitido um novo raio (4) saindo do ponto de interseção entre o raio e o objeto e indo em direção à fonte de luz (sentido contrário ao natural). Como o raio emitido atinge outro objeto em (4), isto significa que algum objeto está obstruindo a luz em (3). Portanto, a região em (3) fica sombreada. No caso do raio em (2), ele atinge um objeto e

não possui nenhum bloqueio em direção à luz, logo o objeto é exibido em sua cor original, sem sombreamento.

Figura 17 – Possíveis caminhos dos raios de luz em um algoritmo básico de traçado de raios: (1) parte do observador, (2) atinge um ponto visível, não sombreado, (3) e (4) atinge pontos sombreados, (5) não intercepta objeto.

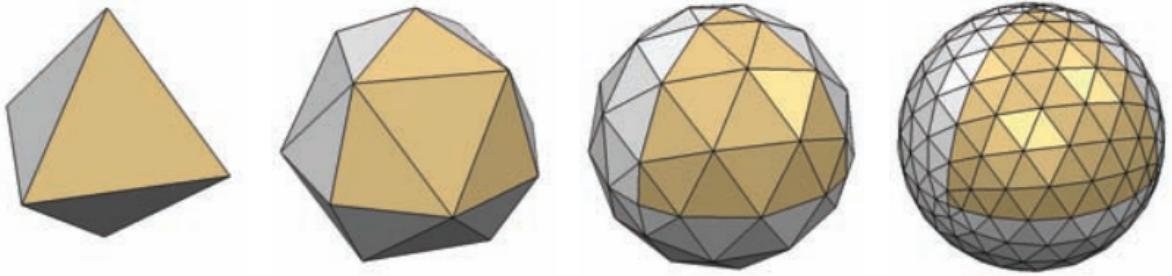


Fonte: Parker *et al.* (2010).

Para que se possa implementar um algoritmo computacional de traçado de raios, é necessária a representação física e matemática dos seguintes aspectos da cena: os objetos que a compõem, as fontes luminosas, os raios emitidos e as interseções (ou colisões) entre raios e objetos.

Existem várias maneiras de se representar um objeto tridimensional no computador. Superfícies “spline” são a representação padrão em sistemas CAD (BOTSCH *et al.*, 2010), porém, malhas de polígonos se tornaram uma alternativa valiosa à tradicional superfície spline, já que sua simplicidade conceitual permite flexibilidade e processamento eficiente (BOTSCH *et al.*, 2010). Vários softwares de modelagem, sejam eles voltados à arquitetura ou à indústria de animação e efeitos especiais, importam, processam e exportam malhas poligonais. As malhas mais comuns são as malhas triangulares (Figura 18) e quadrangulares.

Figura 18 – Esfera representada por uma malha triangular, em quatro níveis de refinamento.



Fonte: Botsch *et al.* (2010).

Um raio de luz tem duas características básicas: sua origem e direção. Como afirmam Shirley e Marschner (2015), uma reta paramétrica tridimensional é ideal para este problema:

$$p(t) = e + td = e + t(r - e), \quad 0 \leq t < \infty, \quad (2.3)$$

onde e é a origem, $d = (r - e)$ é um vetor que caracteriza a direção da luz ou do raio, r é o ponto de destino, t é a distância percorrida pelo raio e p é um ponto na reta, correspondente à distância t (SHIRLEY; MARSCHNER, 2015).

Para verificar a interseção de um raio por um triângulo, pelo método das coordenadas baricêntricas, resolvemos a equação a seguir para t :

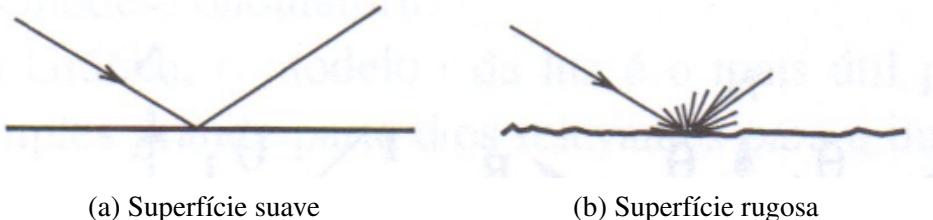
$$e + td = a + \omega(b - a) + \gamma(c - a), \quad (2.4)$$

onde $a, b, c \in \mathbb{R}^3$ são os vértices do triângulo. A interceptação só ocorre se, além de t positivo e finito, ω e γ forem positivos e a soma entre ω e γ for menor que 1 (SHIRLEY; MARSCHNER, 2015).

2.5.2 Colorização de pixels

A definição da cor de um *pixel* na imagem final depende da forma com que consideramos a interação da luz com a superfície do material dos objetos na cena. Velho e Gomes (2007) afirmam que a geometria da superfície pode ser oticamente suave ou rugosa (Figura 19). Em superfícies suaves, a luz se propaga em uma direção de reflexão dominante. Já em superfícies rugosas, a luz é espalhada em várias direções, basando-se em um modelo de microfacetas (microscopicamente, o objeto apresenta imperfeições topográficas).

Figura 19 – Diferença na reflexão de um raio de luz em um metal, considerando uma superfície suave e uma superfície rugosa.



Fonte: Velho e Gomes (2007).

A etapa de colorização ou *shading* se resume em calcular a radiância resultante ao longo de um raio da cena, baseada nas propriedades do material e das fontes de luminosas presentes (AKENINE-MÖLLER *et al.*, 2008). Existem vários modelos de colorização utilizados em computação gráfica, dentre os principais estão o Lambertiano e o de Blinn-Phong. O modelo Lambertiano se baseia na afirmação de que a quantidade de energia de uma fonte de luz que chega em uma superfície depende do ângulo da superfície para com a luz (SHIRLEY; MARSCHNER, 2015). O modelo de Lambert fica descrito pela seguinte equação:

$$L_d = k_d \cdot I_d \cdot \max\{0, \cos(\theta)\}, \quad (2.5)$$

onde L_d indica a cor resultante do *pixel*, k_d é o coeficiente de difusão (uma constante pré-calculada ou dada pelo usuário) ou a cor difusa da superfície, I_d é a intensidade da fonte de luz difusa e θ é o ângulo que o raio de luz faz com a normal da superfície. O uso da função máximo é importante para impedir que um resultado negativo do $\cos(\theta)$ seja computado, o que pode ocorrer devido a eventuais erros de precisão ou a pontos do objeto cuja normal não está “voltada” para a fonte de luz, e portanto deveriam estar obscurecidos. Com isso, superfícies “voltadas” diretamente para a luz recebem iluminação máxima, enquanto superfícies tangentes ou opostas à direção da luz não serão iluminadas (SHIRLEY; MARSCHNER, 2015). No sistema RGB, k_d e I_d são vetores do \mathbb{R}^3 e a multiplicação entre esses vetores se dá componente a componente.

Como os algoritmos de traçado de raios permitem o cálculo dos vetores na direção dos raios de luz e os principais formatos de malhas poligonais permitem associar informações extras a cada uma de suas primitivas, fica mais conveniente o uso da equação de Lambert da seguinte forma:

$$L_d = k_d \cdot I_d \cdot \max\{0, \langle n, \ell \rangle\}, \quad (2.6)$$

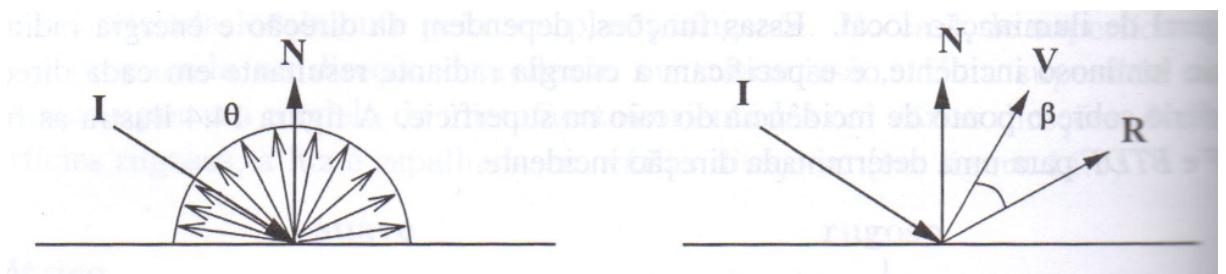
onde n é o vetor normal unitário da superfície e ℓ é o vetor unitário na direção da luz.

Em modelos mais complexos de colorização, a equação de Lambert costuma representar o “termo difuso”, ou seja, a equação de Lambert é responsável por captar a quantia de reflexão difusa (VELHO; GOMES, 2007). Já o “termo especular” vem de um modelo empírico amplamente utilizado para reflexão especular proposto por Phong e atualizado por J. F. Blinn (SHIRLEY; MARSCHNER, 2015). O modelo original de Phong é expresso por:

$$L_s = k_s \cdot I_s \cdot \max\{0, \cos^\rho(\beta)\}, \quad (2.7)$$

onde k_s é o coeficiente reflexão especular da superfície ou a cor especular da superfície, I_s é a intensidade da fonte de luz especular, β é o ângulo entre a direção v de visualização do observador e a direção r de reflexão (Figura 20), e ρ é chamado de expoente de Phong, uma constante que permite a amplificação visual do efeito do brilho especular.

Figura 20 – Representação de um raio luminoso I refletido pela superfície em R , com N sendo a normal da superfície, V a direção de visualização do observador, e β o ângulo entre o observador e o raio refletido.



Fonte: Velho e Gomes (2007).

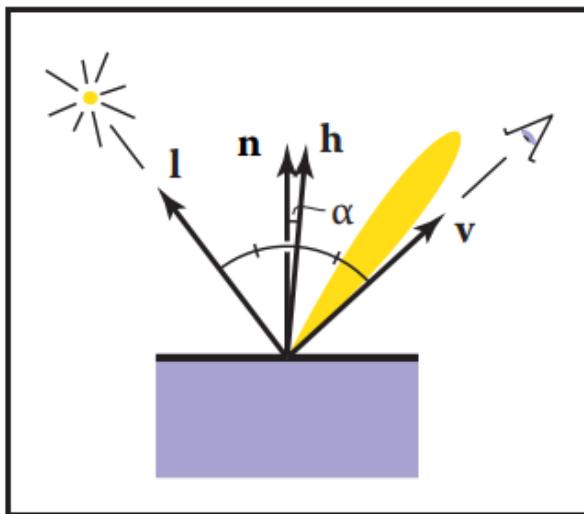
Como é possível perceber a partir da Figura 21, o ângulo entre o raio refletido pela luz e a direção de visualização é proporcional ao ângulo entre a normal da superfície e o vetor h situado na bissetriz entre ℓ e v . Como temos ambos os vetores ℓ e v , podemos obter h a partir da seguinte equação:

$$h = \frac{v + \ell}{\|v + \ell\|}, \quad (2.8)$$

De modo semelhante ao que foi feito para a Equação 2.6, podemos representar a componente especular de Blinn-Phong sob a forma:

$$L_s = k_s \cdot I_s \cdot (\max\{0, \langle n, h \rangle\})^\rho \quad (2.9)$$

Figura 21 – Representação de um raio luminoso l refletido pela superfície na direção da região amarela, com v sendo a direção de visualização do observador, n sendo a normal da superfície, h o vetor na bisetriz de l e v , e α o ângulo entre n e h .



Fonte: Shirley e Marschner (2015).

As equações estabelecidas até aqui se referem à iluminação direta dos objetos em uma cena, ou seja, a cor dos objetos depende apenas do primeiro raio de luz de cada fonte luminosa que colide com ele. No mundo real, porém, observa-se que a partir do momento que um raio entra em um ambiente, ele passa por sucessivas colisões, e consequentes reflexões, tornando todo o ambiente mais iluminado. Ou seja, objetos na cena podem ser iluminados tanto diretamente (devido à primeira colisão com os raios de luz) ou indiretamente (devido a colisões com raios de luz refletidos a partir de outras superfícies). A luz ambiente é o modelo mais simples de iluminação indireta, onde a radiação indireta não varia com a direção e tem um valor constante L_a (AKENINE-MÖLLER *et al.*, 2008). Cenas que não consideram a iluminação ambiente acabam por obscurecer por completo objetos não tocados diretamente pela luz, tornando a imagem menos realista. A presença de um termo de iluminação ambiente L_a indica uma “cor mínima” que o objeto teria na cena. Com isso, a equação de colorização de Blinn-Phong assume a forma final:

$$L_{\text{final}} = L_a + L_d + L_s. \quad (2.10)$$

2.6 Algoritmos e heurísticas de otimização

Com o intuito de definir o melhor arranjo possível para as placas solares em uma região limitada, considerando o efeito do sombreamento, este trabalho utilizará heurísticas de otimização. De acordo com Holtz (2005), *otimização* é o ato de se maximizar ou minimizar

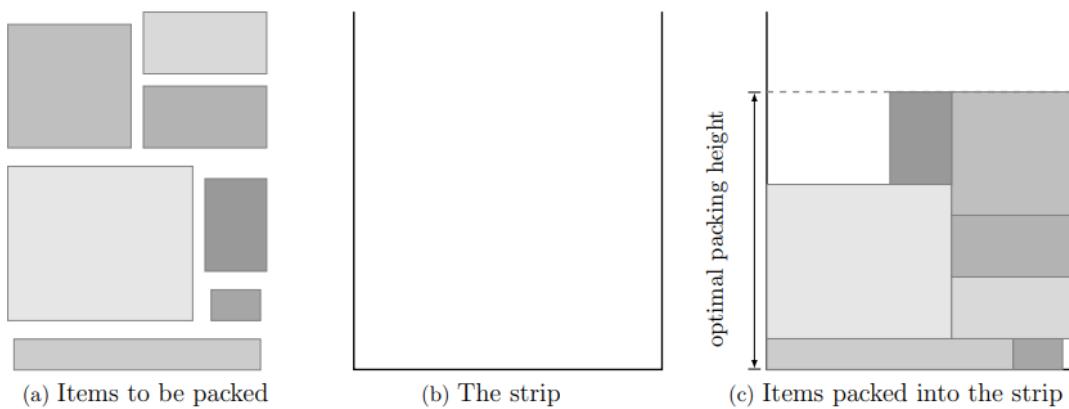
uma função que rege um determinado problema, geralmente balizando-se em restrições nas variáveis da função, denominada *função objetivo*. Métodos para maximizar ou minimizar uma função objetivo se dividem principalmente em dois tipos: métodos *exatos* e *inexatos* (OLIVEIRA, 2019). Métodos exatos garantem que a solução ótima será encontrada, mas costumam exigir recursos computacionais que inviabilizam a obtenção da solução em tempo hábil. Já métodos inexatos não garantem que o resultado será a solução ótima, mas sim uma solução sub-ótima, geralmente suficiente para aplicação no mundo real. O custo computacional reduzido é uma das características que favorecem métodos inexatos, como é o caso das heurísticas de otimização (JOHANNES, 2020).

Como afirma Oliveira *et al.* (2016), métodos heurísticos não encontram soluções ótimas de forma garantida e não são capazes de reconhecer se uma solução ótima foi encontrada. Portanto, a aplicação de heurísticas somente se torna viável quando são tratados problemas onde a solução pode ser comparada e avaliada a partir da observação humana.

O problema de otimização abordado neste trabalho é o do posicionamento de módulos fotovoltaicos. A alocação de módulos em um telhado ou laje pode apresentar várias restrições. Uma restrição essencial é imposta pela geometria do telhado, já que os módulos devem ser inseridos em sua região interna. Outras considerações necessárias são o tamanho do módulo e potenciais obstáculos dentro da região do telhado, como caixas d'água, antenas, escadas, etc. Embora o sombreamento de parte do telhado não seja um impedimento físico como os anteriores, isto pode levar a perdas na eficiência do arranjo, resultando em soluções sub-ótimas.

Matematicamente, a alocação de módulos fotovoltaicos em um telhado pode ser vista como problemas de corte e empacotamento (DÜR; NOWAK, 2008). A resolução destes problemas, de cunho geométrico, têm como objetivo maximizar o uso da área de interesse de forma a minimizar os espaços vazios. No caso do corte, o intuito é evitar o desperdício. Uma aplicação bastante comum de algoritmos de corte está no planejamento do recorte de peças sobre grandes folhas de metal, vidro, entre outros materiais. Já os problemas de empacotamento têm várias vertentes: pode-se buscar utilizar o mínimo de espaço disponível em um pacote de um determinado tamanho, arranjar os objetos de modo a produzir o menor número de pacotes de um tamanho fixo (conhecido como *bin packing*), empilhar objetos de diferentes tamanhos produzindo uma pilha com a menor altura possível (problema denominado *strip packing*, ilustrado na Figura 22), entre outros.

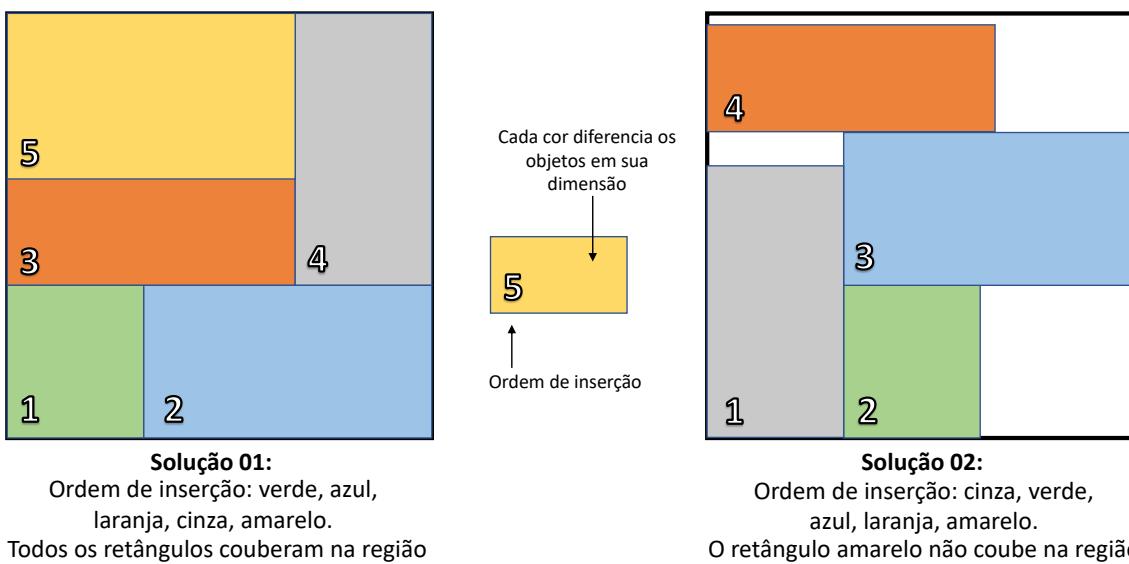
Figura 22 – Exemplificação visual de um problema de *strip packing* bidimensional.



Fonte: Rakotonirainy e Vuuren (2020).

A viabilidade de métodos de otimização, sejam eles exatos ou inexatos, depende de hipóteses simplificadoras. Em problemas de empacotamento, uma possível simplificação é considerar todos os itens retangulares. Um dos principais condicionantes da dificuldade de resolução de um problema de empacotamento é a ordem de inserção dos itens. Como ilustrado na Figura 23, como os objetos possuem dimensões variadas, diferentes ordens de inserção podem levar a soluções distintas.

Figura 23 – Considerações sobre a ordem de inserção em um algoritmo de empacotamento.

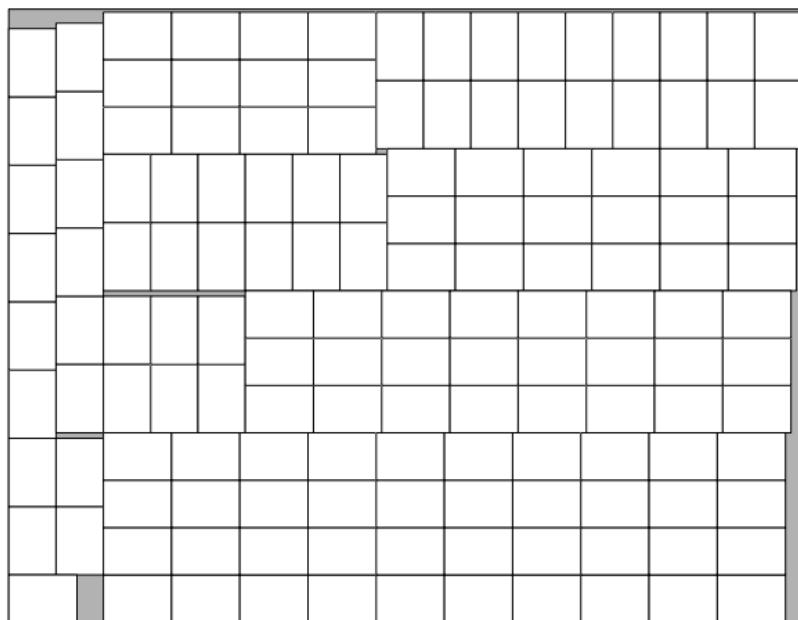


Fonte: Autoral.

Para o problema de inserção de módulos fotovoltaicos em um empreendimento, geralmente, todos os módulos são do mesmo modelo e, portanto, possuem as mesmas dimensões. Como os itens são idênticos, a ordem na qual eles são inseridos é irrelevante (DÜR; NOWAK,

2008), reduzindo o universo de resultados possíveis. Segundo a tipologia de Wäscher *et al.* (2007), este problema se enquadra na categoria IIPP (*Identical Item Packing Problem*, ou problema de empacotamento de itens idênticos). Mais especificamente, se trata de um problema de empacotamento bidimensional de itens retangulares idênticos. Talvez o mais representativo exemplo de um IIPP seja o *Manufacturer's Pallet Loading Packing Problem* (MPLP), ou em tradução livre, Problema de Carregamento de Paletes do Fabricante. Um exemplo prático de um MPLP relacionado à arrumação de pilhas de polpa de celulose em portos brasileiros é apresentado na Figura 24.

Figura 24 – Exemplo prático de um problema MPLP.



Fonte: Birgin *et al.* (2010).

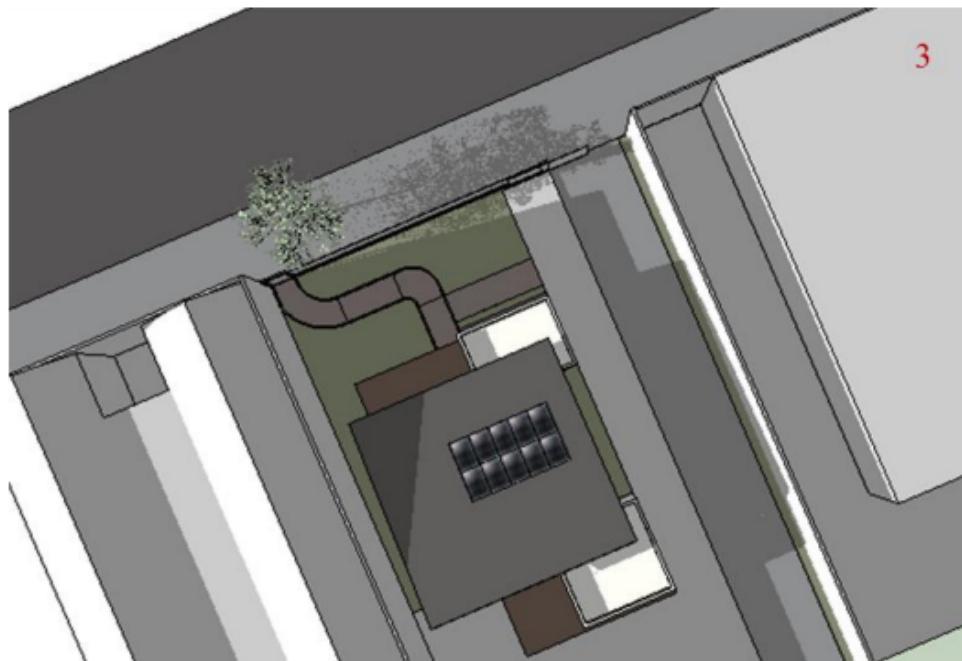
Observando a Figura 24, é possível notar que o algoritmo alterna a orientação dos objetos, colocando alguns na vertical e outros na horizontal, de forma a acomodar o máximo de retângulos. Essa estratégia é permitida neste tipo de aplicação, mas no caso de módulos fotovoltaicos, o uso de apenas uma orientação é mais comum, além da garantia de alinhamento entre os módulos. Mais considerações acerca de hipóteses adotadas neste trabalho serão discutidas na descrição de sua metodologia (Capítulo 3).

2.7 Trabalhos Relacionados

Os principais materiais relacionados ao problema abordado neste trabalho, e que nortearão todo o seu desenvolvimento, foram produzidos por Tortelli (2016), Zomer (2014) e Marinosky *et al.* (2004). Eles analisaram a influência do sombreamento em módulos solares instalados com ou sem estudo prévio da melhor localização. As metodologias utilizadas para delimitação das áreas sombreadas são várias, mas todas empregam softwares com esta finalidade, embora estes softwares não aloquem automaticamente os módulos.

O trabalho de Tortelli (2016) traz um estudo de caso em uma edificação que já possuia um sistema fotovoltaico, o Escritório Verde da Universidade Tecnológica Federal do Paraná. Na posição em que foi instalado, o sistema tinha uma queda de produção ao fim da tarde, devido ao sombreamento promovido por uma edificação vizinha. O trabalho teve como objetivo determinar quantitativamente esta perda, analisando um modelo geométrico no software *SketchUp* quanto à extensão do sombreamento que atingia os módulos (Figura 25). As análises foram realizadas nos dias 21 de cada mês do ano, visando passar pelos equinócios e solstícios, com enfoque apenas nos horários do final da tarde, que eram a origem do problema trazido pelo estudo. O valor de radiação perdida foi obtida através do software RADIASOL, mantido pelo Laboratório de Energia Solar da Universidade Federal do Rio Grande do Sul.

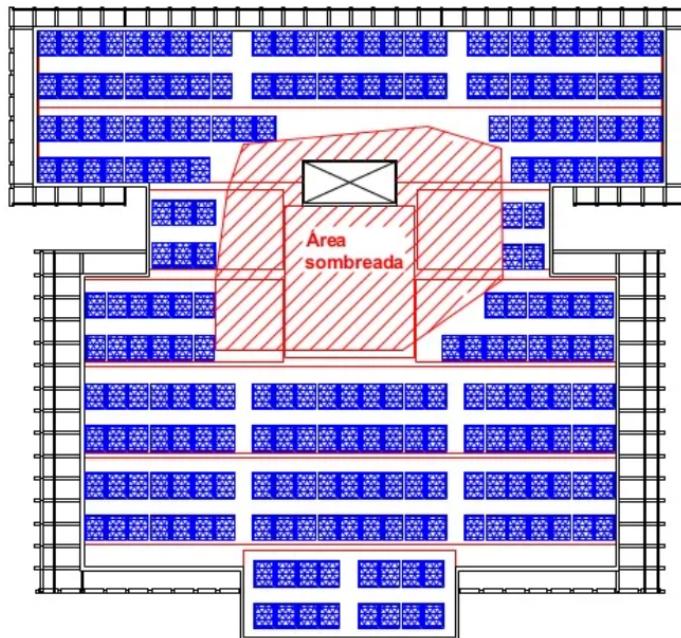
Figura 25 – Modelagem do escritório verde da UTFPR no software *SketchUp*.



Fonte: Tortelli (2016).

O trabalho de Marinosky *et al.* (2004) é um estudo de caso que utiliza o edifício do Conselho Regional de Engenharia, Arquitetura e Agronomia de Santa Catarina como referência para a futura alocação de módulos. Eles analisaram o sombreamento de um reservatório elevado usando um modelo geométrico simplificado construído no software *ECOTECT 5.01* (já descontínuado), considerando quatro dias do ano: dois de equinócio e dois de solstício, com 5 intervalos diários. Apenas foram consideradas sombreadas áreas atingidas por sombras entre nove e quinze horas, visto que este é o período mais importante de produção, devido à intensidade da radiação solar. Os módulos foram posicionados manualmente, somente sobre áreas não sombreadas (Figura 26). Os autores propuseram ainda a instalação de módulos sobre brises localizados na fachada do edifício, caso necessário.

Figura 26 – Resultados de inserção de módulos obtido por Marinosky *et al.* (2004).

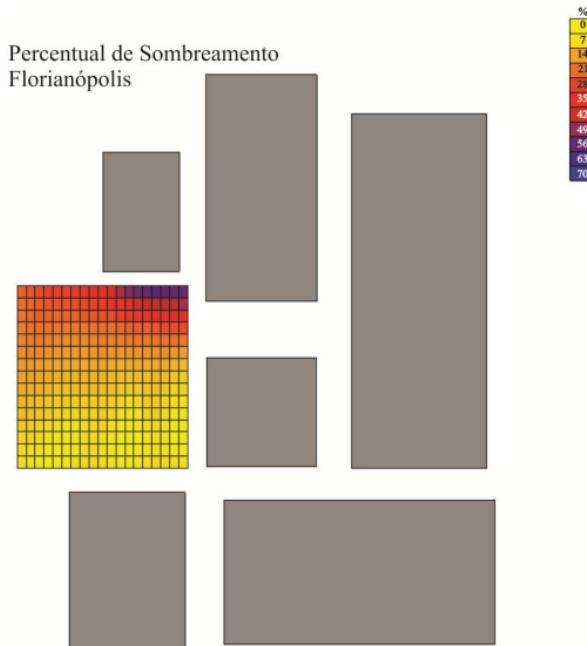


Fonte: Marinosky *et al.* (2004).

A tese de doutorado de Zomer (2014) é, provavelmente, o texto que mais influenciou o presente trabalho. Além de trazer em seu referencial uma extensa coletânia de estudos de caso de sistemas fotovoltaicos de edificações reais, a autora analisa o sombreamento em um caso hipotético: um prédio rodeado por outros prédios, localizado em duas cidades diferentes: Florianópolis e Cingapura. Dados de sombreamento foram obtidos com o auxílio da ferramenta *Shadow Range* do *ECOTECT*, em um dia de cada mês, considerando intervalos fixos de 30 minutos entre 8h e 17h. Como resultado, é produzida uma malha retangular que indica a porcentagem de sombreamento em cada retângulo (Figura 27). Os resultados obtidos com o

ECOTECT foram comparados com outros obtidos por maquetes equipadas com um sensor de radiância. Embora não tenha abordado o problema de posicionamento dos módulos, Zomer (2014) quantificou a energia gerada por sistemas pré-existentes, indo além do que foi trazido por Tortelli (2016).

Figura 27 – Malha de sombreamento em Cingapura, obtida a partir de dados do *ECOTECT*.



Fonte: Zomer (2014).

O problema de alocação de módulos fotovoltaicos pode ser visto como um problema de empacotamento de retângulos idênticos em regiões planas, poligonais. Neste contexto, Dür e Nowak (2008) propuseram uma versão aprimorada de uma heurística chamada *bottom-left*, a qual foi comparada com um modelo de programação linear inteira mista. Elas consideraram telhados planos, representados por polígonos, resolvendo tanto o caso convexo quanto o não-convexo, incluindo furos internos que podem indicar a presença de clarabóias, *shafts* de elevador, entre outras interferências arquitetônicas no telhado. Quatro exemplos foram analisados e, ao final, concluiu-se que os resultados obtidos pela heurística são satisfatórios e requerem um tempo de execução consideravelmente menor, quando comparado ao do modelo de programação linear. Embora tenham citado o problema de sombreamento, a heurística proposta não leva em consideração essa informação, nem a inclinação dos módulos e do telhado.

3 METODOLOGIA

O objetivo principal deste trabalho é implementar um método automático para a alocação de módulos fotovoltaicos em telhados de edificações considerando eventuais regiões de sombreamento. Neste sentido, podem ser definidas duas etapas independentes:

- (i) Construção de um campo escalar que represente quantitativamente o sombreamento de cada ponto pertencente ao telhado ao longo de um dado período;
- (ii) Dado um campo escalar de sombreamento e uma quantidade fixa de módulos fotovoltaicos idênticos, posicionar automaticamente os módulos evitando, sempre que possível, regiões sombreadas.

Todos os códigos desenvolvidos foram escritos na linguagem de programação *Python* e estão disponíveis em <<https://github.com/AMarcosC/TCC-Raytracer>> para download. Os programas foram produzidos utilizando o sistema operacional Ubuntu 20.04.4, rodando nativamente a versão 3.8.10 do *Python*. O código foi escrito no editor de texto *Atom*.

3.1 Construção do campo de sombreamento

O cálculo do campo escalar de sombreamento foi realizado por intermédio da técnica de traçado de raios. Basicamente, são necessários os seguintes dados de entrada:

- um modelo geométrico tridimensional da edificação;
- valores de azimutes e elevações solares, usados para determinar a direção da luz do sol.

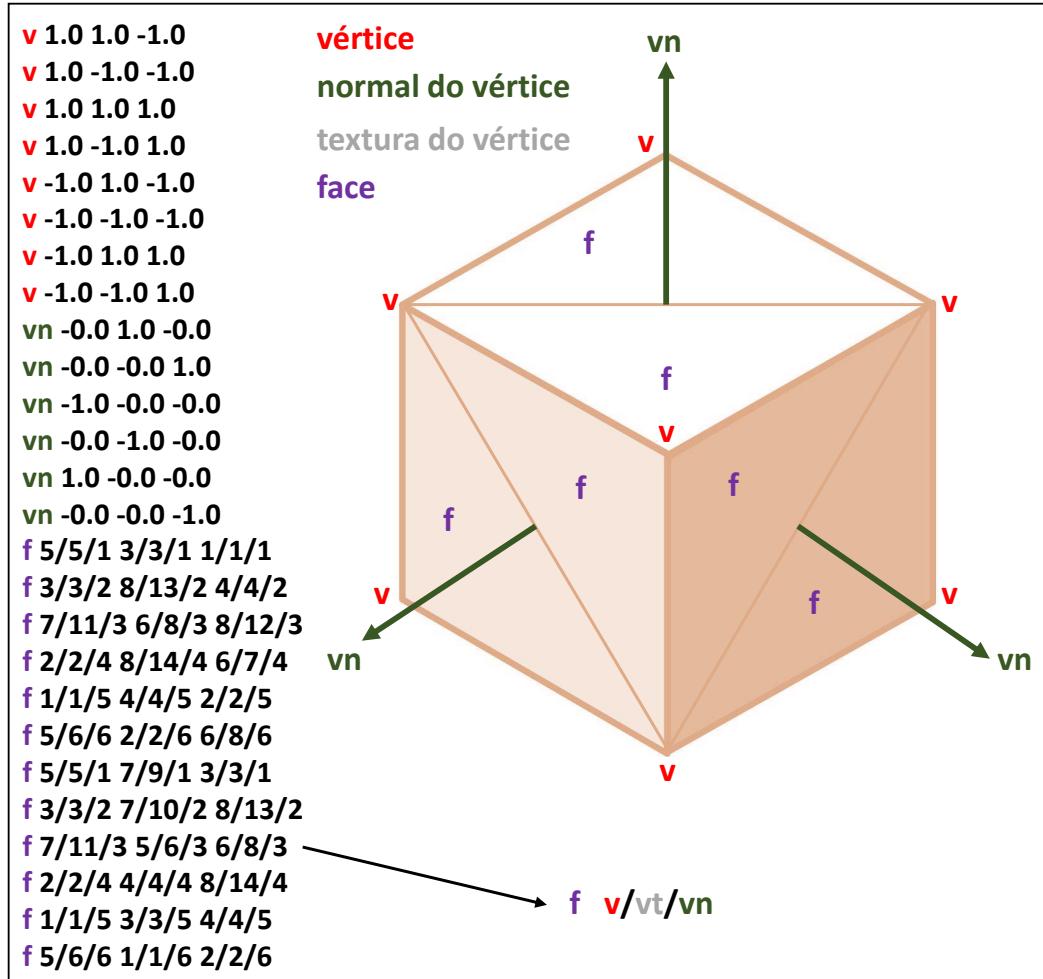
3.1.1 *Modelo geométrico tridimensional*

Por simplicidade, todos os modelos geométricos tratados foram convertidos em malhas triangulares, armazenadas segundo o formato OBJ, desenvolvido pela Wavefront Technologies. Esse tipo de arquivo contém informações sobre os vértices, arestas e faces que compõem uma malha poliédrica.

A estrutura básica de um arquivo OBJ está ilustrada na Figura 28. Nela, está representado um cubo centrado na origem, com comprimento de lado igual a dois. As linhas iniciando em “v” indicam os vértices do objeto, que no caso do cubo, são oito, e os três números que seguem são as coordenadas x , y e z correspondentes. As linhas iniciando em “vn” indicam os vetores normais das faces que compõem a superfície do cubo, também definidos por suas

componentes nas direções x , y e z . Os vetores normais são sempre unitários, o que é conveniente para cálculos de colisão e colorização, comuns à técnica de traçado de raios.

Figura 28 – Estrutura de um arquivo .obj de um cubo.



Fonte: Autoral.

As linhas iniciando em “f” indicam as faces do objeto. Como a superfície do cubo foi triangulada, suas seis faces quadradas dão lugar a doze faces triangulares. Toda linha associada a uma face triangular é caracterizada por três sequências de três números inteiros cada. Os valores em cada sequência são separados por barras e correspondem ao número do vértice, da coordenada de textura (não representado na Figura 28, porém não será importante para o trabalho) e do vetor normal correspondente. Considerando o arquivo da Figura 28 ordenado de cima para baixo, na primeira linha de face, o valor “5/5/1” indica que o primeiro vértice da primeira face está na quinta linha do tipo “v” e que a normal está na primeira linha do tipo “n”.

Embora bastante comum nos principais softwares de modelagem de cunho geral, como o *Blender*, *Cinema4D* e outros, o OBJ não é usual em softwares voltados ao desenho

arquitetônico, como o *Autodesk Revit*. No entanto, é possível obter arquivos OBJ por meio de conversão de um arquivo IFC ou FBX produzido pelo Revit. Neste trabalho, o arquivo FBX foi aberto no Blender, triangulado e exportado como OBJ.

No programa aqui desenvolvido, as malhas triangulares são representadas por uma lista de vértices V , contendo as três coordenadas de cada vértice, e uma lista T de triângulos. Cada triângulo é representado usando uma classe Triângulo. Basicamente, um objeto da classe Triângulo possui três propriedades: os três índices indicando seus vértices em V , sua cor em RGBA (que será utilizada na fase de colorização da imagem) e o seu vetor normal.

3.1.2 Azimutes e elevações solares

A obtenção dos azimutes e elevações solares pode se dar da forma preferida pelo usuário, já que bastam os valores em graus decimais. É importante salientar que o software implementado considera os ângulos como no círculo trigonométrico e, para tanto, os valores de azimute passam pelo tratamento da Equação 2.2. Nos testes realizados, foram utilizados dados extraídos com o auxílio de ferramentas disponíveis no portal SunEarthTools (2022). Com elas, é possível obter valores de azimute e elevação solar a cada hora de um certo dia ou em intervalos menores definidos pelo usuário e salvá-los em um arquivo CSV (do inglês, valores separados por vírgula). Os dados do *SunEarthTools* foram validados por testes em escala real, como apresentado na Seção 3.1.4.

Os valores angulares de azimute e elevação, um sistema de coordenadas polares, precisa ser transformado em um vetor que represente a direção dos raios solares naquele dado momento do dia. As equações para criar um vetor unitário a partir de coordenadas polares são:

$$\begin{aligned}\ell_x &= \cos(\theta_{el}) \cdot \cos(\theta_{Atrig}), \\ \ell_y &= \cos(\theta_{el}) \cdot \sin(\theta_{Atrig}), \\ \ell_z &= \sin(\theta_{el}),\end{aligned}\tag{3.1}$$

onde θ_{el} é a elevação solar, θ_{Atrig} é o azimute solar no círculo trigonométrico e ℓ_x, ℓ_y, ℓ_z são as componentes do vetor unitário da direção da luz.

3.1.3 Funcionamento do traçado de raios

Como mencionado anteriormente, a técnica de traçado de raios é capaz de simular diversos efeitos da ótica. Para o caso particular do problema em estudo, a obtenção de sombras

“duras” é suficiente para atingir o resultado desejado, isto é, basta determinar se um ponto está na sombra ou está iluminado, sem meio termo. No entanto, para dar uma melhor noção de profundidade às imagens produzidas, foi aplicada uma versão ponderada do modelo de iluminação Lambertiano com luz ambiente às áreas não sombreadas. Com efeito, a cor de um ponto não sombreado é dada por:

$$L_{\text{final}} = p_a \cdot L_a + p_d \cdot L_d, \quad 0 < p_a + p_d \leq 1, \quad (3.2)$$

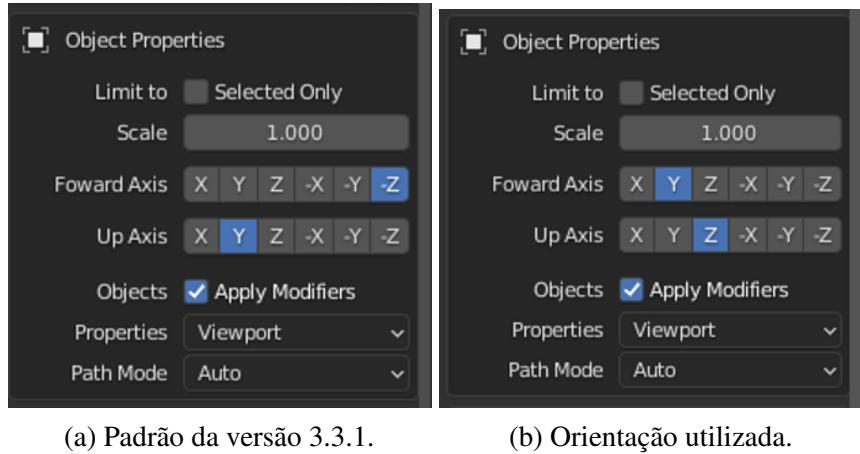
onde p_a e p_d são pesos escolhidos pelo usuário e L_{final} é a cor final do pixel. Valores de p_a e p_d que desrespeitem os limites impostos pela Equação 3.2 podem provocar erros no programa, já que podem conduzir a valores de cor não existentes no sistema RGBA. Foram usados $p_a = 0,4$ e $p_d = 0,6$ por produzirem imagens visualmente superiores, determinados experimentalmente. Por simplicidade, foi assumido que $I_d = I_a = (1, 1, 1, 1)$ e $k_d = k_a = C$. Com isso, a Equação 3.2 se reduz a:

$$L_{\text{final}} = (p_a + p_d \max \{0, \langle n, \ell \rangle\}) \cdot C, \quad (3.3)$$

onde C representa a cor do objeto especificada pelo usuário para cada arquivo OBJ, n o vetor normal à superfície do objeto e ℓ é a direção da luz, obtida a partir da Equação 3.1.

A projeção utilizada na produção das imagens é a paralela, já que se deseja manter as proporções reais do modelo e visualisá-lo em planta. Para simplificar o problema, a tela é plana e se situa no plano x - y , ou seja, os raios que saem dela estão na direção do eixo z , no sentido negativo. Esta é uma consideração importante para a exportação do modelo. No Blender, por exemplo, o padrão é ter o eixo $-z$ para indicar o sentido frente/trás do modelo e y para indicar o sentido cima/baixo. Já o software produzido utiliza y para indicar o sentido frente/trás e z para indicar o sentido cima/baixo. Para contornar isso em versões antigas do Blender, era necessário rotacionar o modelo, mas versões recentes, a partir da 3.2, permitem exportar o modelo OBJ com a configuração correta, como mostra a Figura 29.

Figura 29 – Configuração necessária para exportação do arquivo OBJ no Blender com a orientação correta dos eixos coordenados.



(a) Padrão da versão 3.3.1.

(b) Orientação utilizada.

Fonte: Autoral.

Para determinar o tamanho e a posição da tela de projeção é necessário passar por todos os triângulos da cena para garantir que estejam no interior do volume de visão. Por segurança, adiciona-se às dimensões da tela um “forramento” ϵ maior ou igual do que a maior dimensão do módulo fotovoltaico que se deseja utilizar. Isto é necessário para evitar que o algoritmo implementado tente colocar módulos em posições fora do volume de visão. O procedimento para obtenção dos valores extremos da tela está detalhado no Algoritmo 1. Nele, uma constante τ é utilizada para representar um valor muito grande de distância, quase infinito dada a escala dos objetos na cena, que por padrão foi assumido igual a 10^{40} . Ao final, a tela de projeção fica caracterizada pelo retângulo $[X_{\min}, X_{\max}] \times [Y_{\min}, Y_{\max}]$, deslocado uma unidade acima do maior valor de z existente na malha.

No Algoritmo 1, a variável Z_{\max} obtida é utilizada para garantir que a tela se situe acima do modelo. Com os pontos extremos da tela, será possível determinar o seu tamanho em pixels. O usuário deverá inserir a densidade de pixels, aqui chamada de γ_p , explicitando quantos pixels existirão por unidade de distância utilizada. Um exemplo de como o valor de γ_p influencia na imagem e na precisão do modelo é mostrado na Figura 30: as silhuetas ficam mais definidas à medida em que se aumenta o valor de γ_p . É possível perceber que na Figura 30(a) o modelo não possui sombras na parte central superior, que só são computadas nas Figuras 30(b) e 30(c).

Algoritmo 1: Determinação das coordenadas da tela de projeção.

Dados: lista de vértices V , valor de forramento ε , $\tau = 10^{40}$

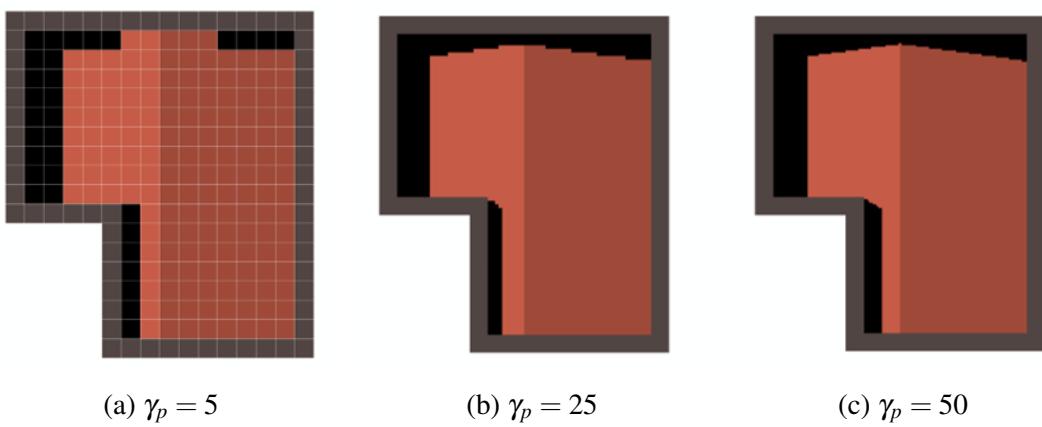
Resultado: Coordenadas extremas da tela

```

1 inicialização;
2  $X_{\min} \leftarrow \tau;$ 
3  $X_{\max} \leftarrow -\tau;$ 
4  $Y_{\min} \leftarrow \tau;$ 
5  $Y_{\max} \leftarrow -\tau;$ 
6  $Z_{\max} \leftarrow -\tau;$ 
7 para cada vértice em  $V$  faça
8   se vértice.x >  $X_{\max}$  então
9     |    $X_{\max} \leftarrow$  vértice.x;
10    se vértice.x <  $X_{\min}$  então
11      |    $X_{\min} \leftarrow$  vértice.x;
12    se vértice.y >  $Y_{\max}$  então
13      |    $Y_{\max} \leftarrow$  vértice.y;
14    se vértice.y <  $Y_{\min}$  então
15      |    $Y_{\min} \leftarrow$  vértice.y;
16    se vértice.z >  $Z_{\max}$  então
17      |    $Z_{\max} \leftarrow$  vértice.z;
18  $X_{\min} \leftarrow X_{\min} - \varepsilon;$ 
19  $X_{\max} \leftarrow X_{\max} + \varepsilon;$ 
20  $Y_{\min} \leftarrow Y_{\min} - \varepsilon;$ 
21  $Y_{\max} \leftarrow Y_{\max} + \varepsilon;$ 
22  $Z_{\max} \leftarrow \max \{1, Z_{\max} + 1\};$ 
23 retorna  $X_{\min}, X_{\max}, Y_{\min}, Y_{\max}, Z_{\max}$ 

```

Figura 30 – Exemplo de imagens de um modelo produzidas com diferentes valores de γ_p .



Fonte: Autoral.

Se a unidade de distância do modelo for o metro, então γ_p é dado em *pixels* por metro. Por exemplo, um γ_p de valor 50 indica que para cada metro haverão 50 *pixels*, e portanto, um *pixel* equivaleria a 2 cm em medidas reais. A quantidade de *pixels* da imagem produzida nos

sentidos horizontal e vertical será, portanto:

$$Q_h = \lceil (X_{\max} - X_{\min}) \gamma_p \rceil, \quad (3.4)$$

$$Q_v = \lceil (Y_{\max} - Y_{\min}) \gamma_p \rceil, \quad (3.5)$$

com X_{\max} , X_{\min} , Y_{\max} e Y_{\min} obtidos do Algoritmo 1, Q_h sendo a quantidade de *pixels* no sentido horizontal, e Q_v a quantidade de *pixels* no sentido vertical. O número total de *pixels* na imagem será o produto entre Q_h e Q_v .

Para a construção do campo de sombreamento, são necessárias ainda as seguintes matrizes, todas com dimensões $Q_v \times Q_h$:

- uma matriz M_{coord} que associa a cada *pixel* o seu ponto na tela, sendo estes pontos a origem dos raios emitidos pelo traçado de raios;
- uma matriz que terá as coordenadas apenas dos pontos onde um raio colide com elementos do telhado, denominada “região viável”, ou *RV*;
- uma matriz que atribui valores para *pixels* sombreados, apenas na “região viável”, chamada de matriz H na Figura 31;
- uma matriz para cada imagem gerada, ou seja, para cada par de azimute e elevação ao longo do dia.

Antes de iniciar a produção de cada imagem, delimita-se a região de soluções viáveis, aqui denominada *região viável*. A região viável é importante para recortar da imagem apenas áreas onde é fisicamente possível instalar módulos fotovoltaicos, informação que será útil também para o algoritmo de alocação dos módulos. Parte principal desta tarefa é criar duas listas de triângulos distintas: uma lista contendo apenas triângulos das regiões em que se pode colocar módulos (telhados, lajes e afins) e outra lista contendo apenas estruturas que podem eventualmente causar o sombreamento da cena, onde não serão inseridos módulos (paredes, reservatórios, platibandas, chaminés, etc.), ou seja, todo o resto do modelo. No Autodesk Revit, essa tarefa é simples, bastando exportar um arquivo FBX apenas com o telhado e outro arquivo com o restante do modelo, convertendo-os para OBJ, como descrito anteriormente. Com isso, são construídas duas malhas triangulares, cada uma com suas listas de vértices e triângulos exclusivas. A obtenção da região viável considera a reunião dessas malhas, como mostrado no Algoritmo 2.

Algoritmo 2: Delimitação da região viável.

Dados: lista de todos os triângulos T da cena; Lista de triângulos T_{rv} da área que se pode inserir o sistema fotovoltaico; matriz M_{coord} dos pontos da tela; $Q_v; Q_h; \tau$

Resultado: Matriz com as coordenadas da região viável

```

1 inicialização;
2  $RV \leftarrow$  matriz vazia de dimensões ( $Q_v, Q_h$ );
3 para  $i \leftarrow 0$  até  $Q_v$  faça
4   para  $j \leftarrow 0$  até  $Q_h$  faça
5      $pos\_atual \leftarrow M_{coord_{i,j}}$ ;
6      $dist\_atual \leftarrow \tau$ ;
7     para cada triângulo  $t$  em  $T$  faça
8       Encontrar distância de interceptação  $d$  a partir da Equação 2.4;
9       se  $d < dist\_atual$  e o triângulo  $t$  estiver contido em  $T_{rv}$  então
10          $dist\_atual \leftarrow d$ ;
11         Encontrar ponto de interceptação  $p$  a partir da Equação 2.3;
12          $RV_{i,j} = p$ ;
13 retorna  $RV$ 

```

Após definida a região viável, aplica-se a técnica de traçado de raios para obter uma imagem da cena, conforme mostrado no Algoritmo 3. A matriz *imagem* é então salva no formato PNG usando o pacote *PIL*.

Algoritmo 3: Produção de imagens a partir do traçado de raios.

Dados: Todos os triângulos T da cena; matriz M_{coord} ; Q_v ; Q_h ; p_a ; p_d ; $\tau = 10^{40}$

Resultado: Matriz com valores de cores para formação da imagem

```

1 inicialização;
2 imagem  $\leftarrow$  matriz vazia de dimensões ( $Q_v$ ,  $Q_h$ );
3 para  $i \leftarrow 0$  até  $Q_v$  faça
4   para  $j \leftarrow 0$  até  $Q_h$  faça
5     pos_atual  $\leftarrow M_{\text{coord}_{i,j}};
6     dist_atual  $\leftarrow \tau;
7     cor_atual  $\leftarrow (0, 0, 0, 0)$  (Transparente em RGBA);
8     para cada triângulo  $t$  em  $T$  faça
9       Encontrar distância de interceptação  $d$  a partir da Equação 2.4, com a origem
10      em pos_atual e a direção do raio em  $-z$ ;
11      se  $d < dist\_atual$  então
12        dist_atual  $\leftarrow d;
13        cor_atual  $\leftarrow$  cor obtida pela Equação 3.3;
14        Encontrar ponto de interceptação  $p$  a partir da Equação 2.3, com a origem
15        em pos_atual e distância em dist_atual ;
16        para cada outro triângulo  $t_{\text{outro}}$  em  $T$  que não seja  $t$  faça
17          Saber se há interceptação com esse outro triângulo  $t_{\text{outro}}$  a partir da
18          Equação 2.4, com a origem em  $p$  e a direção do raio  $\ell$ ;
19          se houver interceptação com o triângulo  $t_{\text{outro}}$  então
20            cor_atual  $\leftarrow (0, 0, 0, 255)$  (Cor preta, valor mínimo de cor);
21            não verificar outros triângulos (quebrar laço);
22
23   imagem $i,j$   $\leftarrow cor\_atual
24 retorna imagem$$$$ 
```

Agora, é possível construir uma matriz MS com o mapa de sombras correspondente com base nas matrizes *imagem* e *RV*. Se o pixel (i, j) estiver fora da região viável, faz-se $MS_{i,j} = -1$. Caso contrário, tem-se duas situações. Se o valor armazenado em $imagem_{i,j}$ for preto opaco, isto é, $(0, 0, 0, 255)$, o pixel estará sob sombra. Isto será indicado fazendo $MS_{i,j} = 1$. Caso contrário, faz-se $MS_{i,j} = 0$. O Algoritmo 4 sumariza este processo sob a forma de pseudocódigo.

Algoritmo 4: Determinação das áreas sombreadas.

Dados: Matriz $imagem$; Matriz RV ; Q_v ; Q_h ;

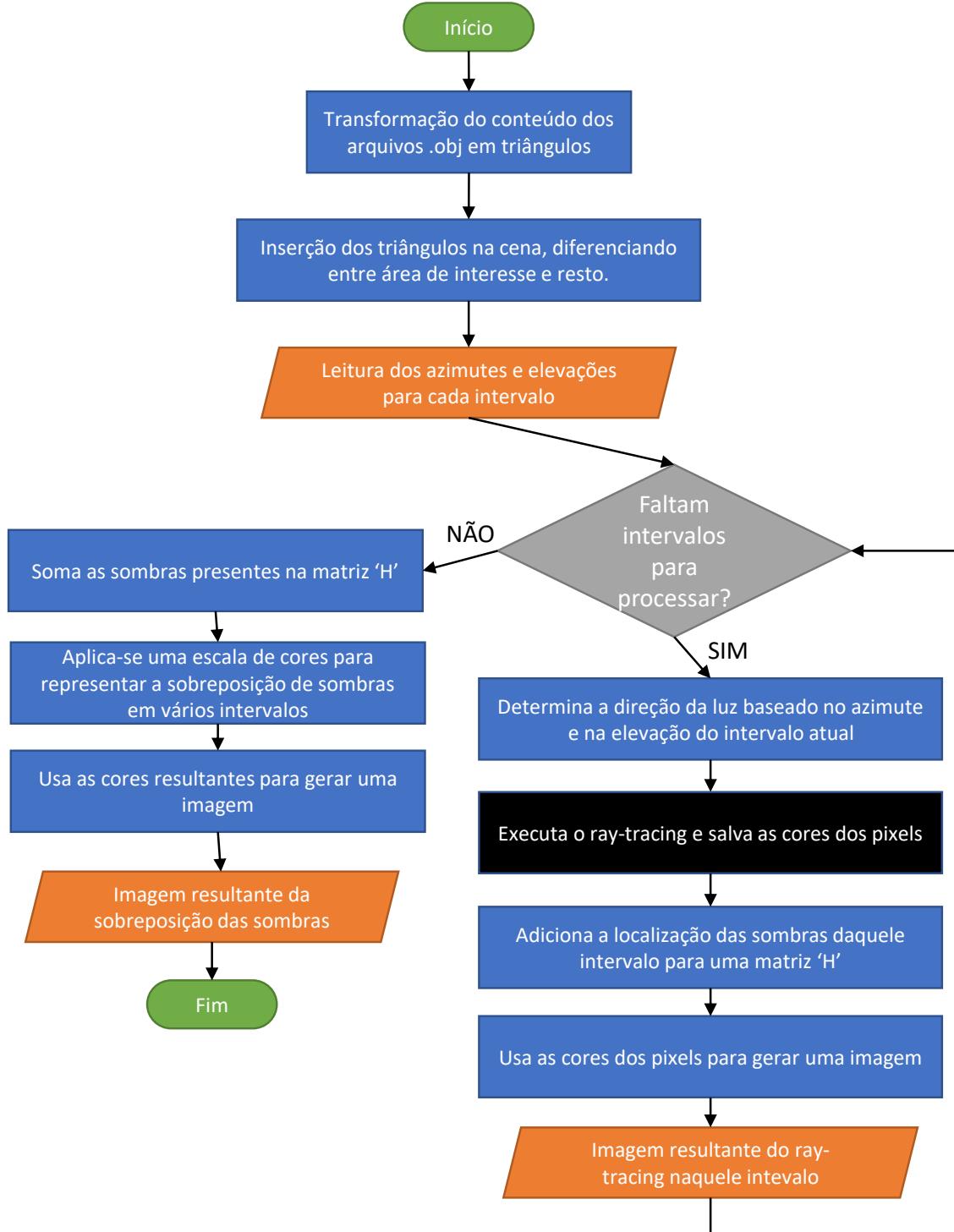
Resultado: Matriz com indicação dos *pixels* onde há sombra

```

1 inicialização;
2  $MS \leftarrow$  matriz vazia de dimensões ( $Q_v, Q_h$ );
3 para  $i \leftarrow 0$  até  $Q_v$  faça
4   para  $j \leftarrow 0$  até  $Q_h$  faça
5     se  $RV_{i,j}$  for nulo (não existe valor) então
6       |    $MS_{i,j} \leftarrow -1$ ;
7     senão
8       |   se  $imagem_{i,j} = (0,0,0,255)$  (cor preto) então
9         |     |    $MS_{i,j} \leftarrow 1$ ;
10        senão
11        |     |    $MS_{i,j} \leftarrow 0$ ;
12 retorna  $MS$ 
```

O código tem como produto final uma imagem para cada hora desejada do dia, e um campo escalar de sombreamento H , resultante do somatório de todas as matrizes MS produzidas a partir do Algoritmo 4. Atribuindo-se cores para a intensidade dos valores que se situarão entre 0 e um número P_l de intervalos considerados, o programa retorna uma imagem que apresenta as regiões onde há maior ou menor sombreamento ao longo do dia. Note que ao usar o valor de -1 nos *pixels* fora da região viável, o valor destes *pixels* fica cada vez mais negativo a medida que as matrizes se somam, não entrando na faixa de cores da imagem produzida, que começa em 0. Um fluxograma simplificado da etapa de geração do campo escalar de sombreamento é mostrado na Figura 31. As matrizes retornadas pelos Algoritmos 2 e 4 serão os principais dados de entrada da etapa de alocação das placas solares.

Figura 31 – Fluxograma do algoritmo implementado para produzir o campo de sombreamento.



Fonte: Autoral.

3.1.4 Validação do modelo de traçado de raios

Para verificar se o modelo implementado se aproxima da realidade, foram conduzidos experimentos com maquetes físicas reais, as quais foram modeladas no computador e processadas pelo programa desenvolvido. A primeira observação ocorreu em 20 de setembro de 2022, em

Juazeiro do Norte/CE, com dois modelos: um quadrangular (Figura 32) e um circular (Figura 33), criados a partir da reutilização de material descartável. Os dados de azimute e elevação para esta data e localidade estão na Tabela 1.

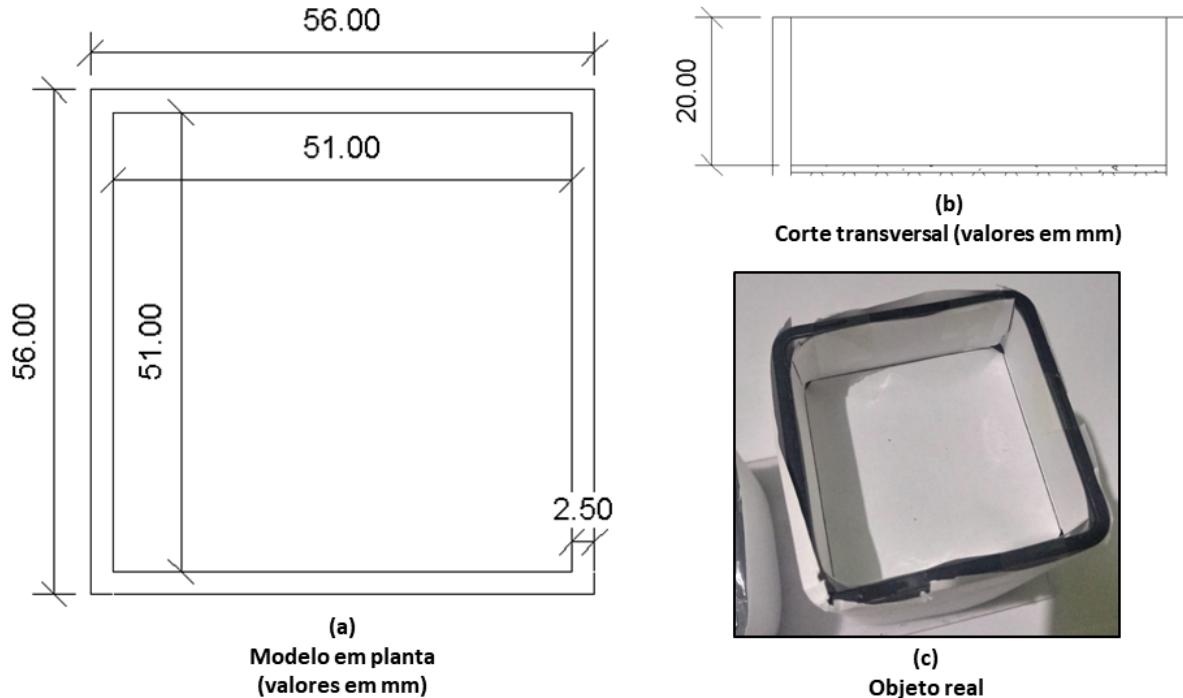
Tabela 1 – Informações de elevação e azimute para o dia 20/09/2022.

| Dados - 20 set 2022 - Juazeiro do Norte (CE) | | |
|--|--------------|-------------|
| Hora | Elevação (°) | Azimute (°) |
| 8h | 36,81 | 83,29 |
| 9h | 51,53 | 79,21 |
| 10h | 65,95 | 71,00 |
| 11h | 78,81 | 43,42 |
| 12h | 79,04 | 317,80 |
| 13h | 66,27 | 289,17 |

Coordenadas: 7° 12' 57,176"S; 39° 18' 44,088"W

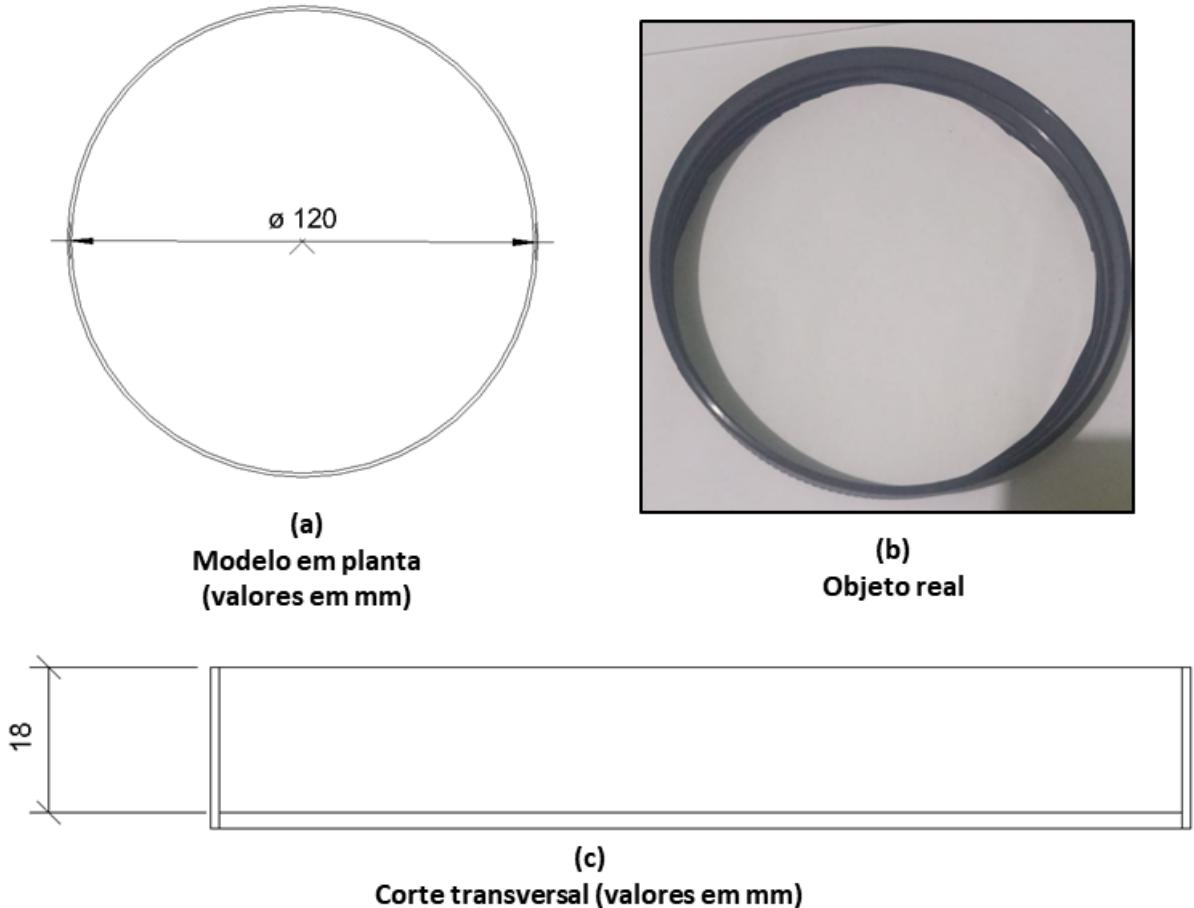
Fonte: SunEarthTools (2022).

Figura 32 – Primeiro modelo para validação do traçado de raios.



Fonte: Autoral.

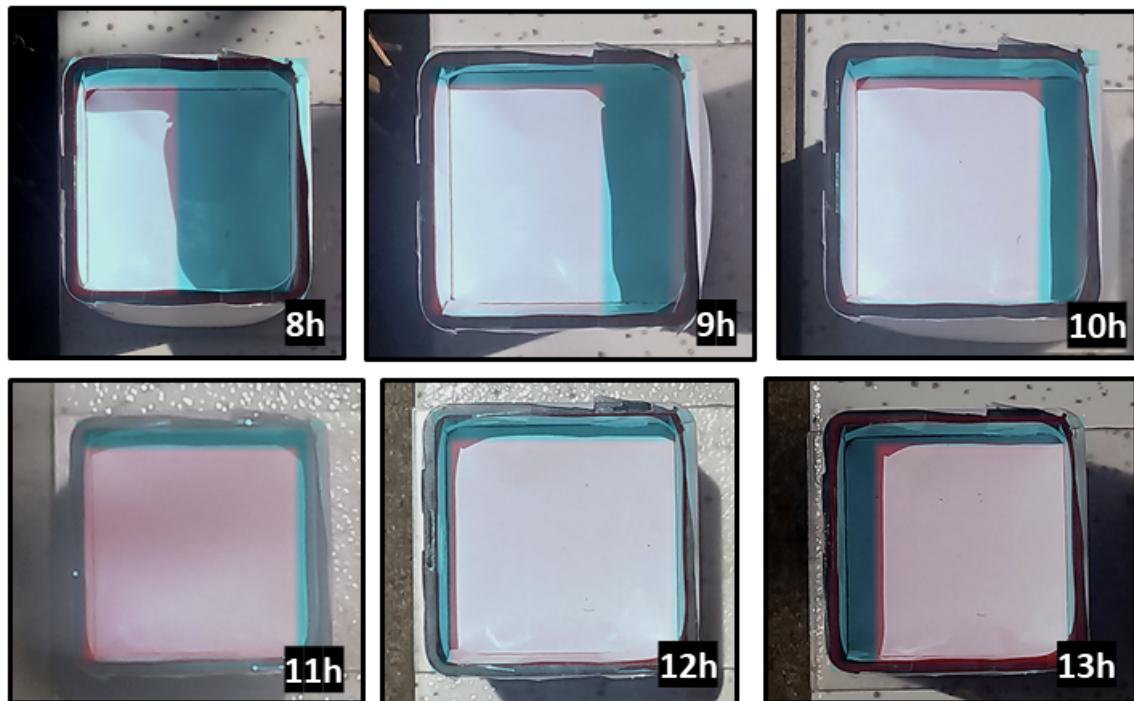
Figura 33 – Segundo modelo para validação do traçado de raios.



Fonte: Autoral.

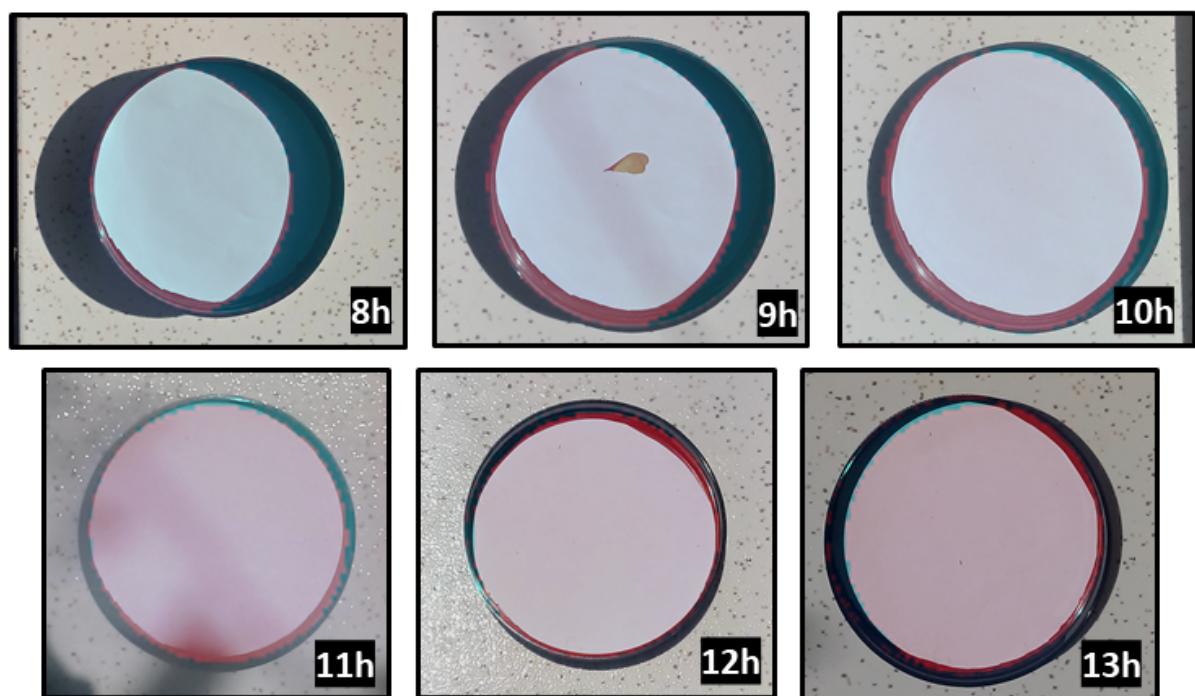
Para cada hora do dia foram tiradas fotos utilizando um telefone celular. O modelo foi nivelado e referenciado ao norte também a partir de aplicativos de celular. As imagens renderizadas pelo programa, para os períodos apresentados na Tabela 1 foram sobrepostas às imagens tiradas nos mesmos horários, como apresentado nas Figuras 34 e 35. A tonalidade em ciano é a sombra produzida pelo programa.

Figura 34 – Resultados do primeiro modelo de validação.



Fonte: Autoral.

Figura 35 – Resultados do segundo modelo de validação.



Fonte: Autoral.

Por mais que seja possível a constatação visual da proximidade do encaixe da sombra real com a sintetizada computacionalmente, este teste em escala possui duas limitações principais. A primeira é a impossibilidade de se obter a área sombreada quantitativamente, e a segunda é o fato de que as fotos tiradas por uma câmera de celular, como também qualquer outra câmera, possuem perspectiva: diferentemente das plantas nas Figuras 32 e 33, e nas imagens produzidas pelo programa, que representam os objetos em uma projeção paralela, a câmera do celular produz uma imagem a partir dos raios luminosos que são detectados pelo sensor após serem concentrados pela lente, produzindo uma imagem em perspectiva. Por exemplo, enquanto a planta em (a) da Figura 32 mostra apenas o topo das paredes do objeto, as imagens da Figura 34 mostram o topo e também a face de algumas paredes.

Com o intuito de contornar estes problemas, foi produzido um terceiro modelo, cuja representação na realidade foi feita em isopor, como mostrado na Figura 36. Nele, o piso é vazado, possibilitando a inserção de um papel milimetrado, onde os limites da sombra foram desenhados a lápis, sendo este papel posteriormente escaneado. Desta forma, é possível obter uma planificação da sombra, propícia à sobreposição das imagens produzidas pelo programa, como também medidas das sombras que podem ser guardadas e conferidas após a observação, que ocorreu no dia 21 de outubro de 2022. Os dados de azimute e elevação para a referida data, em Juazeiro do Norte - CE, estão listados na Tabela 2. As fotos tiradas nos períodos listados na Tabela 2 estão na Figura 37.

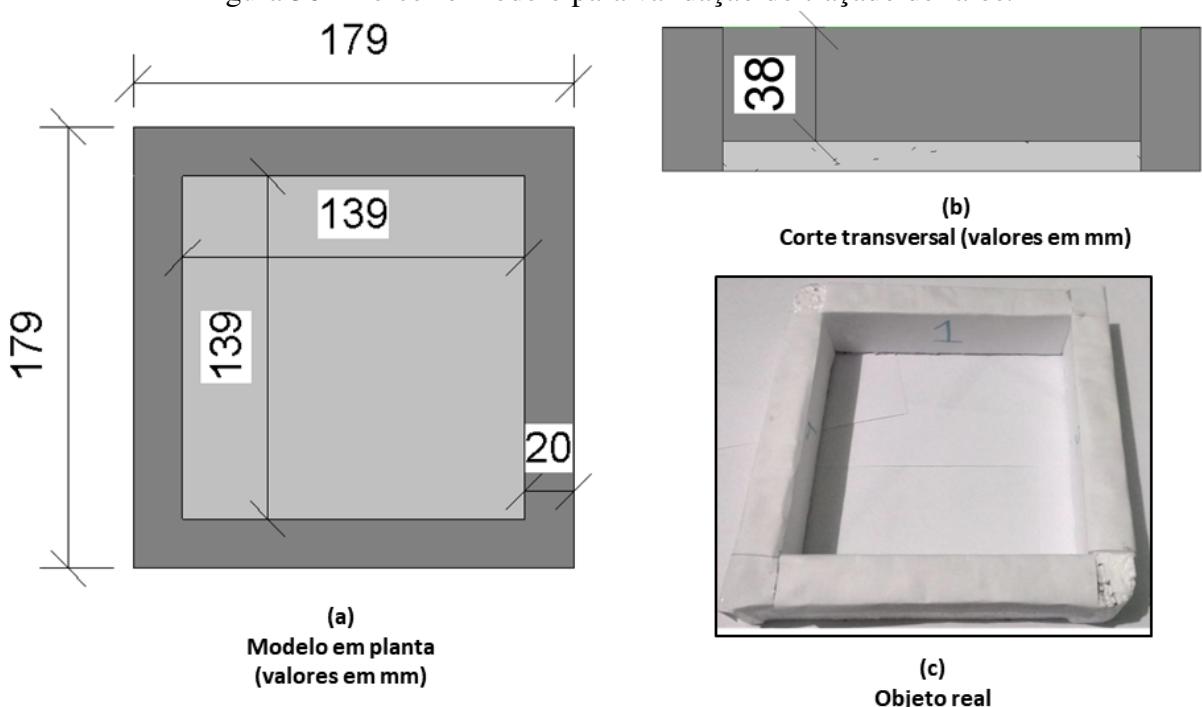
Tabela 2 – Informações de elevação e azimute para o dia 21/10/2022.

| Dados - 21 out 2022 - Juazeiro do Norte (CE) | | |
|--|--------------|-------------|
| Hora | Elevação (°) | Azimute (°) |
| 8h | 40,07 | 98,03 |
| 9h | 54,80 | 98,50 |
| 10h | 69,47 | 101,58 |
| 11h | 83,50 | 124,09 |
| 12h | 79,92 | 248,26 |
| 13h | 65,51 | 259,64 |

Coordenadas: 7° 12' 57,176"S; 39° 18' 44,088"W

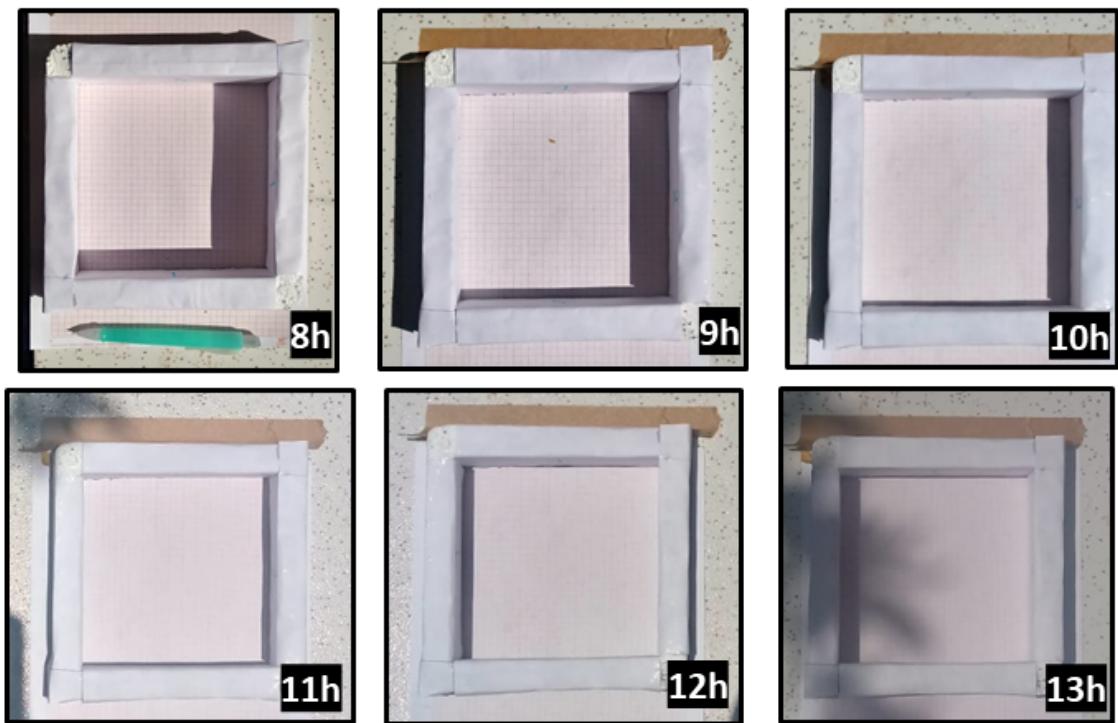
Fonte: SunEarthTools (2022).

Figura 36 – Terceiro modelo para validação do traçado de raios.



Fonte: Autoral.

Figura 37 – Fotos tiradas durante a observação do terceiro modelo.



Fonte: Autoral.

A sobreposição entre as imagens geradas pelo programa e a planificação das sombras durante a observação estão presentes na Figura 38, com a silhueta desenhada em lápis representando a sombra aferida durante as observações, e a região em ciano representando a sombra resultante do programa. Para calcular a área das sombras geradas pelo programa, produziu-se um código que utiliza os mapas de calor separados de cada imagem (0 onde não há sombra e 1 onde há sombra) e a área de um *pixel* (a partir da densidade de *pixels* γ_p). A área será a quantidade de *pixels* com valor 1, multiplicada pela área de um *pixel*. Já as medidas da planificação são obtidas com ajuda das marcações do papel milimetrado. Os resultados são mostrados na Tabela 3.

Tabela 3 – Comparativo entre as áreas sombreadas obtidas.

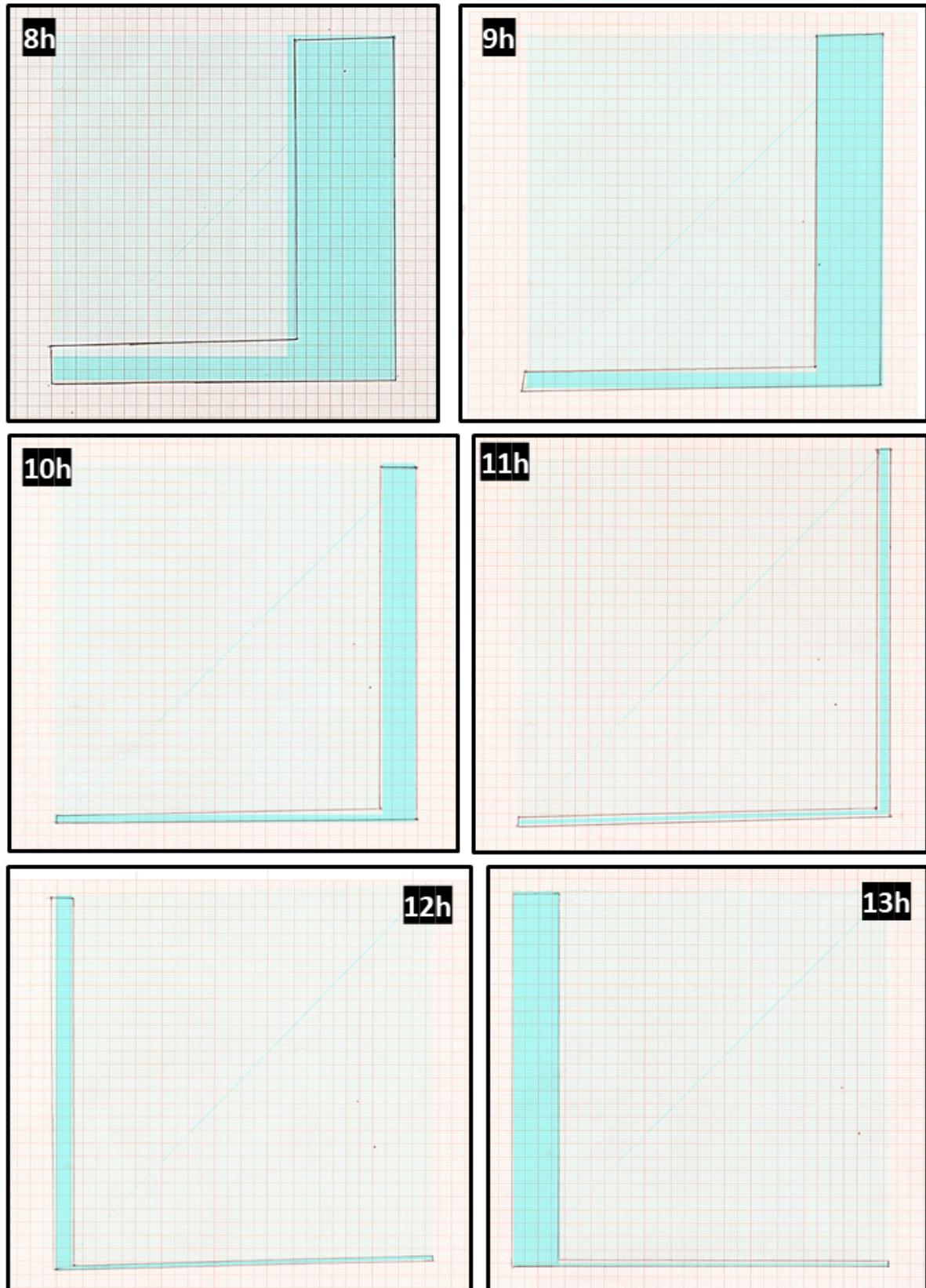
| Resultados - Área de sombreamento - Terceiro modelo de Validação - 21 out 2022 | | | | |
|--|---|--|---------------------------------------|--------------------------|
| Hora | Área planificada na observação (cm ²) | Área obtida pelo programa (cm ²) | Diferença absoluta (cm ²) | Diferença percentual (%) |
| 8h | 71,44 | 71,3248 | 0,1152 | 0,16 |
| 9h | 45,18 | 44,152 | 1,028 | 2,33 |
| 10h | 23,84 | 24,7744 | -0,9344 | -3,77 |
| 11h | 10,97 | 9,1616 | 1,8084 | 19,74 |
| 12h | 12,4 | 11,9408 | 0,4592 | 3,85 |
| 13h | 26,755 | 26,2144 | 0,5406 | 2,06 |

Fonte: Autoral.

Analizando os resultados da Tabela 3, percebe-se que as diferenças percentuais entre as sombras reais e as obtidas no programa não são significativas, com excessão da medição de 11h. Como nesse período a silhueta da sombra já é muito fina, qualquer imprecisão no traçado causa diferenças percentuais mais acentuadas. Além disso, como fica mais perceptível nos horários de 11h e 12h, quando a silhueta está mais afilada, a maquete em isopor tem um desvio de esquadro na base, que pode ser outra fonte de imprecisão da medição.

Os resultados da Figura 38 e da Tabela 3 se complementam, porém uma baixa diferença percentual entre as áreas não significa que a silhueta tem um bom ajuste. Por exemplo, no horário de 8h, tem-se a menor diferença percentual das observações, muito abaixo de todas as outras, porém é possível perceber na Figura 38 que a sombra produzida pelo programa ultrapassa a silhueta traçada na parte direita, mas é significativamente menor que a silhueta na base. Estas áreas, portanto, se compensam, resultando na baixa diferença percentual.

Figura 38 – Resultados visuais do terceiro modelo de validação.



Fonte: Autoral.

São inúmeras as fontes de erro da aferição destas sombras. O manuseio de bússola e de nível de bolha exige um operador experiente, visto que a oscilação entre valores é normal nestes instrumentos. O céu parcialmente nublado pode criar uma sombra com região de penumbra, podendo promover distorções no traçado da silhueta. Uma maquete feita com material leve pode se mover indesejadamente, sendo necessário algum método para fixação, como ocorre a partir das 9h na Figura 37. Além disso, a precisão das medidas do modelo tridimensional são costumadas refletir em uma maquete, mesmo com materiais de alta qualidade.

3.2 Alocação automática de módulos fotovoltaicos

A alocação dos módulos fotovoltaicos, também chamados de placas solares ou simplesmente placas ao longo do texto, será realizada por intermédio de uma heurística de otimização baseada no trabalho de Dür e Nowak (2008), mas agora serão levadas em consideração informações de sombreamento previamente construídas na fase anterior. Além da consideração das sombras, algumas adições ou modificações do modelo de Dür e Nowak (2008) precisam ser feitas para aplicabilidade no problema real, a citar:

- consideração da inclinação do telhado e das placas, quando houver;
- consideração de um telhado de múltiplas águas;
- consideração de um modelo não alinhado ao norte;
- consideração da possibilidade de utilizar placas em orientações diferentes.

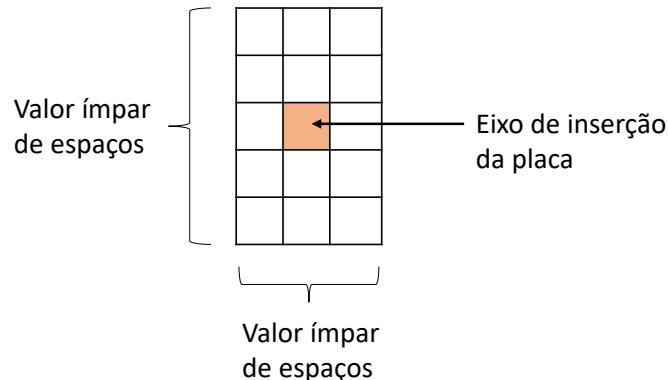
3.2.1 Considerações sobre a heurística desenvolvida

O primeiro passo para a aplicação da metodologia de Dür e Nowak (2008) é a discretização do espaço disponível para os módulos. O trabalho delas utiliza medidas exatas dos limites do polígono onde se deseja inserir os retângulos, gerando neste polígono uma malha que atende a dois critérios:

- a malha deve ter um espaçamento de forma que as dimensões do módulo sejam um múltiplo ímpar de espaços desta malha (Figura 39), garantindo que haja sempre um espaço geometricamente no centro do módulo, que será utilizado como referência para a locação;
- os espaços desta malha devem ser menores que as aberturas dentro do polígono.

Um algoritmo que retorne quantitativamente o valor do espaçamento desta malha não é explicitado no trabalho de Dür e Nowak (2008).

Figura 39 – Hipótese de Dür e Nowak (2008) sobre a quantidade de subdivisões de uma placa

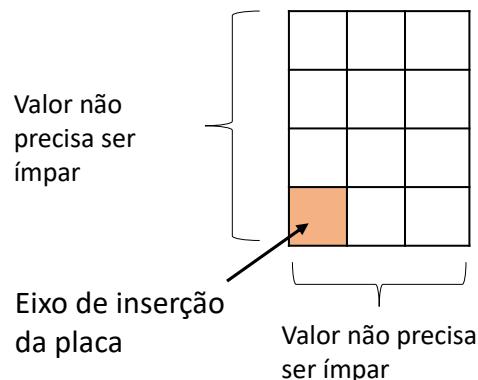


Fonte: Autoral.

Na solução aqui proposta, o mais natural é que se utilize a discretização ocorrida na etapa (i) do trabalho, tornando o *pixel* a menor unidade de área. Por simplicidade, as dimensões das placas são arredondadas para valores múltiplos das dimensões de um *pixel*. Naturalmente, quanto maior for o valor da densidade de *pixels* γ_p da imagem, menor será o erro introduzido. De fato, se por acaso a unidade do modelo for o metro e a precisão do modelo for de centímetros (precisão de 0,01), um valor de γ_p igual a 100 (cem *pixels* por metro), tornaria a medida de um *pixel* igual a 1 cm de lado, sendo desnecessário o arredondamento.

Uma primeira diferença entre a heurística aqui proposta e àquela de Dür e Nowak (2008) surge devido ao esquema de arredondamento adotado para as placas solares. Aqui, as placas serão locadas usando como base seu canto inferior esquerdo, como se pode ver na Figura 40. Nesse caso, não há mais necessidade de se criar um espaço/pixel localizado no centro da placa. Com isso, a quantidade de subdivisões nas duas direções principais podem ser ímpares ou pares.

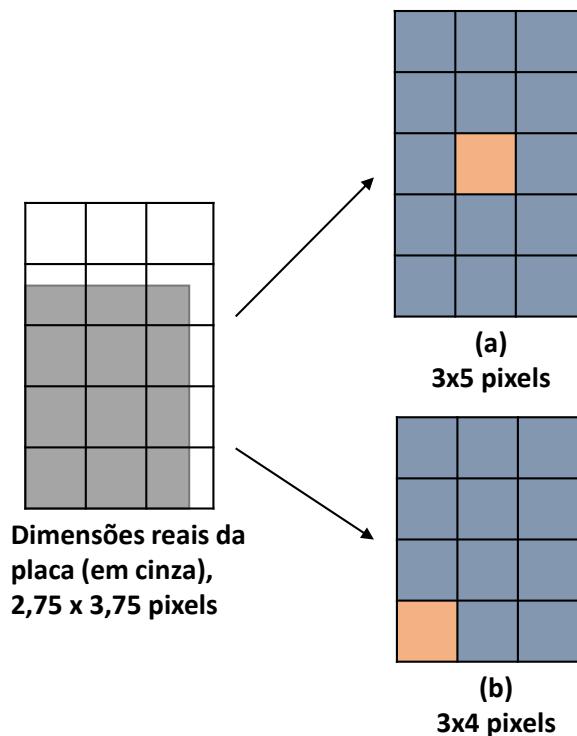
Figura 40 – Pixel base para a inserção de uma placa na heurística proposta.



Fonte: Autoral.

Uma placa solar como a apresentada no exemplo da Figura 41, teria uma dimensão muito maior no modelo devido ao arredondamento, se medidas ímpares fossem obrigatórias, como mostrado em (a), o que não ocorre no arredondamento feito pelo programa, que não obriga um número ímpar de espaços no arredondamento, em (b). Esta afirmação não é uma crítica ao modelo de Dür e Nowak (2008): no modelo delas, as dimensões da placa e dos furos no telhado são o que determinam a quantidade de subdivisões, enquanto que na metodologia proposta neste trabalho, esta informação é dependente da imagem produzida anteriormente, sendo necessário adaptar as dimensões da placa a esta imagem.

Figura 41 – Diferença causada pela obrigatoriedade de dimensões ímpares de espaços.



Fonte: Autoral.

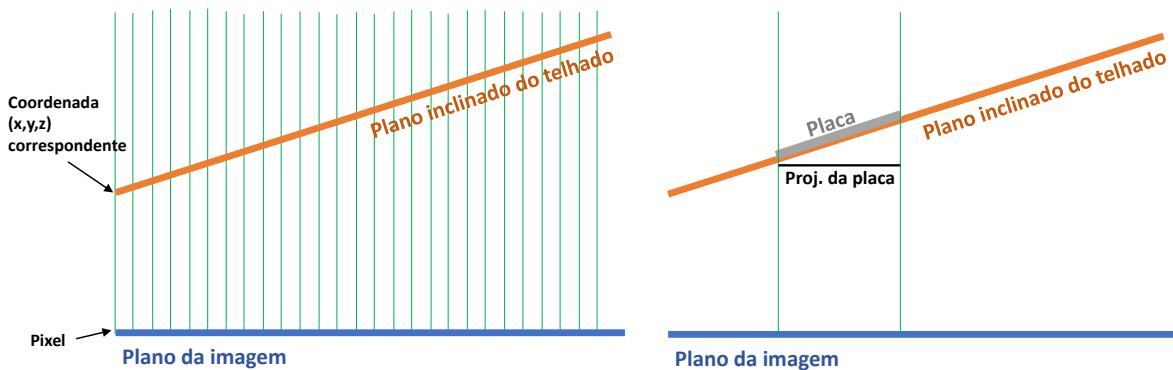
Uma complicador do problema real é sem dúvida sua natureza tridimensional. Enquanto a instalação de placas solares ocorre, geralmente, em um plano inclinando, o campo de sombreamento produzido possui apenas duas dimensões. Assim, o problema real seria um problema de empacotamento tridimensional. Para reduzí-lo ao caso bidimensional, são adotadas as seguintes hipóteses: (a) as dimensões de uma placa em *pixels* serão baseadas na sua projeção ortogonal sobre o plano da imagem e (b) somente será possível processar uma águia do telhado por vez, em telhados de múltiplas águias.

Como se pode perceber pela Figura 42, a imagem gerada na etapa (i) não distorce as distâncias no plano x - y do modelo, como numa planta arquitetônica, porém a existência de uma inclinação no eixo z dos objetos do modelo implica que a dimensão no sentido da inclinação da placa solar é menor na imagem do que na realidade. Em outras palavras, a dimensão real da placa, advinda do manual do fabricante, é a hipotenusa à inclinação, mas a dimensão que deve constar na imagem planificada é a do cateto adjacente à inclinação. Portanto, é necessário que se corrija a dimensão inclinada da placa, a partir da equação:

$$dim_{\text{proj}} = dim_{\text{incl}} \cdot \cos \alpha, \quad (3.6)$$

onde dim_{proj} é a dimensão projetada na imagem, dim_{incl} é a dimensão a ser inclinada, e α é a inclinação do telhado (ou região viável). A dim_{incl} dependerá da orientação desejada para a placa (vertical ou horizontal) e também do sentido da inclinação da região viável (se é no eixo x ou no eixo y da imagem). A orientação e o ângulo α da inclinação da região viável são detectados automaticamente, com base nas coordenadas x , y e z de seus *pixels* correspondentes, como ilustrado na Figura 42. As dimensões utilizadas para o arredondamento da quantidade de *pixels* por placa são aquelas corrigidas pela Equação 3.6. No caso de cobertas planas ou quase planas, como lajes, geralmente as placas não seguem a inclinação da laje, e são instaladas com uma inclinação mínima para uma melhor eficiência do sistema. Nesse caso, o usuário deverá fornecer o valor de α manualmente.

Figura 42 – Estratégia de planificação do problema.

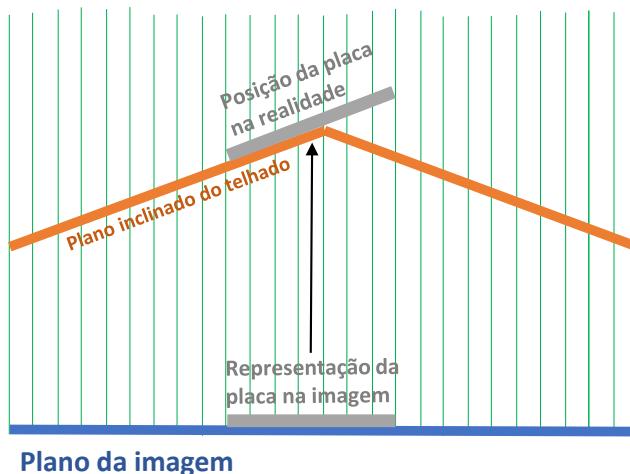


Fonte: Autoral.

Quanto a um modelo de múltiplas águas, como o algoritmo utilizará o plano da imagem gerada para inserção das placas solares, uma placa poderia ser inserida em um local onde há mudança de inclinação, mesmo dentro da região viável, reproduzindo locações que não

são usuais na realidade (Figura 43). Para contornar este problema, basta separar as águas do telhado e processá-las uma de cada vez. Outra possibilidade seria sempre verificar se todos os *pixels* cobertos por uma placa possuem a mesma inclinação. Contudo, isto pode levar à criação de um número razoável de tentativas frustradas, o que impactaria significativamente no custo computacional da fase de alocação.

Figura 43 – Ilustração da dificuldade no tratamento de telhados com duas águas.



Fonte: Autoral.

No código desenvolvido, cada placa solar é representada por uma classe Placa, contendo as seguintes características:

- sua identificação numérica, chamada de *ID*, referente à ordem na qual foi inserida;
- sua cor, apenas ilustrativa, que por padrão é uma cor aleatória;
- as coordenadas reais de inserção no modelo, ou seja, as coordenadas das bordas da placa em medidas reais;
- as coordenadas dos *pixels* pertencentes ao contorno da placa;
- sua pontuação relativa ao sombreamento, aqui denominada de *coeficiente de sombreamento*¹. Esta informação será útil para classificar resultados da heurística de inserção de placas, como será visto adiante.

Para decidir se é possível inserir uma placa solar em uma determinada região, duas matrizes são fundamentais: a matriz dos campos de sombreamento acumulados, doravante denominada *mapa de sombreamento*, e uma matriz de igual dimensão, aqui denominada *mapa de locação*. Na heurística proposta, cada *pixel* define um potencial canto inferior esquerdo

¹ Não confundir com o termo homônimo utilizado na indústria da construção civil para designar uma medida de desempenho térmico de esquadrias de vidro.

para a inserção de uma nova placa. Para exemplificar este processo, será usado o modelo da Figura 30, agora com γ_p igual a 3 (valor muito pequeno para permitir a visualização dos *pixels*), considerando quatro intervalos de tempo. Assume-se que as placas estão na posição vertical, com medidas de 3×4 *pixels*, tal como na Figura 41(b).

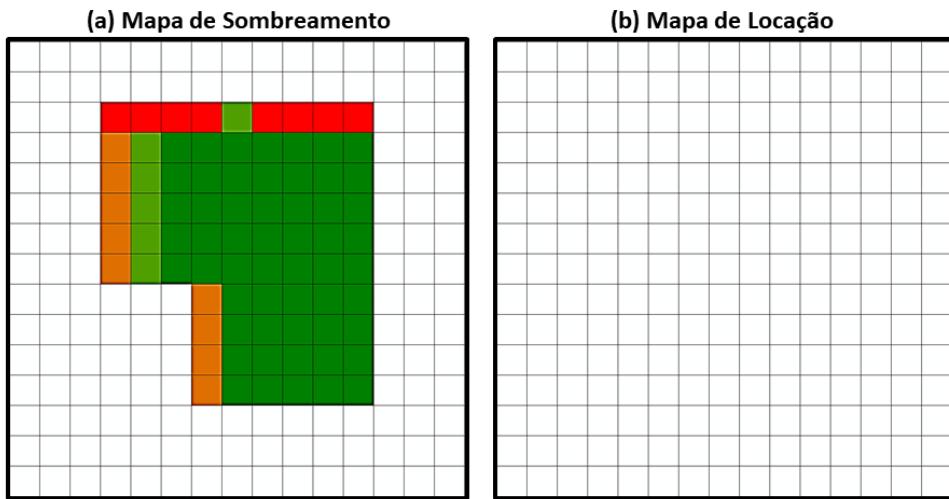
A Figura 44 traz o mapa de sombreamento calculado, tanto com as cores atribuídas ao sombreamento, quanto com os valores numéricos associados. O valor *zero* indica que não houve sombreamento em nenhum horário e o valor 4 indica que o *pixel* permaneceu sombreado o tempo todo. Os valores negativos indicam *pixels* fora da região viável. O estado inicial dos mapas de sombreamento e locação é mostrado na Figura 45.

Figura 44 – Mapa de sombreamento do exemplo.

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 4 | 4 | 4 | 4 | 1 | 4 | 4 | 4 | 4 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | 3 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | 3 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | 3 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | 3 | 0 | 0 | 0 | 0 | 0 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 |
| -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 | -4 |

Fonte: Autoral.

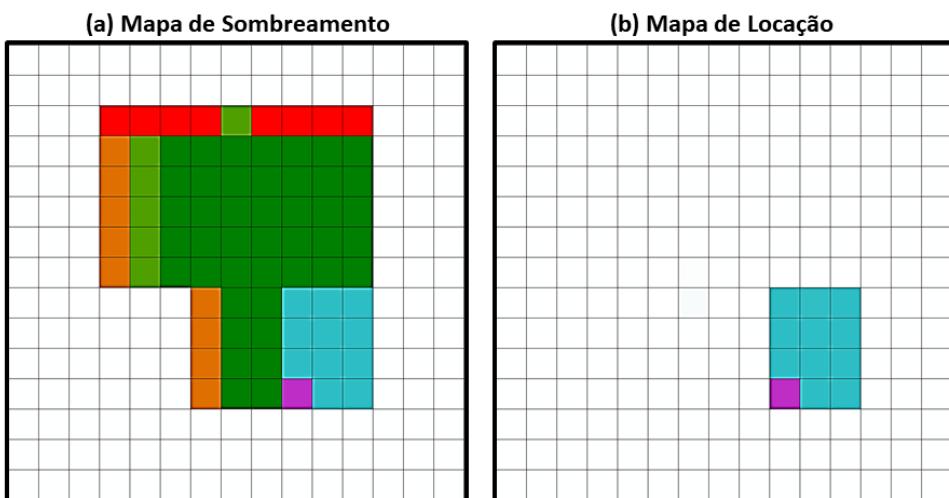
Figura 45 – Estado inicial dos mapa de sombreamento e locação.



Fonte: Autoral.

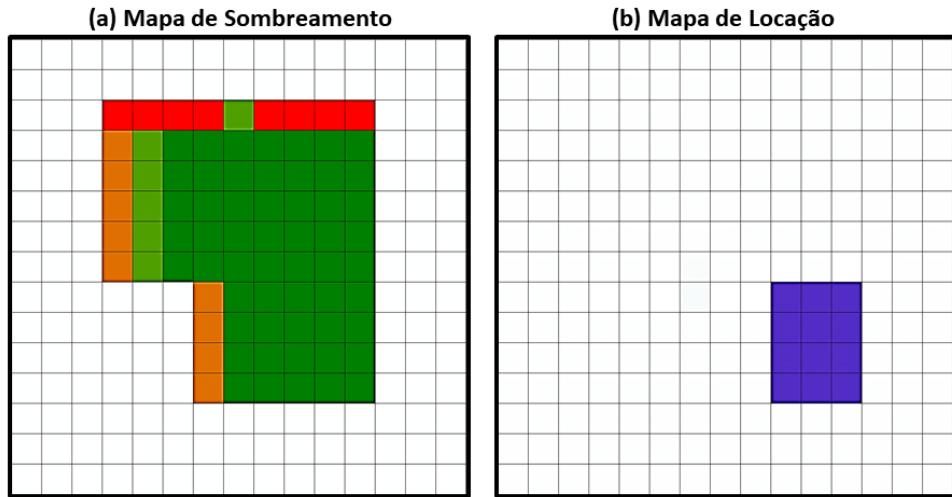
Suponha que, iniciando do canto inferior direito da imagem, os *pixels* são processados até chegar no indicado pela cor rosa na Figura 46. Para locar a placa solar, todos os *pixels* em azul claro (que delimitam o tamanho da placa) deverão atender a duas condições ditas *ideais*: estarem em lugar sem sombreamento (valor nulo no mapa de sombreamento) e não sobreponem nenhuma outra placa no mapa de locação. Como se pode verificar na Figura 44, todos os valores na região em azul claro têm valor 0 e o mapa de locação, na Figura 45, está completamente vazio, permitindo a locação de uma nova placa nesta posição. Uma nova placa é então inserida, armazenando-se seu *ID* nos respectivos coeficientes do mapa de locação e, com isso, o estado das matrizes passa a ser o indicado na Figura 47. Durante a inserção, é criado um objeto da classe Placa que é guardado em uma lista.

Figura 46 – Pixel atual na verificação da imagem para locação da placa.



Fonte: Autoral.

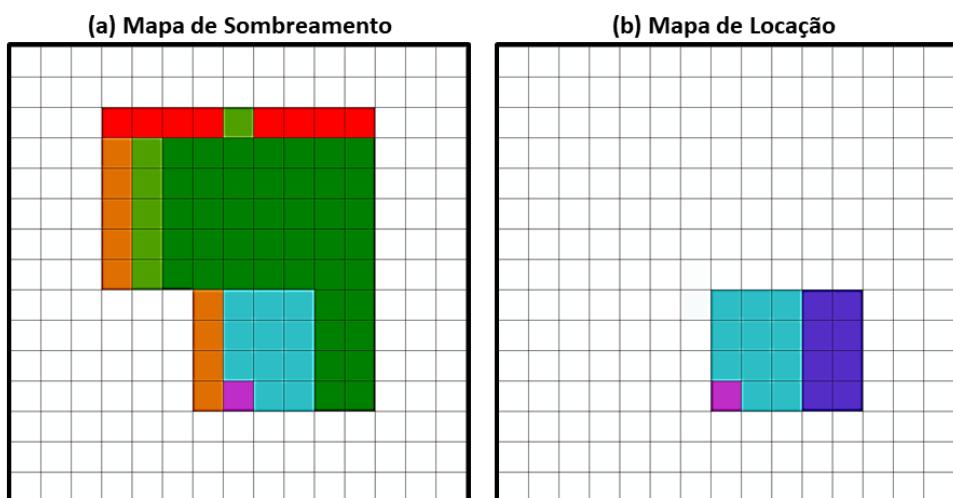
Figura 47 – Novo estado das matrizes após a inserção da placa da Figura 46.



Fonte: Autoral.

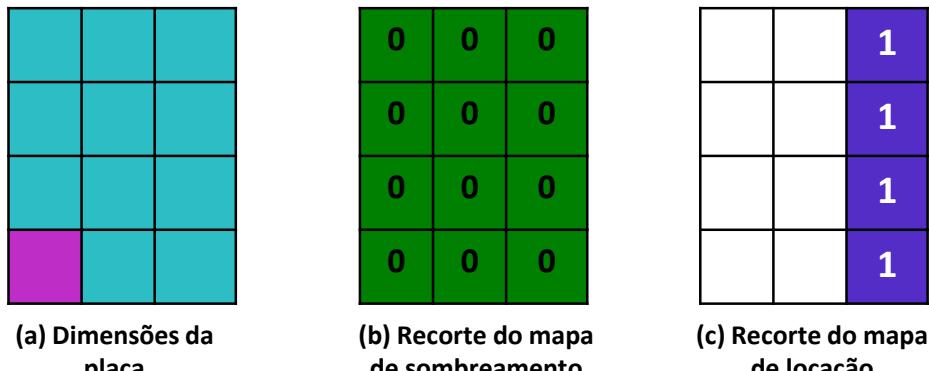
Agora, fosse verificada a possibilidade da inserção de uma placa na posição indicada na Figura 48, não seria possível inserí-la. Como se percebe em um recorte das duas matrizes apenas na região analisada (Figura 49), por mais que todos os valores na matriz mapa de sombreamento sejam 0, indicando que a área não está sombreada, o mapa de locação possui áreas com valores iguais a 1, indicando que há uma área já ocupada pela placa inserida anteriormente. Como uma placa nesta posição sobreporia uma outra já posicionada, a inserção não é efetuada.

Figura 48 – Segunda posição para inserção de placa.



Fonte: Autoral.

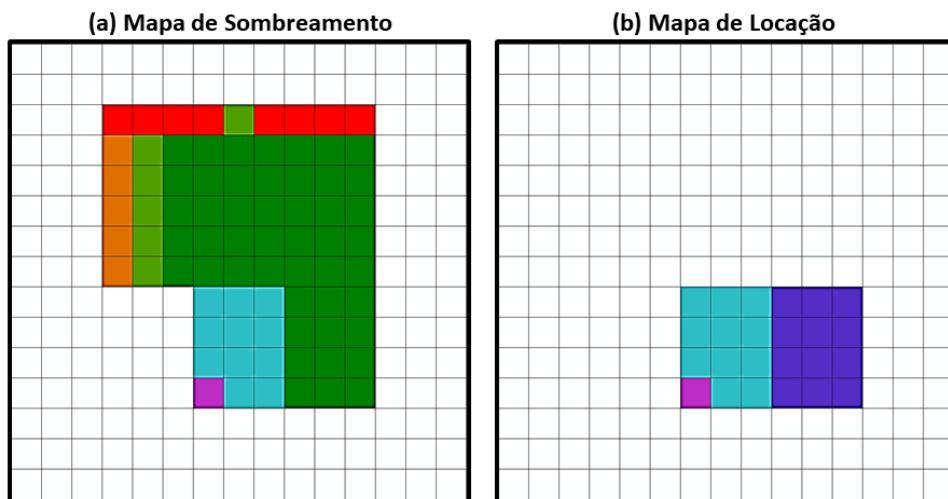
Figura 49 – Recorte da região indicada na Figura 48.



Fonte: Autoral.

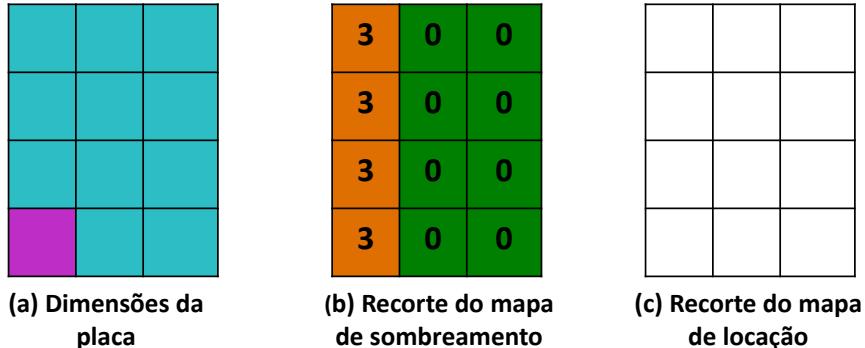
Novamente, se estivéssemos verificando a possibilidade de uma placa na posição indicada na Figura 50, não se poderia inserí-la: um recorte das duas matrizes apenas na região analisada (Figura 51), não indica nenhum valor no mapa de locação, porém existem valores maiores que zero no mapa de sombreamento, indicando que aquela área sofre algum sombreamento durante o período observado. Neste caso, a placa não é inserida, já que estaria em área sombreada.

Figura 50 – Terceira posição para inserção de placa.



Fonte: Autoral.

Figura 51 – Recorte da região indicada na Figura 50.



Fonte: Autoral.

O objetivo do trabalho, porém, não é proibir a inserção de placas solares em áreas sombreadas, mas apenas evitar sempre que possível este comportamento. Dessa forma, se faz necessária a aplicação de algum critério que classifique as placas quanto ao sombreamento, de forma a priorizar a inserção de placas em locais menos sombreados. Como já foi mostrado, o mapa de sombreamento, um campo escalar referente aos sombreamentos acumulados, possui valores inteiros que indicam a quantidade de intervalos de tempo em que aquele espaço, representado por um *pixel*, permanece sombreado. Isto conduz à seguinte definição do coeficiente de sombreamento s de uma placa:

$$s = \frac{\sum_{i=i_0}^{i_f+1} \sum_{j=j_0}^{j_f+1} (P_t - S_{ij})}{P_t(i_f - i_0 + 1)(j_f - j_0 + 1)}, \quad (3.7)$$

onde i_0 é o menor índice vertical de um pixel da imagem coberto pela placa, i_f é o maior índice vertical, j_0 é o menor índice horizontal de um pixel da imagem a ser coberto, j_f é o maior índice horizontal, P_t é o número de períodos considerados na construção do mapa de sombreamento e S_{ij} é o coeficiente no mapa de sombreamento correspondente ao pixel (i, j) . Esta expressão é de certa forma uma “média inversa” dos valores do mapa de sombreamento na região analisada.

Para exemplificar o uso da Equação 3.7, considere o exemplo de inserção das Figuras 50 e 51, onde $P_t = 4$. Os valores necessários para a composição do coeficiente de sombreamento desta placa estão apresentados na Figura 52. Com base neles, obtém-se:

$$s = \frac{(1 \cdot 8) + (0,25 \cdot 4)}{(12 - 9 + 1) \cdot (9 - 7 + 1)} = 0,75. \quad (3.8)$$

Figura 52 – Exemplificação do cálculo do coeficiente de sombreamento da placa da Figura 50.

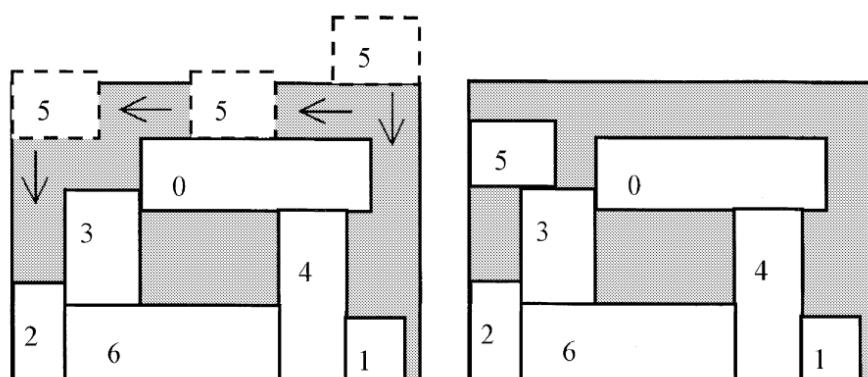
| | | | |
|---|----------------------------------|--|--|
| (a) valores do índice i | (b) valores do índice j | (c) Recorte do mapa de sombreamento | (d) Valor de $(P_t - H_{ij})/ P_t$ |
| 9 9 9 10 10 10 11 11 11 12 12 12 | 7 8 9 7 8 9 7 8 9 7 8 9 | 3 0 0 3 0 0 3 0 0 3 0 0 | 0,25 1 1 0,25 1 1 0,25 1 1 0,25 1 1 |
| | | | |
| | | | |
| | | | |

Fonte: Autoral.

Com este exemplo, é possível verificar que o coeficiente de sombreamento s é função da extensão da área sombreada (quanto maior o número de *pixels* com algum valor de sombreamento, menor o valor de s), como também da duração do sombreamento nesta área (um maior período de sombreamento implica num valor menor de s). Desta forma, a inserção de placas em áreas sombreadas priorizará regiões associadas aos maiores valores de s .

Um dos fatores que mais influencia o resultado da locação das placas solares é a ordem adotada para a travessia da matriz de regiões viáveis. No trabalho de Dür e Nowak (2008), ocorre o aperfeiçoamento de uma heurística de empacotamento chamada *bottom-left* (inferior-esquerdo). Esta heurística estabelece que os objetos devem ser ancorados no ponto mais abaixo e mais a esquerda possível de um recipiente. Para tanto, o objeto é colocado o máximo possível para baixo e, depois, o máximo possível para a esquerda, repetindo isso até que o objeto não possa mais se mover nem para baixo, nem para a esquerda (Figura 53). Assim, por definição, o primeiro objeto será ancorado sempre na extremidade inferior esquerda do recipiente, daí o nome *bottom-left*.

Figura 53 – Heurística clássica de bottom-left.

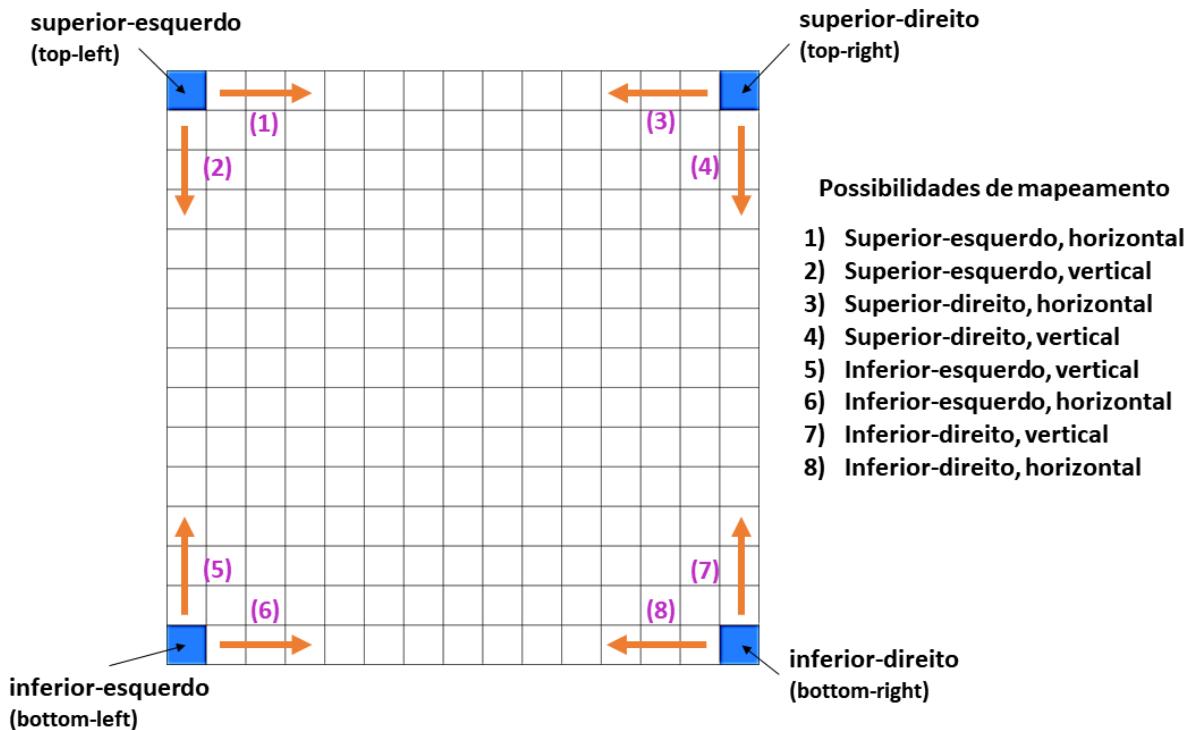


Fonte: Hopper e Turton (2001).

A metodologia utilizada no *bottom-left* clássico e na heurística aprimorada de Dür e Nowak (2008) é diferente da aplicada neste trabalho: por mais que também ocorra a priorização de uma das extremidades do recipiente, o posicionamento do retângulo não passa por estados intermediários, como ilustrado no exemplo de execução do *bottom-left* clássico mostrado na Figura 53. Observe que o item 5 é deslocado incrementalmente, em direções alternadas, até atingir sua posição final.

A rigor, a matriz das regiões viáveis pode ser percorrida partindo-se de qualquer coeficiente e dando preferência a diferentes pontos de ancoragem. De fato, no total, há uma quantidade exponencial de mapeamentos possíveis. Por simplicidade, neste trabalho, foram considerados apenas quatro pontos de partida distintos, a saber, os cantos inferior-esquerdo, inferior-direito, superior-esquerdo e superior-direito da matriz; com pontos de ancoragem preferenciais homônimos. Além disso, duas ordens triviais de travessia foram utilizadas: horizontal primeiro e vertical primeiro. Isso tudo resulta em 8 mapeamentos distintos (Figura 54).

Figura 54 – Casos possíveis de travessia da região viável.

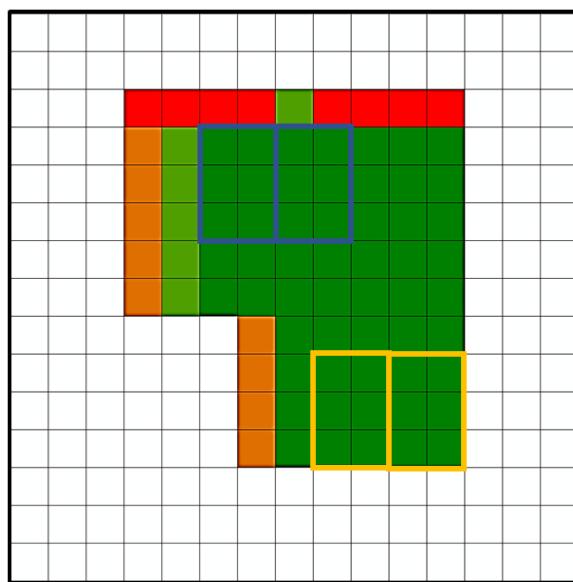


Fonte: Autoral.

Embora implementações tradicionais do *bottom-left* empregando diferentes mapeamentos produzam resultados visualmente distintos, a quantidade de objetos alocados permanece quase inalterada. A necessidade de se empregar mais de um mapeamento surge do fato de que

o “recipiente” poderá assumir formas variadas, geradas pelo mapa de sombreamento. Para a maioria dos leiautes de cobertas, as sombras se concentram em algumas das bordas do telhado. Colocar módulos em áreas não-sombreadas, mas que estejam próximas a áreas sombreadas, enquanto há outras áreas livres mais distantes de sombras, não parece prudente. Um exemplo dessa situação está mostrado na Figura 55. As duas placas locadas na silhueta azul, inseridas partindo-se do canto superior-esquerdo com mapeamento horizontal, possuem o mesmo coeficiente de sombreamento ($s = 1$) das placas locadas na silhueta amarela, inseridas a partir do canto inferior-direito com mapeamento horizontal. Contudo, as placas em amarelo encontram-se mais distantes de áreas sombreadas.

Figura 55 – Dois arranjos de placas com o mesmo coeficiente de sombreamento, mas partindo de extremidades diferentes.



Fonte: Autoral.

Em casos envolvendo um número elevado de placas solares, necessitando cobrir quase que totalmente a região viável, a preferência por um ponto de ancoragem específico terá impacto maior apenas no aspecto visual da solução obtida. Já em casos onde o número de placas necessárias é reduzido quando comparado à região viável disponível, diferentes esquemas de ancoragem podem levar a resultados consideravelmente díspares.

3.2.2 *Modalidades de inserção de módulos*

Foram definidas duas modalidades de inserção: livre e alinhada. Na modalidade livre, as matrizes são mapeadas a partir de um dos casos mostrados na Figura 54, utilizando

uma única orientação para as placas solares, horizontal ou vertical, escolhida pelo usuário². Nesta modalidade, as placas são inseridas sempre que as condições ideais são atendidas. Um pseudocódigo desta modalidade está apresentado no Algoritmo 5.

Algoritmo 5: Pseudo-código para a modalidade livre, inserindo apenas nas áreas não sombreadas.

Dados: Matriz H ; Matriz RV ; dimensões da placa em pixels

Resultado: Lista L de placas inseridas; Mapa de locação ML

```

1  $ML \leftarrow \emptyset;$ 
  /* mesmas dimensões de  $H$  */ *
2  $L \leftarrow \emptyset;$ 
3  $count \leftarrow 0;$ 
4 para pixel  $p$  mapeado na ordem definida pelo usuário faz
5   se for possível a inserção (atender às condições ideais) então
6      $count \leftarrow count + 1;$ 
7     Atribuir o valor  $count$  aos coeficientes de  $ML$  correspondendo aos pixels
      cobertos pela nova placa;
8     Adicionar objeto Placa à lista  $L$ ;
9 retorna  $L, ML$ 

```

Sempre que possível, a inserção de uma placa é efetivada nas linhas 6, 7 e 8 do Algoritmo 5. Para tanto, é necessário primeiro determinar os *pixels* que serão cobertos pela nova placa (linha 7). Em seguida, são calculadas e armazenadas as propriedades mencionadas anteriormente, como a coordenada do ponto de ancoragem da placa, seus *pixels* limítrofes e o coeficiente de sombreamento.

A liberdade de escolha das posições das placas solares possibilitada pela modalidade livre muitas vezes conduz a soluções inviáveis do ponto de vista prático. Podem ser obtidos leiautes desalinhados que ficam ainda mais desajeitados com a consideração do coeficiente de sombreamento. Além de desagradáveis esteticamente, eles dificultam a instalação. Este problema será contornado pela modalidade alinhada, como descrito a seguir.

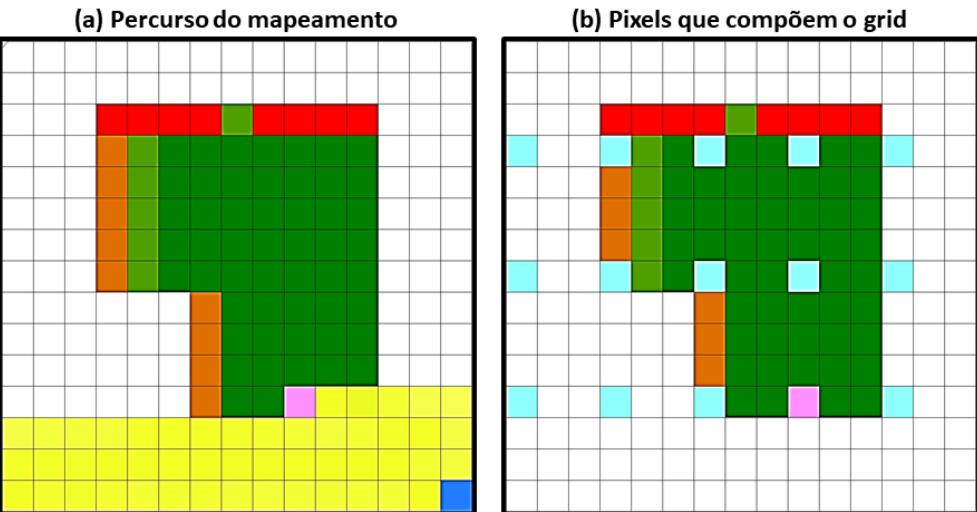
Na modalidade alinhada, as placas permanecerão sempre alinhadas entre si. Isto é feito com o auxílio de uma grade regular auxiliar GV , definida a partir da primeira placa locada, contendo as posições viáveis para a inserção de novas placas. Assim como anteriormente, não é permitida a alternância da orientação das placas.

A Figura 56 ilustra a construção da grade GV de pixels viáveis, para o caso de placas 3 por 4, orientadas verticalmente. Com os *pixels* percorridos da forma mostrada na Figura 56(a),

² Não obstante, a alternância de orientação dentro de um arranjo não é usual em sistemas fotovoltaicos de pequeno porte.

partindo-se do canto inferior-direito (em azul escuro), com mapeamento horizontal, percorre-se os *pixels* em amarelo até atingir o primeiro *pixel* onde será possível inserir uma placa, indicado em rosa, o qual definirá a *origem* da grade. Os retângulos que compõem a grade possuem as mesmas dimensões das placas.

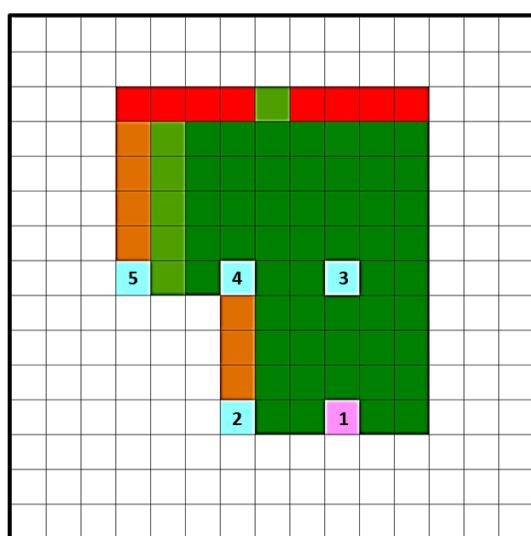
Figura 56 – Formação da grade na modalidade alinhada.



Fonte: Autoral.

Observe que nem todos os *pixels* da Figura 56(b) podem comportar placas. Por este motivo, alguns pixels da grade precisarão ser removidos de *GV*. Note ainda que não será mais necessário verificar a sobreposição entre placas, tendo em vista a forma com que agora estão espaçadas. A Figura 57 contém os pixels de ancoragem viáveis segundo o mapeamento adotado.

Figura 57 – Grade de pixels viáveis, numerados pela ordem estabelecida no mapeamento.



Fonte: Autoral.

É provável que, em alguma situação, seja necessário inserir placas em regiões parcialmente sombreadas para atingir uma determinada potência de pico. Obviamente, quanto menos sombreada for uma placa, melhor para o sistema como um todo. Tendo isso em mente, as placas solares serão inseridas com base nos valores de coeficiente de sombreamento.

O Algoritmo 6 sintetiza os passos de locação da modalidade alinhada.

Algoritmo 6: Heurística de locação de placas solares na modalidade alinhada.

Dados: Matriz H (mapa de sombreamento); Matriz RV referente à região viável; tamanho da placa em pixels; orientação prioritária da placa dada pelo usuário; caso da Figura 54 que determinará a ordem de mapeamento; número n_p de placas pré-determinado.

Resultado: Lista L de placas inseridas; Matriz ML (mapa de locação)

```

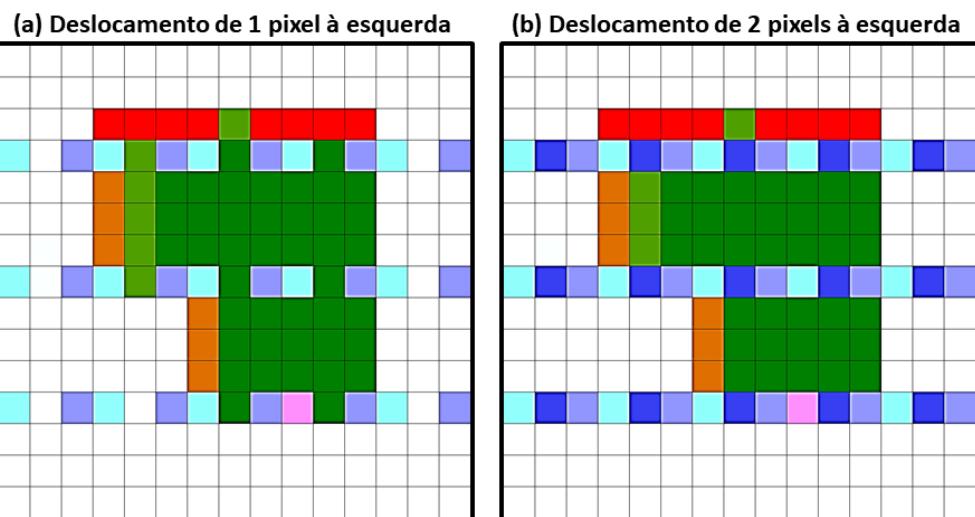
1  $ML :=$  matriz vazia com as mesmas dimensões de  $H$ ;
2  $L \leftarrow \emptyset$ ;
3  $count \leftarrow 0$ ;
4  $GV :=$  lista com todos os pontos do grid em ordem de mapeamento;
5 para pixel  $p$  em  $GV$  faz
6   | se não for possível inserir placa (placa ficaria fora da região viável) então
7   |   | retirar  $p$  de  $GV$ ;
8 para pixel  $p$  em  $GV$  faz
9   | se score  $s$  da placa na posição = 1 &  $count < n_p$  então
10  |   | executar inserção em  $p$ ;
11  |   | retirar  $p$  de  $GV$ ;
12 enquanto  $count < n_p$  faz
13  |   |  $s_{temp} \leftarrow s$ ;
14  |   |  $pos_{temp} \leftarrow$  índices de  $p$ ;
15  |   | se  $GV \neq \emptyset$  então
16  |   |   | para pixel  $p$  em  $GV$  faz
17  |   |   |   | se score  $s$  da placa na posição >  $s_{temp}$  então
18  |   |   |   |   |  $s_{temp} \leftarrow 0$ ;
19  |   |   |   |   |  $pos_{temp} \leftarrow \emptyset$ ;
20  |   |   | senão
21  |   |   |   | Avisar ao usuário que não se pode mais inserir placas;
22  |   |   |   | Encerrar execução (quebrar laço);
23  |   |   | executar inserção em  $pos_{temp}$ ;
24  |   |   | retirar  $pos_{temp}$  de  $GV$ ;
25 retorna  $L, ML$ 

```

A formação da grade de pixels viáveis, como ilustrado na Figura 56(b), depende da origem da grade. Por exemplo, ao mover a origem uma unidade para a esquerda, obtém-se a grade desenhada em lilás na Figura 58(a). Movendo-se a origem duas unidades para a esquerda, o resultado é a grade em azul escuro da Figura 58(b). Caso se mova novamente a

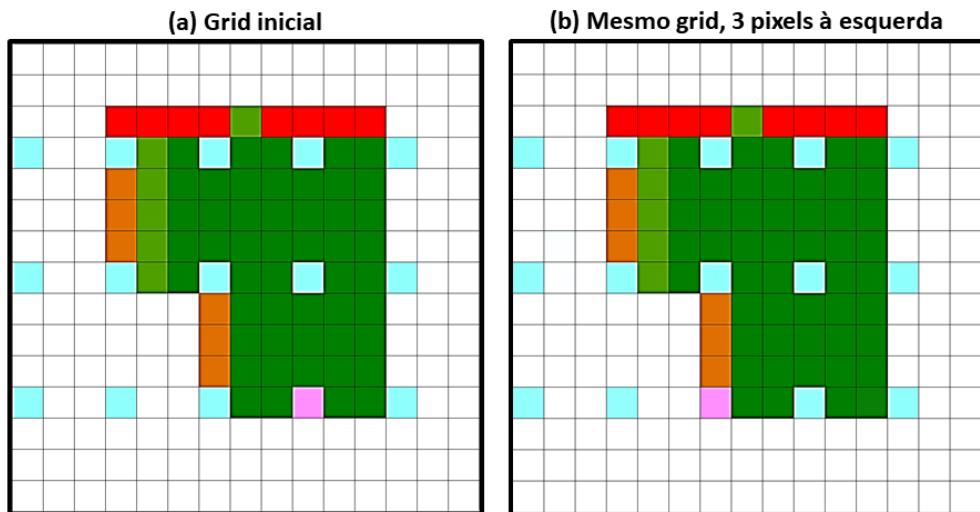
origem uma unidade na mesma direção, obtém-se a grade inicial, como mostra a Figura 59. Esse comportamento também ocorre no caso de deslocamentos verticais. Desta forma, na modalidade alinhada, pode-se determinar todas as locações possíveis. O pixel em rosa juntamente com os *pixels* em marrom na Figura 60 representam todas as opções de origem que correspondem a grades distintas, fixada a orientação vertical. De modo geral, considerando ambos os casos de orientação vertical e horizontal, há um total de $N = 2n_x n_y$ soluções distintas, para placas de $n_x \times n_y$ pixels. É importante salientar que todos os mapeamentos indicados na Figura 54 produzem as mesmas grades, diferindo apenas na ordem com que as placas são inseridas.

Figura 58 – Formação de novas grades a partir do deslocamento da origem.



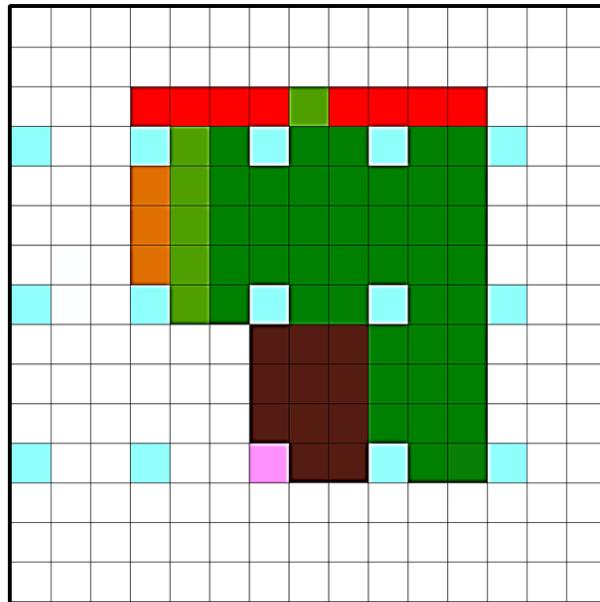
Fonte: Autoral.

Figura 59 – Grades idênticas com origens distintas, indicadas em rosa.



Fonte: Autoral.

Figura 60 – Número de resultados possíveis para formação do grid.



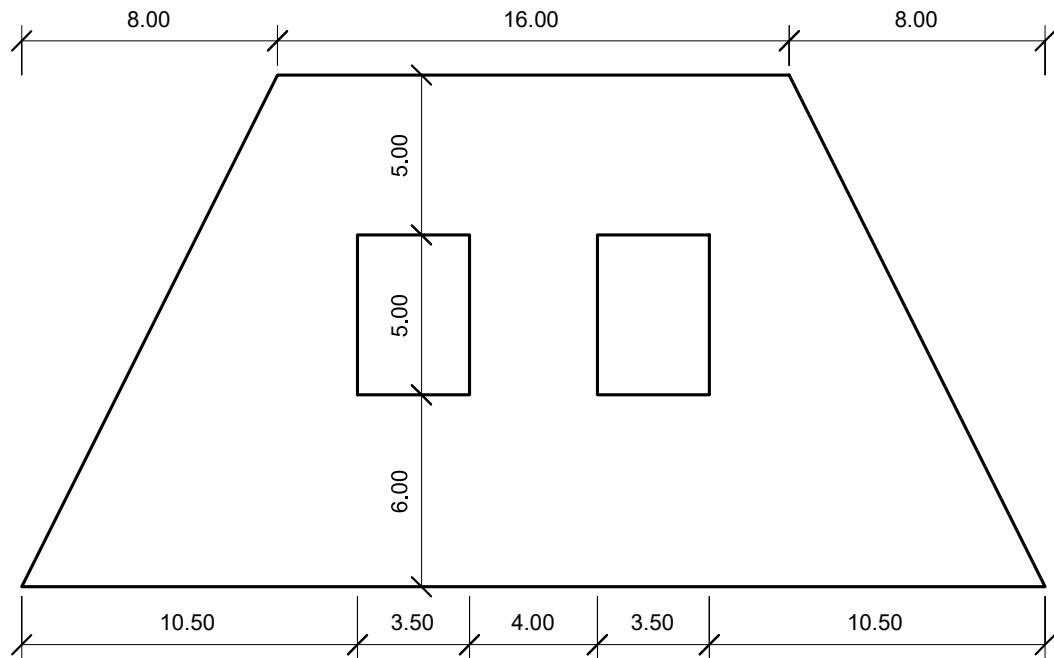
Fonte: Autoral.

3.2.3 Validação da heurística desenvolvida

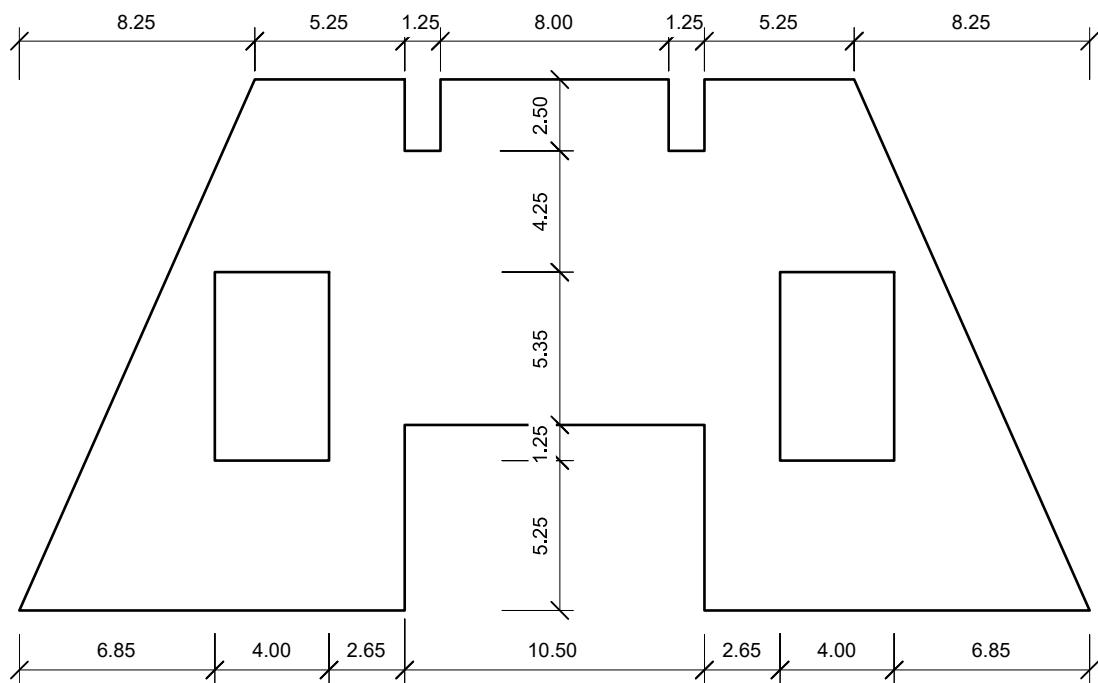
Com o intuito de verificar a eficácia das heurísticas de locação de placas propostas neste trabalho, realiza-se aqui uma comparação dos resultados obtidos nos Exemplos 3 e 4 do trabalho de Dür e Nowak (2008). Os modelos geométricos dos telhados e das placas usados nestes exemplos foram inferidos a partir de imagens contidas no referido artigo com o auxílio do Autodesk AutoCAD e do Autodesk Revit. As medidas estimadas estão apresentadas nas Figuras 61(a) e 61(b). A heurística de Dür e Nowak (2008) se aproxima à modalidade livre, e portanto, esta será a modalidade empregada.

Para o Exemplo 3, um telhado sem inclinação e trapezoidal, com dois furos internos, a heurística desenvolvida de Dür e Nowak (2008) aloca 22 placas, como mostrado na Figura 62(a). Com ancoragem preferencial no canto inferior-esquerdo e mapeamento vertical, a heurística desenvolvida neste trabalho também consegue alocar 22 placas, em um posicionamento semelhante ao de Dür e Nowak (2008), como mostra a Figura 62(b), onde os números indicam a ordem de inserção.

Figura 61 – Medidas estimadas para os Exemplos 3 e 4 de Dür e Nowak (2008) com o auxílio do Autodesk AutoCAD e do Autodesk Revit.



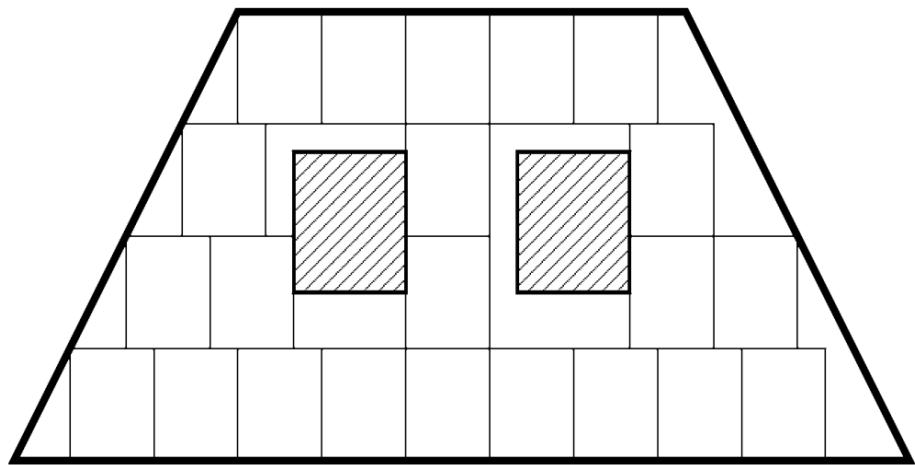
(a) Exemplo 3.



(b) Exemplo 4.

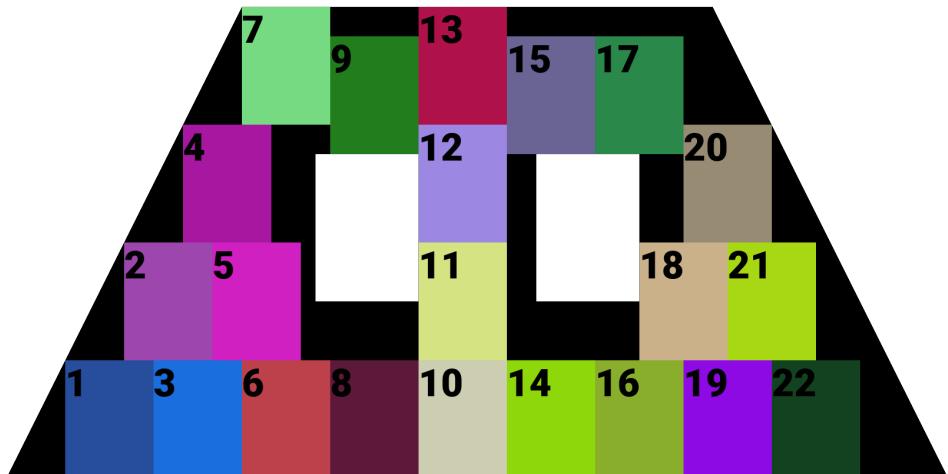
Fonte: Autoral.

Figura 62 – Resultados obtidos para o Exemplo 3 segundo Dür e Nowak (2008) e a heurística proposta neste trabalho.



(a) Heurística de Dür e Nowak (2008).

Fonte: Dür e Nowak (2008).



(b) Heurística proposta.

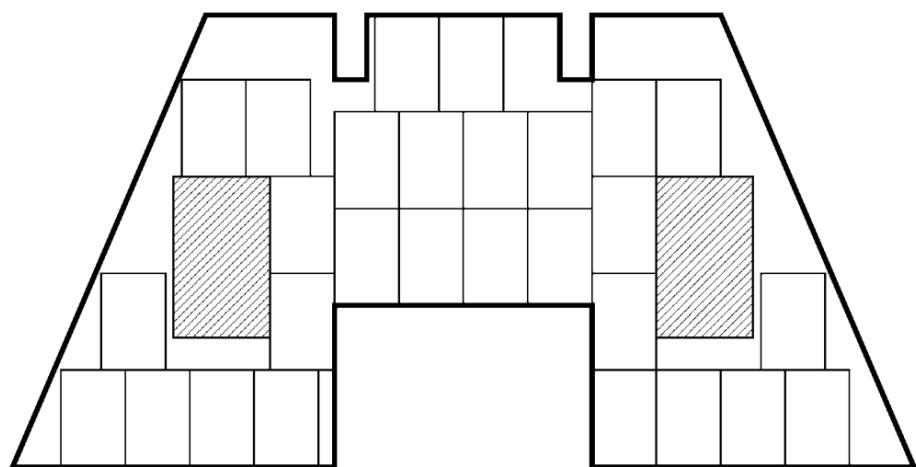
Fonte: Autoral.

A maior diferença entre as Figuras 62(a) e 62(b) é a alocação das placas 9, 15, 17 e 20. Como no mapeamento vertical da imagem a placa 17 é inserida antes da 20, a última acaba ficando deslocada. Já para as placas 9 e 15, o fato de não estarem alinhadas com as placas 7 e 13 pode ser resultado de um menor espaçamento da grade usada na heurística desenvolvida neste trabalho. Em uma malha com o espaçamento muito do maior que a utilizada na Figura 62(b), que possui um γ_p de 50, ou seja 50 pixels por unidade de medida, mover a placa um espaço para baixo já a faria entrar na abertura do telhado, impossibilitando sua inserção. Não obstante, não

fica claro em Dür e Nowak (2008) se alguma técnica de alinhamento ou de aderência (*snapping*) é usada.

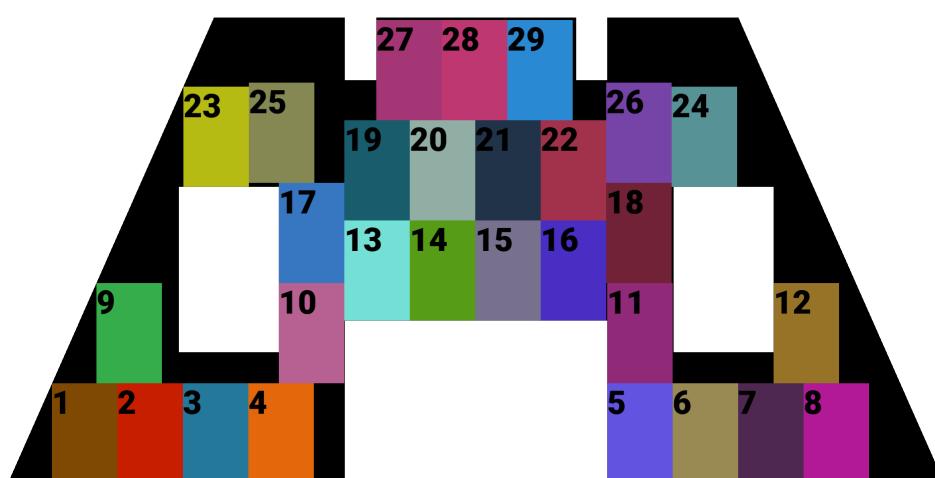
Para o Exemplo 4, um telhado sem inclinação, côncavo devido a reentrâncias produzidas no modelo, com dois furos internos, a heurística de Dür e Nowak (2008) consegue alocar 28 placas, como mostrado na Figura 63(a). A heurística desenvolvida neste trabalho, com ancoragem no canto inferior-esquerdo, porém com mapeamento horizontal, consegue posicionar 29 placas, uma a mais do que em Dür e Nowak (2008), conforme ilustrado na Figura 63(b).

Figura 63 – Resultados obtidos para o Exemplo 4 segundo Dür e Nowak (2008) e a heurística proposta neste trabalho.



(a) Heurística de Dür e Nowak (2008).

Fonte: Dür e Nowak (2008).



(b) Heurística proposta.

Fonte: Autoral.

As diferenças nos resultados para o Exemplo 4 são menores do que aquelas observadas para o Exemplo 3, parecendo ser apenas consequência de erros introduzidos nas medidas do telhado e das placas. A alocação da placa adicional, 29, reforça a teoria de que Dür e Nowak (2008) utilizam uma malha com espaçamento maior, visto que a placa 27, no resultado delas, poderia ser alocada um pouco mais para o lado esquerdo, encostando na reentrância do telhado, e provavelmente não o faz porque mover um espaço da malha à esquerda já a faria entrar nesta reentrância. A soma desta fina abertura à esquerda com a grande abertura à direita já poderia ser suficiente para a instalação da vigésima nona placa. Como a placa 27 é inserida encostada à reentrância na heurística proposta (Figura 63(b)), uma placa adicional consegue ser inserida. É possível verificar o mesmo espaçamento comparando a alocação da placa 12.

Do modo com que as placas 9, 12 e 27 foram alocadas por Dür e Nowak (2008) no Exemplo 4, a aproximadamente um terço da placa inferior na Figura 63(b), é provável que o tamanho horizontal da placa seja de 3 espaços, e por consequência das hipóteses adotadas por elas (Figura 39), um tamanho vertical de 5 espaços, ocupando um total de 15 espaços por placa. Com o γ_p de 50 do resultado dado pela heurística proposta aqui, uma placa de 3×4 unidades como a do Exemplo 4 teria 150 *pixels* por 200 *pixels*, ocupando uma área de 30.000 espaços (que aqui coincide com a quantidade de *pixels* cobertos por uma placa). O aumento do número de subdivisões, porém, resulta em um acréscimo considerável no tempo de execução. Enquanto que os resultados dos Exemplos 3 e 4 foram executados na casa dos milésimos por Dür e Nowak (2008), os mesmos exemplos levaram várias horas para serem executados pela heurística produzida neste trabalho. Embora mais barata computacionalmente, a modalidade alinhada não foi considerada na validação, por se afastar muito da ideia usada por Dür e Nowak (2008). A comparação entre os resultados está presente A Tabela 4 contém os tempos de execução de cada método aqui considerado, aplicados aos Exemplos 3 e 4.

Tabela 4 – Quantidade de placas alocadas e tempo de execução relativos às heurísticas de Dür e Nowak (2008) e deste trabalho, aplicadas aos Exemplos 3 e 4.

| Exemplo | Dür e Nowak (2008) | | Heurística proposta | |
|---------|--------------------|------------------------|---------------------|------------------------|
| | placas | tempo de execução (ms) | placas | tempo de execução (ms) |
| 3 | 22 | 70 | 22 | 33718025 |
| 4 | 28 | 93 | 29 | 40381080 |

Fonte: Autoral.

4 RESULTADOS

As heurísticas propostas foram avaliadas experimentalmente sobre modelos de edificações desenvolvidos no *Autodesk Revit 2021*. A execução do código produzido se dará em duas máquinas, a partir de agora denominadas Máquina 1 e Máquina 2, ambos computadores pessoais. A Máquina 1 é um computador de mesa equipado com:

- Processador AMD FX-6300 Black Edition, 6 núcleos físicos e 0 núcleos lógicos, com clock de 3,5 GHz e 8 MB de cache L3;
- 8 Gb de RAM do tipo DDR3 de 1600 MHz, distribuídos em 2 slots com 4 Gb cada;
- Armazenamento em HDD de 320 Gb, 5400 RPM;
- Sistema operacional Ubuntu 22.04.1.

Já a Máquina 2 é um notebook com:

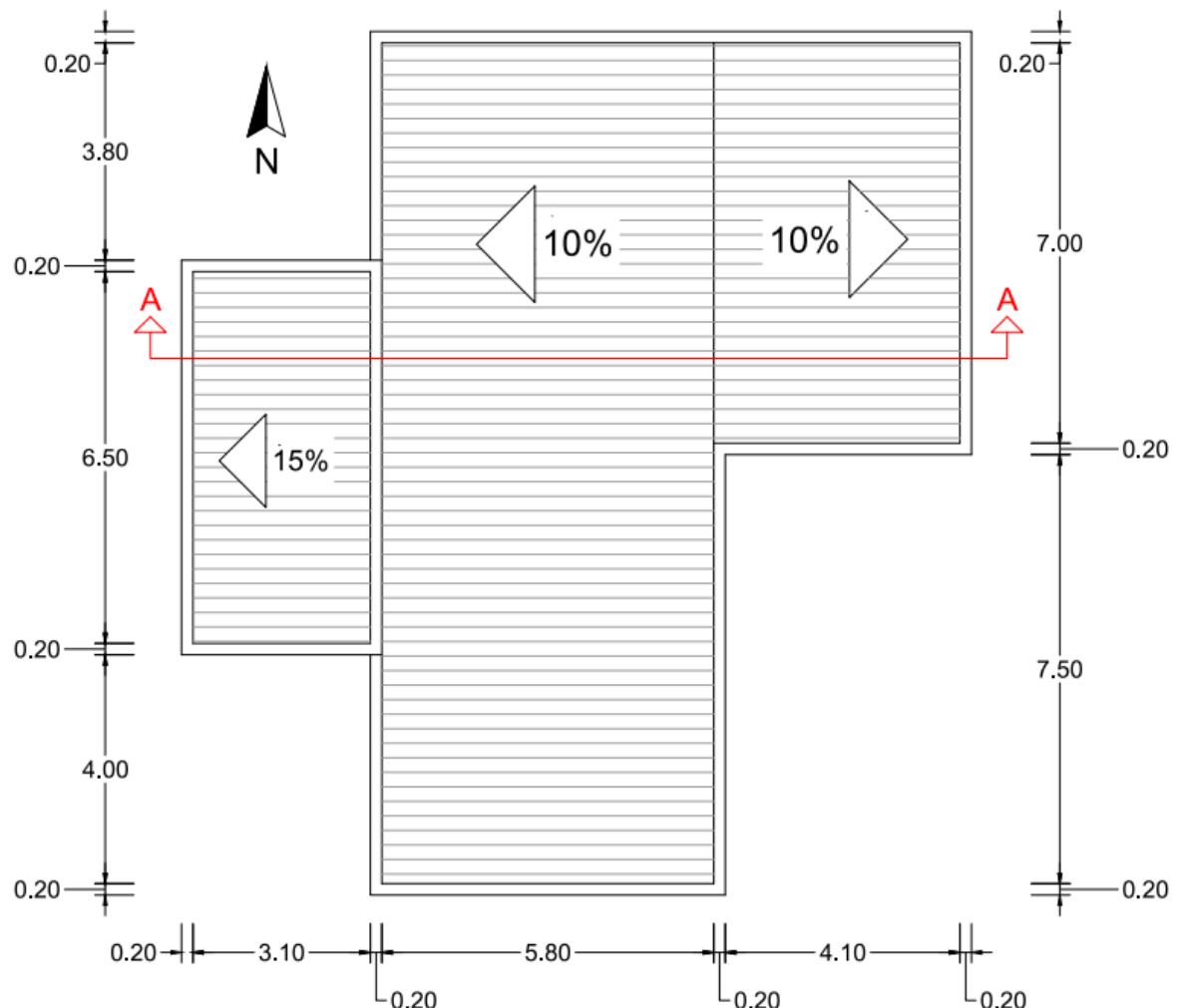
- Processador Intel Core i5-7300HQ, 4 núcleos físicos e 4 núcleos lógicos, com clock de 2,5 GHz e 6 MB de cache L3;
- 16 Gb de RAM do tipo DDR4 de 2666 MHz, em dois slots de 8 Gb cada;
- Armazenamento em HDD de 1 Tb, 5400 RPM;
- Sistema operacional Ubuntu 20.04.1.

Em todos os casos analisados, sempre será empregada a modalidade alinhada, tendo em vista que esta produz leiautes construtivamente mais realizáveis quando comparados às soluções obtidas pela modalidade livre.

4.1 Exemplo 1

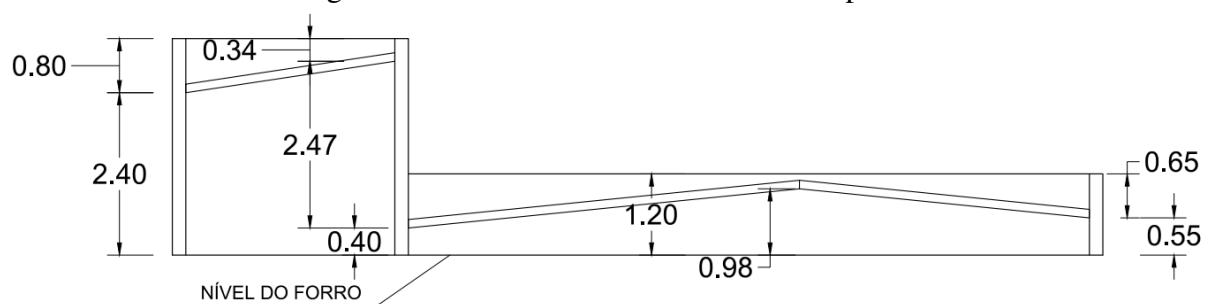
O primeiro exemplo é um modelo simples, que conta apenas com paredes, um telhado de duas águas, onde serão inseridos módulos fotovoltaicos, e um telhado de uma água para a caixa d'água, onde não serão inseridos módulos. A intenção deste cômodo mais alto para a caixa d'água é provocar o surgimento de uma sombra mais pronunciada no lado esquerdo do telhado. As medidas em metros do modelo tridimensional são mostradas nas Figuras 64 e 65. Uma vista 3D do modelo, que possui 216 faces triangulares, é apresentada na Figura 66. As imagens foram produzidas com $\gamma_p = 50$ e, portanto, um *pixel* equivale a 2 centímetros, já que a unidade do modelo é o metro.

Figura 64 – Planta do modelo do Exemplo 1.



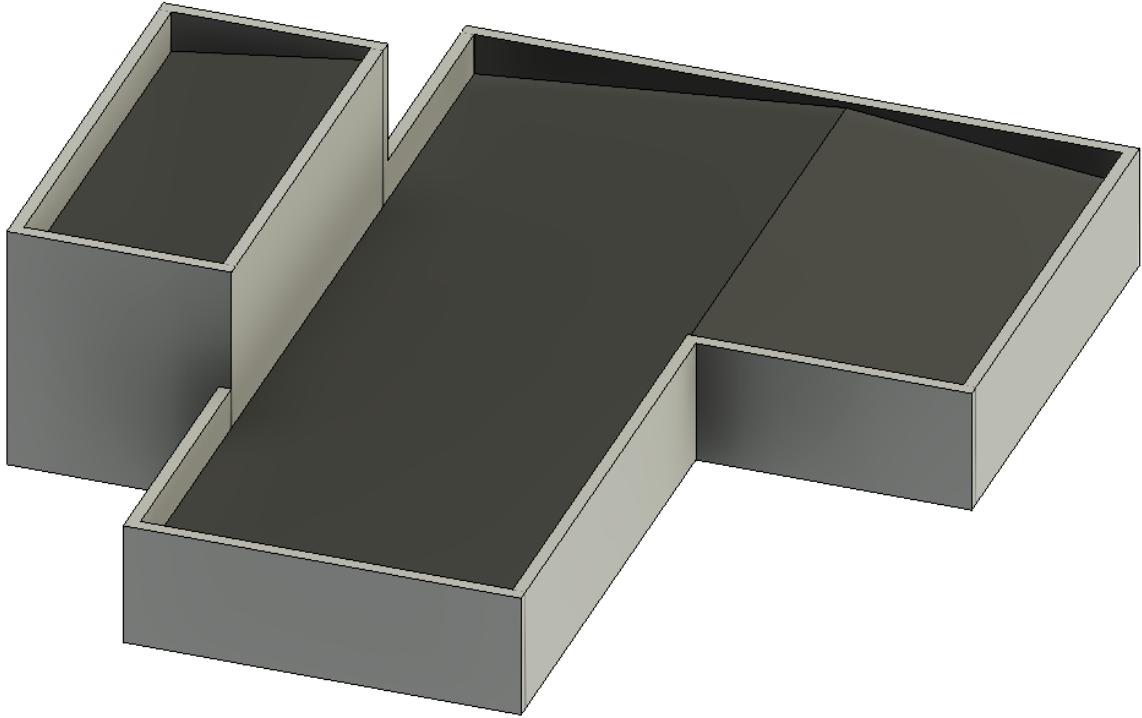
Fonte: Autoral.

Figura 65 – Corte AA do modelo do Exemplo 1.



Fonte: Autoral.

Figura 66 – Vista 3D do modelo do Exemplo 1.



Fonte: Autoral.

O plano de trabalho do modelo se situa alinhado ao norte, e portanto $\theta_{A_{off}} = 0$. Para a produção do mapa de sombreamento, foram utilizadas as 8 horas mais iluminadas dos dias 20 de março de 2022 (equinócio de outono), 21 de junho de 2022 (solstício de inverno), 22 de setembro de 2022 (equinócio de primavera) e 21 de dezembro de 2022 (solstício de verão), de modo a captar o movimento das áreas sombreadas ao longo de todo o ano. Os dados de azimute e elevação solar para estas datas estão na Tabela 5.

Como o telhado possui duas águas, é necessário processar as águas esquerda e direita separadamente, como foi discutido no Capítulo 3. Para a água da esquerda, a construção do mapa de sombreamento requereu um total de 18 horas e 56 minutos de tempo de CPU, mesmo explorando o multiprocessamento disponível na Máquina 1. O mapa foi construído em 4 etapas e os resultados de cada etapa foram acumulados ao final.

Tabela 5 – Informações de elevação e azimute solar, em graus, para Juazeiro do Norte-CE

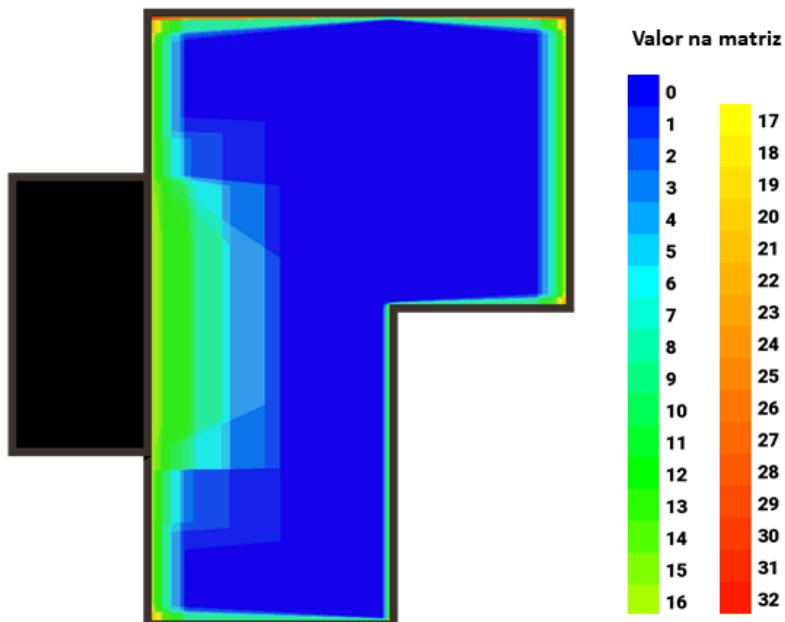
| Hora | 20 mar 2022 | | 21 jun 2022 | | 22 set 2022 | | 21 dez 2022 | |
|------|---------------|------------|---------------|------------|---------------|------------|---------------|------------|
| | θ_{el} | θ_A | θ_{el} | θ_A | θ_{el} | θ_A | θ_{el} | θ_A |
| 9h | 33,52 | 85,28 | 28,37 | 58,40 | 37,11 | 84,22 | 35,95 | 113,79 |
| 10h | 48,31 | 81,91 | 40,51 | 50,54 | 51,87 | 80,37 | 49,37 | 117,91 |
| 11h | 62,92 | 75,75 | 50,94 | 37,60 | 66,37 | 72,67 | 61,95 | 127,95 |
| 12h | 76,73 | 57,64 | 57,89 | 17,05 | 79,49 | 45,63 | 71,69 | 153,31 |
| 13h | 81,84 | 331,98 | 58,93 | 350,69 | 79,49 | 314,24 | 72,74 | 199,34 |
| 14h | 69,88 | 290,26 | 53,53 | 327,76 | 66,37 | 287,20 | 64,02 | 229,06 |
| 15h | 55,49 | 280,66 | 43,92 | 312,73 | 51,87 | 279,49 | 51,73 | 240,87 |
| 16h | 40,77 | 276,33 | 32,18 | 303,60 | 37,11 | 275,64 | 38,40 | 245,72 |

Coordenadas: 7° 12' 57,176"S; 39° 18' 44,088"W

Fonte: SunEarthTools (2022).

No caso da água da direita, foram necessárias 7 horas e 36 minutos, também na Máquina 1. O tempo de execução foi menor que no lado esquerdo devido a dois fatores: a região viável é menor e o código implementado realiza um pré-mapeamento que pode ser reutilizado, como será explicado mais adiante. O resultado combinado dos mapas de sombreamento pode ser visto na Figura 67.

Figura 67 – Mapa de sombreamento combinado das águas esquerda e direita do telhado do Exemplo 1.



Fonte: Autoral.

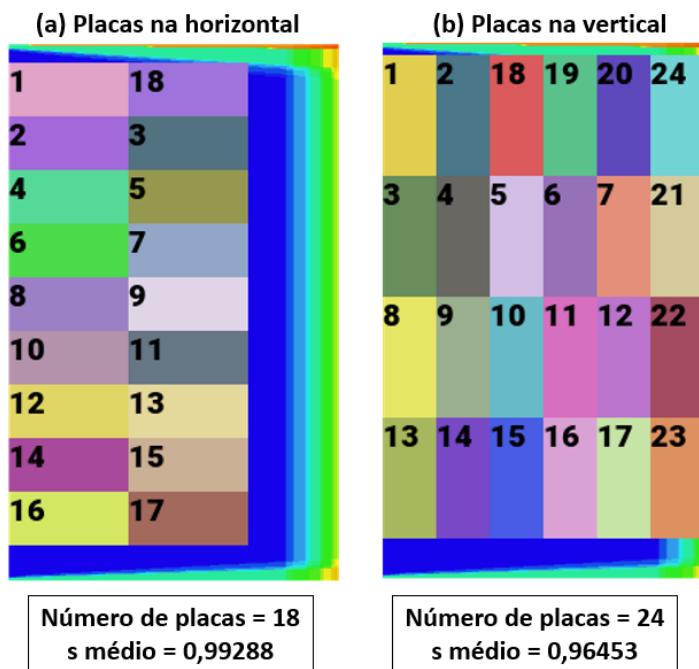
As placas que se deseja inserir têm 1,56 por 0,70 metros e 3 centímetros de espessura. As medidas corrigidas pela inclinação são $dim_{proj_1} = 1,56 \cdot \cos(0,1 rad) = 1,5522m$ e $dim_{proj_2} =$

$0,70 \cdot \cos(0,1 \text{ rad}) = 0,6965 \text{ m}$. Estes cálculos são realizados automaticamente pelo programa. Como a inclinação é pequena e a precisão da imagem é de 2 centímetros, a placa projetada na imagem continuaria tendo 1,56 m por 0,70 m, após arredondamento. Estes valores representam 78 pixels e 35 pixels, respectivamente. É preciso lembrar que apenas uma das medidas está inclinada em cada caso de alocação, a depender da orientação da placa. Particularmente aqui, a pequena inclinação não altera significativamente nenhuma das medidas após a correção.

Supondo que a edificação necessite de 60 placas, é razoável iniciar a inserção pela água do lado direito, já que ela está menos sombreada, e o restante das placas serão posicionadas na água esquerda. Para ambas, foram testadas as duas orientações de placa, vertical e horizontal.

Antes de procurar todos os casos possíveis e definir o melhor, foi executado um teste preliminar, priorizando o canto superior-esquerdo, com mapeamento horizontal, apenas para verificar a quantidade máxima de placas que a água direita pode comportar, considerando ambas as orientações. Para isso, foi fixado um número elevado de placas n_p , mais precisamente, $n_p = 50$, impossível de serem locadas nesta referida água. O resultado é o apresentado em na Figura 68(a), requerendo 5 minutos e 29 segundos, na Máquina 2. A maior parte deste tempo é gasto durante a busca pelo primeiro pixel onde é possível inserir uma placa não sombreada, a origem da grade. Após isso, a inserção das demais placas levou 4 segundos. Foram locadas 18 placas, com coeficiente de sombreamento médio de 0,99288.

Figura 68 – Teste na água direita do Exemplo 1 para as duas orientações da placa.



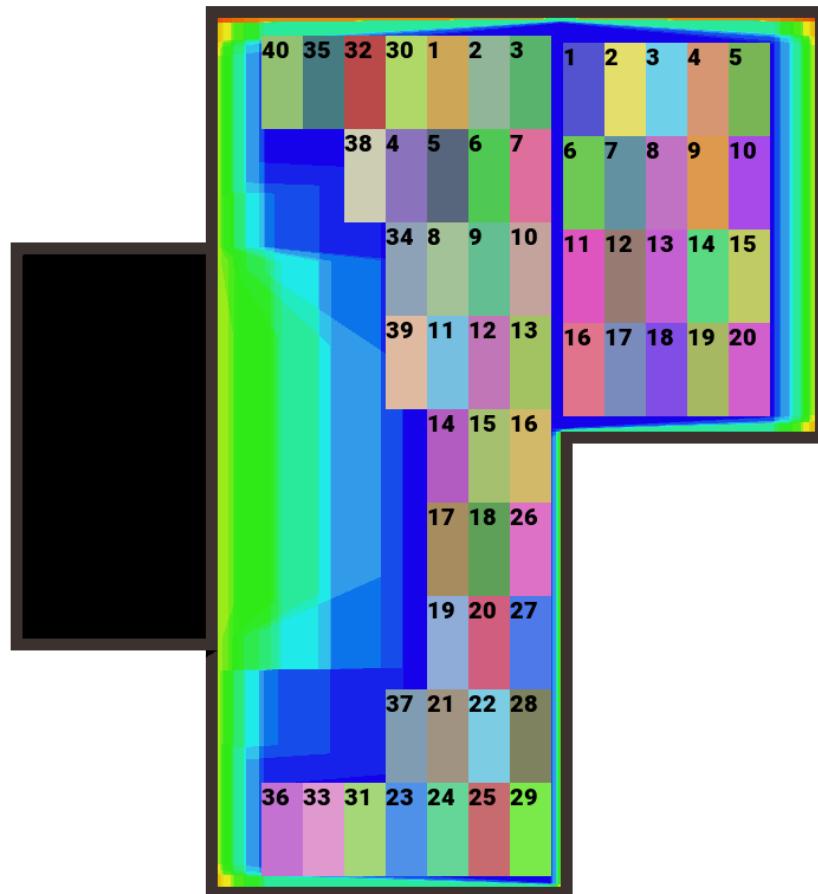
Fonte: Autoral.

Para as mesmas condições anteriores, porém considerando as placas na posição vertical, foi obtido o resultado da Figura 68(b), com um tempo de execução de 6 minutos e 13 segundos na Máquina 2. Neste caso, foram inseridas 24 placas, resultando em um coeficiente de sombreamento médio de 0,96453.

Analizando ambos os resultados, é importante observar que o arranjo vertical utiliza melhor o espaço, embora corresponda ao menor coeficiente de sombreamento médio. Isto acontece por causa das placas 21 a 24, localizadas em áreas muito mais sombreadas em relação às demais. Retirando-se estas placas, obtém-se um total de 20 unidades, ainda maior que a quantidade obtida no arranjo horizontal. Dessa forma, decidiu-se por alocar as 60 placas todas com orientação vertical, distribuindo 20 delas para a água direita e 40 para o lado esquerdo.

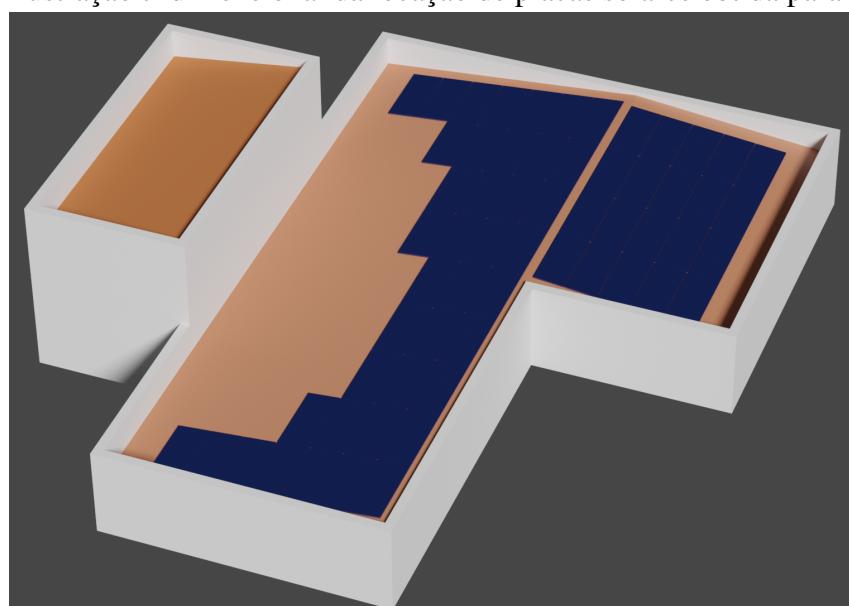
Utilizando os valores de n_p obtidos no teste preliminar para cada água, o programa foi executado para calcular, dentre todos os arranjos possíveis, qual aquele com maior coeficiente de sombreamento médio. Utilizando a Máquina 2, foram necessários 24 minutos e 34 segundos para o processamento da água direita e 34 minutos e 37 segundos para a água da esquerda. Os coeficientes de sombreamento médio das águas direita e esquerda foram iguais a 1 e 0,9968, respectivamente. Os resultados combinados de ambas as águas está apresentado na Figura 69 e um modelo tridimensional ilustrando as placas solares instaladas está na Figura 70.

Figura 69 – Mapa de locação combinado das águas esquerda e direita do telhado do Exemplo 1.



Fonte: Autoral.

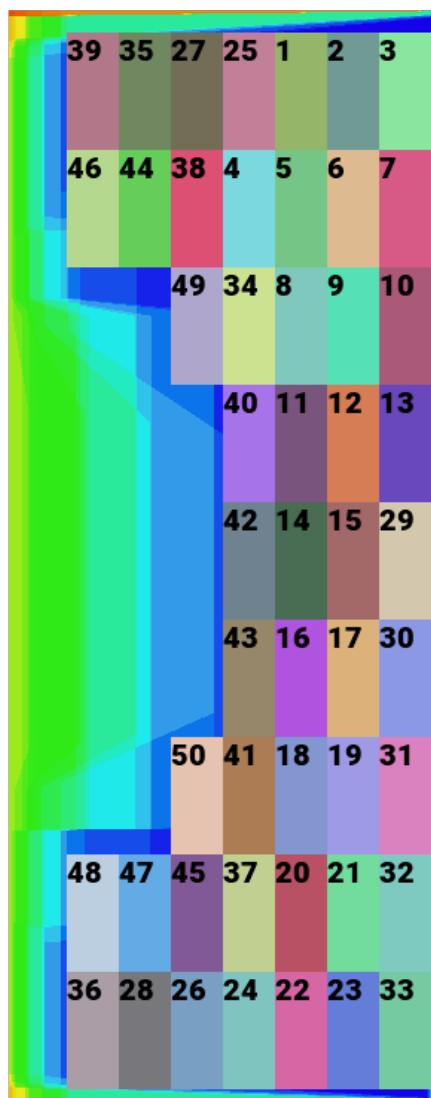
Figura 70 – Ilustração tridimensional da locação de placas solares obtida para o Exemplo 1.



Fonte: Autoral.

Algo interessante de se fazer na água esquerda, que ainda comportaria mais placas solares, e possui um elevado coeficiente de sombreamento médio, é analisar a capacidade de expansão do sistema calculado. Tomando agora $n_p = 50$ para a água esquerda, obteve-se o resultado da Figura 71, calculado em 36 minutos e 23 segundos, com um coeficiente de sombreamento médio de 0,9882. Como a numeração das placas presente no mapa de locação corresponde à ordem decrescente do coeficiente de sombreamento, fica definida então a melhor localização para a instalação de novas placas solares no futuro.

Figura 71 – Mapa de locação de placas para uma eventual expansão do sistema do Exemplo 1.



Fonte: Autoral.

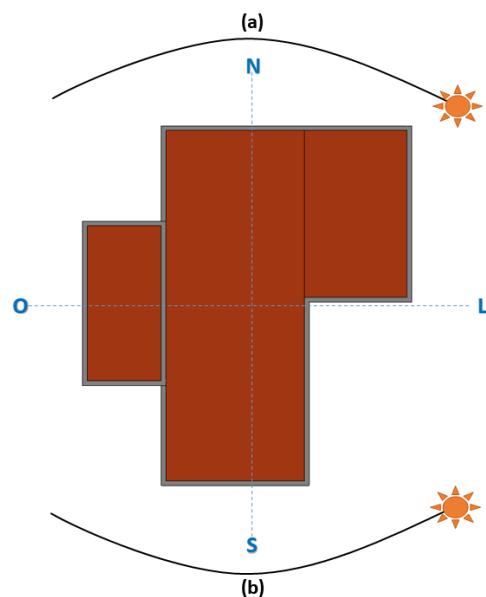
A análise da Figura 71 deve ser feita da seguinte forma: o sistema é inicialmente executado com as placas de 1 a 40. Caso, em um momento futuro, se queira expandir o sistema, adicionando mais 3 placas, por exemplo, as placas menos sombreadas que deveriam ser

adicionadas são as de número 41, 42 e 43. Caso, em outro momento, se queira adicionar mais 2 placas, seriam adicionadas as placas 44 e 45, e assim sucessivamente.

Observe que os mapas de sombreamento e de locação das placas solares no lado direito do telhado seriam facilmente determinados a partir de uma análise *in loco*. De fato, a sombra produzida neste lado é mais previsível, e se concentra nas bordas, devido à platibanda. Por isso, o arranjo das placas tomou uma forma simples, retangular. Já no caso da água do lado esquerdo, que recebe interferência do compartimento da caixa d'água, a situação é mais complexa. Um dia de visita à edificação certamente seria insuficiente para determinar toda a mancha de sombreamento aqui calculada.

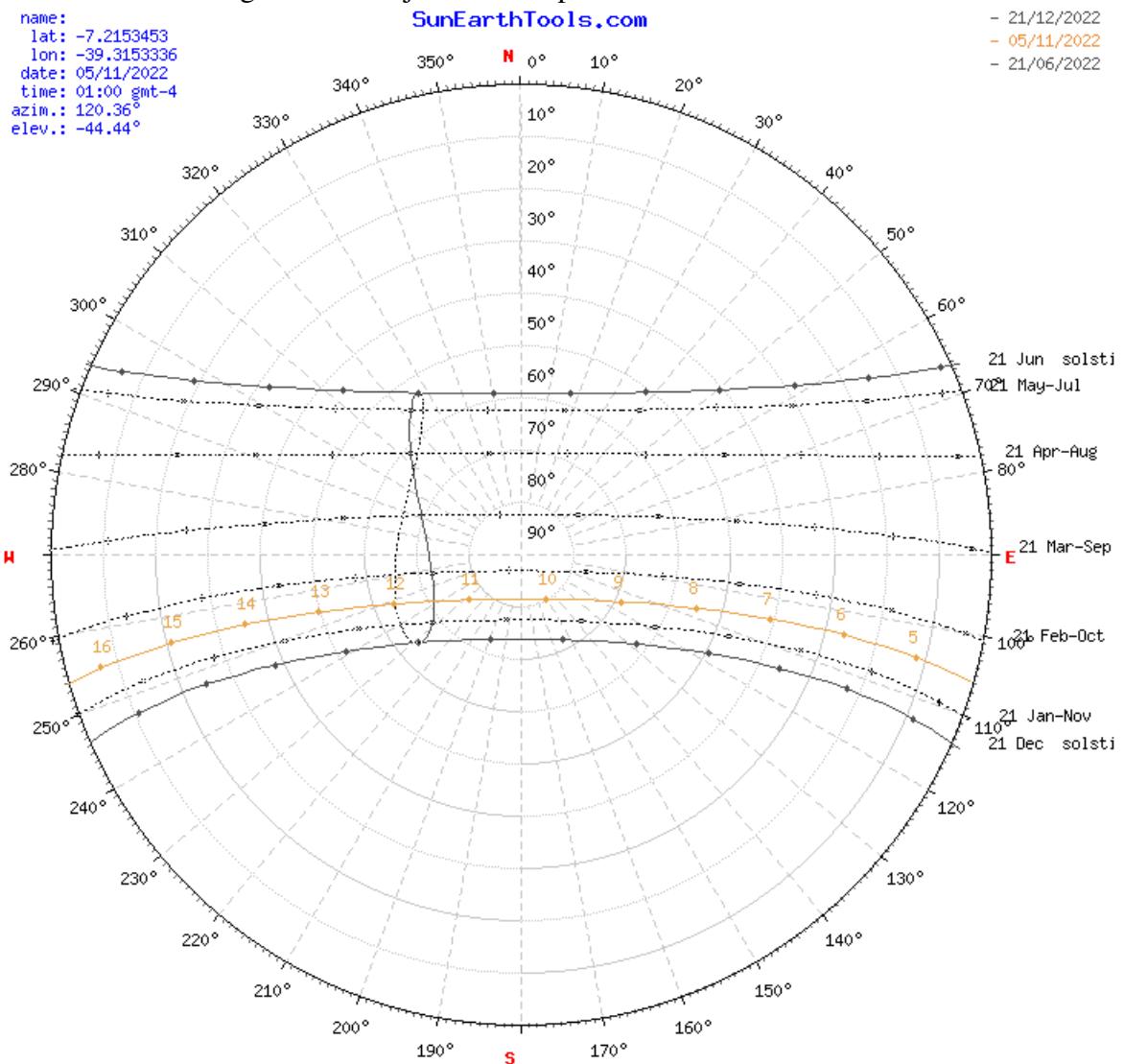
No caso particular da localidade em estudo, a cidade de Juazeiro do Norte/CE, ainda ocorre o seguinte fenômeno na trajetória solar. No solstício de inverno e nos dois equinócios, a trajetória solar é semelhante à da Figura 72(a), ou seja, o sol vai de leste a oeste apontando para o norte. Já no solstício de verão, o sol perfaz uma trajetória semelhante à da Figura 72(b), ou seja, o sol vai de leste a oeste apontando para o sul. Como Juazeiro do Norte está relativamente próxima ao equador, a diferença de dias onde o sol faz a trajetória (a) para os dias nos quais o sol faz a trajetória em (b) é pequena, ou seja, o equinócio ocorre muito perto do eixo leste-oeste, como mostrado na Figura 73.

Figura 72 – Representação da trajetória solar no Exemplo 1.



Fonte: Autoral.

Figura 73 – Trajetória solar para Juazeiro do Norte/CE.



Fonte: SunEarthTools (2022).

Desta forma, a observação do empreendimento no solstício de verão ou em dias próximos a este evento iria apontar sombras apenas nas bordas inferiores e laterais do telhado, e seria representativa para um menor número de dias do ano. Já a observação no solstício de inverno iria apontar sombras apenas nas bordas superiores e laterais do telhado, e representaria um número maior de dias do ano, mas não por uma grande margem. Ou seja, para uma análise mais completa, seria necessária a observação em pelo menos dois dias do ano, com uma diferença aproximada de 6 meses entre eles. Este tempo atrasaria a execução de um possível projeto, e é por isso que ferramentas computacionais, como as desenvolvidas neste trabalho, são utilizadas para este fim. Além disso, quando o empreendimento está em fase de projeto, ou seja, ainda não

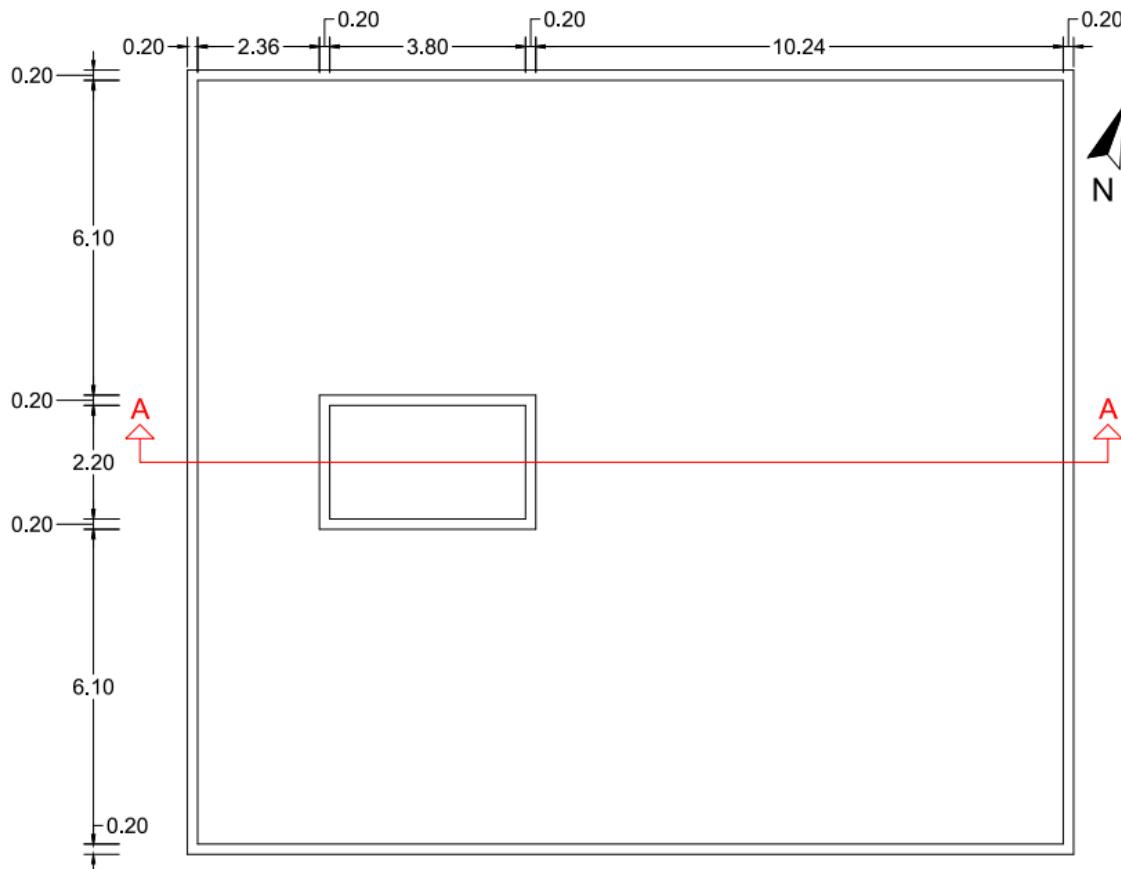
foi construído, é impossível fazer esse tipo de observação, sendo essencial o uso de ferramentas computacionais.

Embora o tempo de execução para a construção do mapa de sombreamento seja razoavelmente elevado, ele ainda é bem menor que o tempo necessário para se fazer as mesmas observações *in loco*. Além disso, utilizando-se de dados como os fornecidos pelo SunEarthTools (2022), é possível analisar qualquer dia do ano, no passado ou futuro.

4.2 Exemplo 2

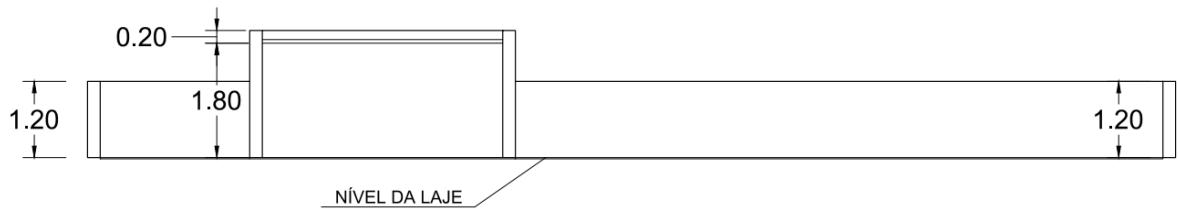
O segundo exemplo analisado serve para ilustrar peculiaridades referentes a edificações com coberturas planas, com inclinação nula. O modelo utilizado está detalhado nas Figuras 74, 75 e 76. Ele consiste de 172 faces triangulares. Trata-se de um terraço, onde serão inseridos os módulos, com uma casa de máquinas. Como a laje possui inclinação nula, os módulos serão inseridos com uma angulação arbitrária de $\sigma = 20^\circ$, faceando o norte (eixo y), na orientação horizontal.

Figura 74 – Planta do modelo do Exemplo 2.



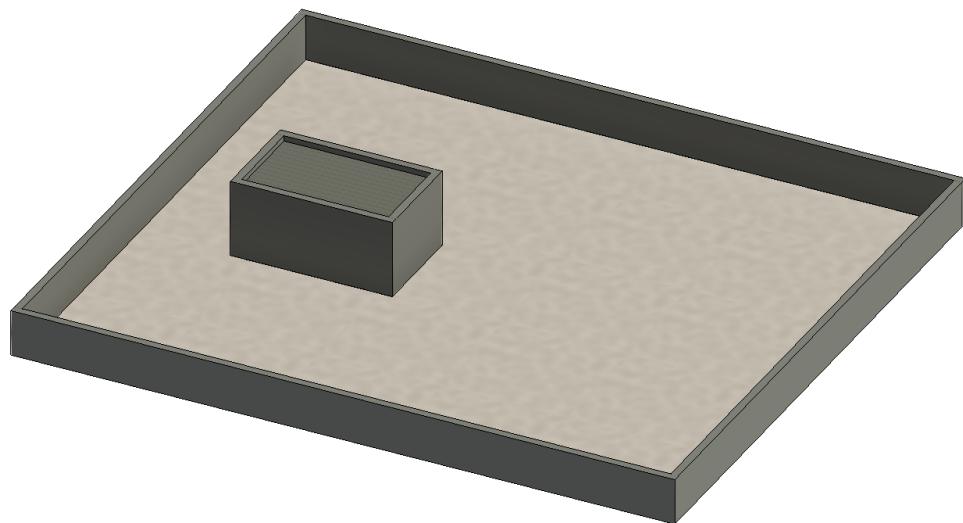
Fonte: Autoral.

Figura 75 – Corte AA do modelo do Exemplo 2.



Fonte: Autoral.

Figura 76 – Vista 3D do modelo do Exemplo 2.

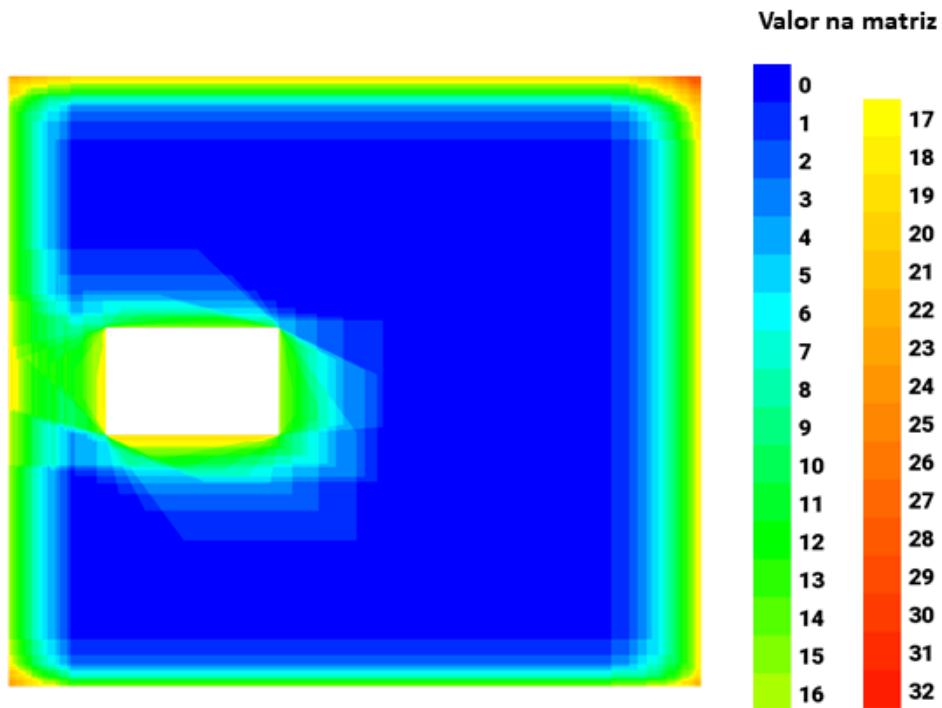


Fonte: Autoral.

Os dias de análise e a localidade serão os mesmos do Exemplo 1, porém com um ângulo de desvio do norte em relação ao plano de trabalho $\theta_{A_{off}} = 20^\circ$. O mapa de sombreamento produzido é mostrado na Figura 77, calculado após 7 horas e 52 minutos de tempo de CPU na Máquina 1.

Como a coberta é inteiramente plana, ou seja, o valor de z é igual em todos os pontos da região viável, não houve necessidade de se decompor o modelo para gerar múltiplos mapas de sombreamento. Utilizando o mesmo modelo de placa solar do Exemplo 1, há espaço suficiente para a alocação de 100 placas. Para a orientação horizontal, a correção da medida que será efetivamente inclinada, que é a menor medida da face da placa, é dada por: $dim_{proj_2} = 0,70 \cdot \cos(20^\circ) = 0,65778\text{ m}$.

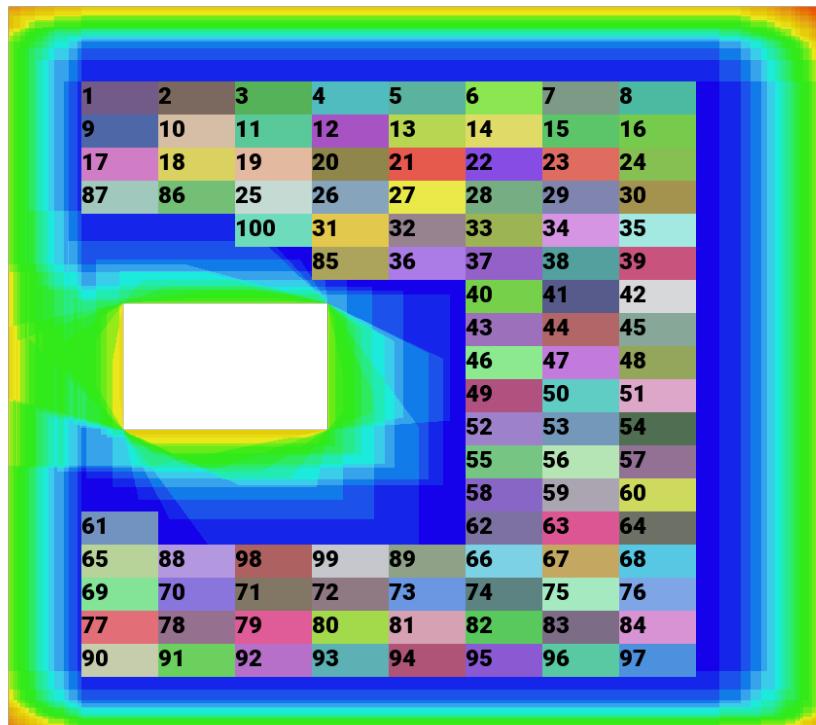
Figura 77 – Mapa de sombreamento da região viável do Exemplo 2.



Fonte: Autoral.

A precisão dos pixels deste modelo também é de 2 centímetros, isto é, $\gamma_p = 50$, assim como no Exemplo 1. Com isso, o arredondamento das dimensões resulta em placas solares de 1,56 m por 0,66 m, o que equivale a 78 por 33 pixels. As placas foram inseridas priorizando o canto superior-esquerdo, com mapeamento horizontal, sem avaliar todas as soluções possíveis com relação ao coeficiente de sombreamento, ou seja, a primeira placa sem sombreamento definiu a origem da grade de posições viáveis. O resultado obtido está exibido na Figura 78, o qual foi calculado em 5 minutos e 50 segundos na Máquina 2, apresentando coeficiente de sombreamento médio de 0,996.

Figura 78 – Resultado da alocação de placas do Exemplo 2.

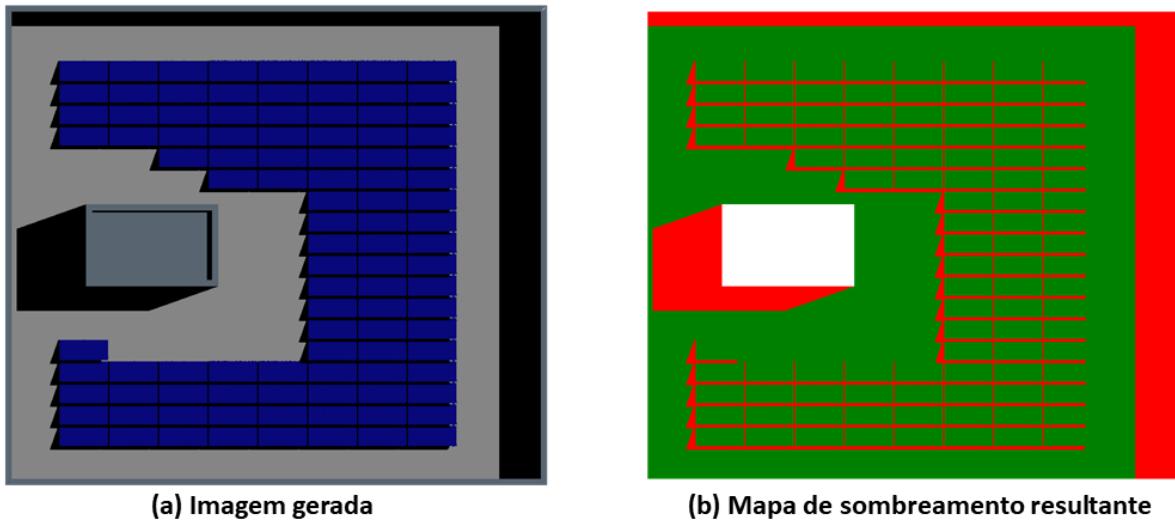


Fonte: Autoral.

Renderizando o modelo tridimensional com as placas solares locadas da Figura 78 usando traçado de raios, fixando as 10 horas da manhã do dia 21 de junho de 2022 (Tabela 5), obtemos a Figura 79(a) e o mapa de sombreamento na Figura 79(b), com verde indicando zonas iluminadas e vermelho indicando zonas sombreadas. Para tanto, foi necessário um total de 4 horas e 57 minutos na Máquina 1. O tempo muito maior para gerar apenas uma imagem ocorre porque a inserção das placas no modelo tridimensional adiciona 1200 triângulos à cena, já que cada placa solar é um paralelepípedo de 12 faces triangulares. Para gerar apenas uma imagem, o programa não se beneficia do pré-mapeamento de objetos.

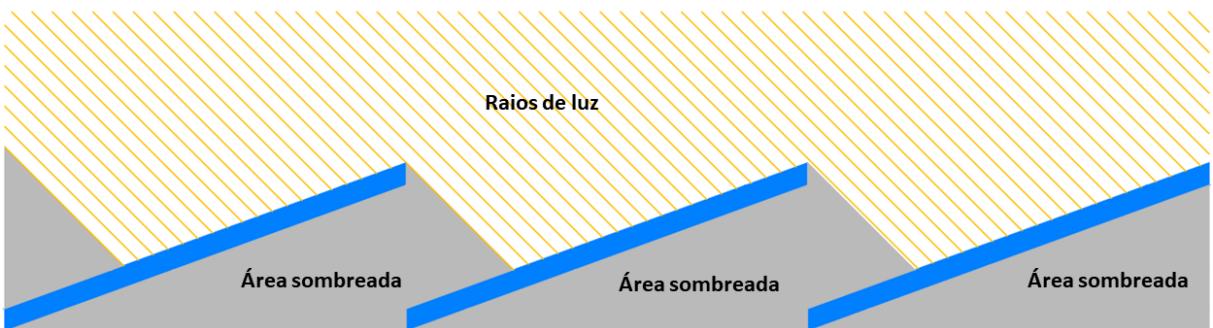
O que se pode perceber pela análise da Figura 79 é que as placas solares agora são agentes de sombreamento, e por estarem justapostas, acabam por sombrear as placas imediatamente atrás delas, como mostra a Figura 80. Isso é algo comum em sistemas inseridos em áreas planas, e para se mitigar este problema, basta que se aplique um espaçamento entre as placas. Na modalidade alinhada, a implementação deste espaçamento é simples, bastando que se produza uma grade com um espaçamento maior na direção em que a placa está inclinada. Já na modalidade livre, a implementação não é tão simples, já que o programa precisa lidar com a verificação da sobreposição entre placas.

Figura 79 – Imagens geradas para a primeira alocação do Exemplo 2, considerando as 10h do dia 21/06/2022.



Fonte: Autoral.

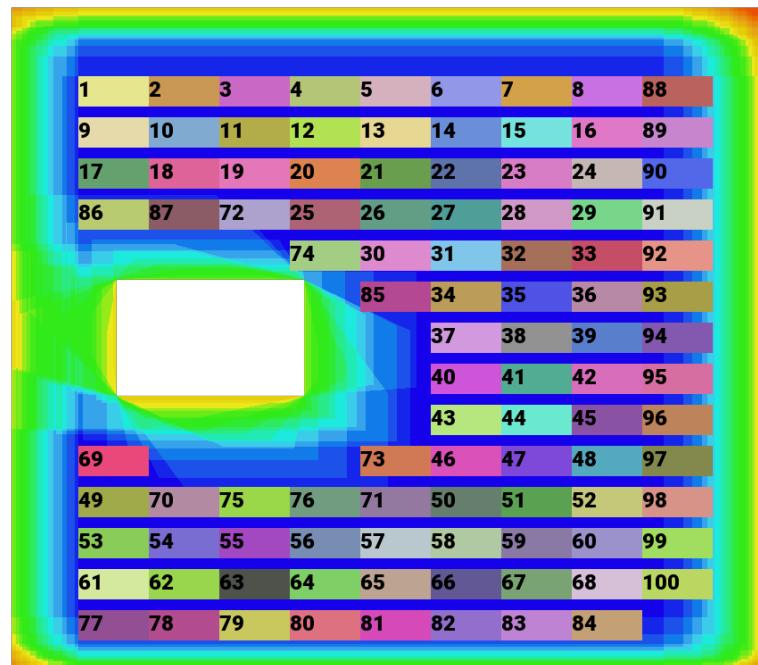
Figura 80 – Sombreamento devido a oclusões ocasionadas por placas próximas umas das outras.



Fonte: Autoral.

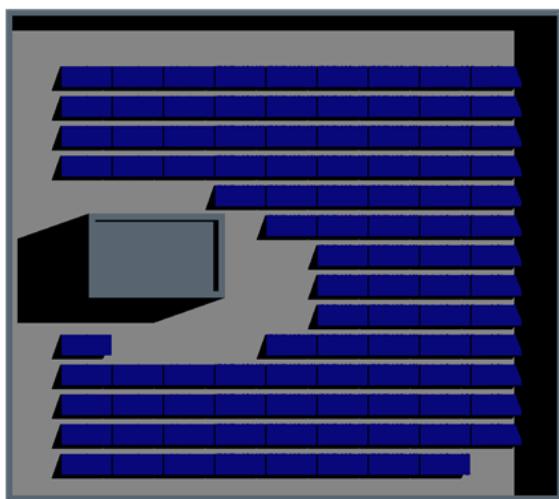
Com efeito, na modalidade alinhada, basta a adição de uma variável “afastamento”, que adicionará um valor em *pixels* à grade, no sentido da inclinação da placa, que neste exemplo é o eixo *y*. O valor de afastamento adotado neste exemplo foi de 24 centímetros, equivalente a 12 pixels. As 100 placas foram alocadas como mostra a Figura 81. O tempo de execução foi de 5 minutos e 43 segundos na Máquina 2, resultando em um coeficiente de sombreamento médio de 0,9871, sem a verificação de todos os resultados possíveis. O modelo tridimensional do mapa de locação renderizado via traçado de raios está na Figura 82. Como se pode perceber também com a Figura 83, o espaçamento de 24 centímetros foi suficiente para evitar o sombreamento entre placas, no horário considerado.

Figura 81 – Resultado da alocação de placas do Exemplo 2, com espaçamento adicional de 12 pixels.

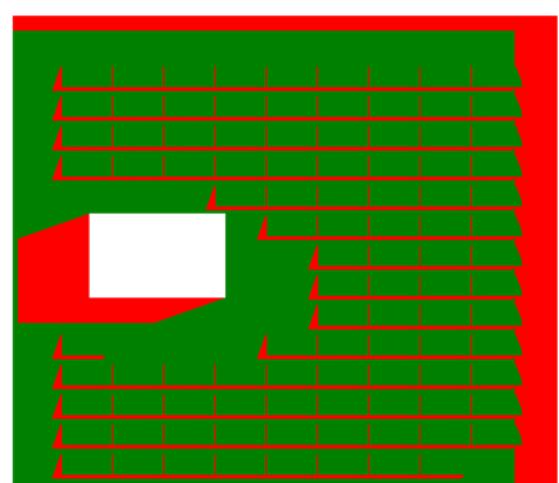


Fonte: Autoral.

Figura 82 – Imagens geradas da alocação com espaço adicional para o Exemplo 2, para as 10h do dia 21/06/2022.



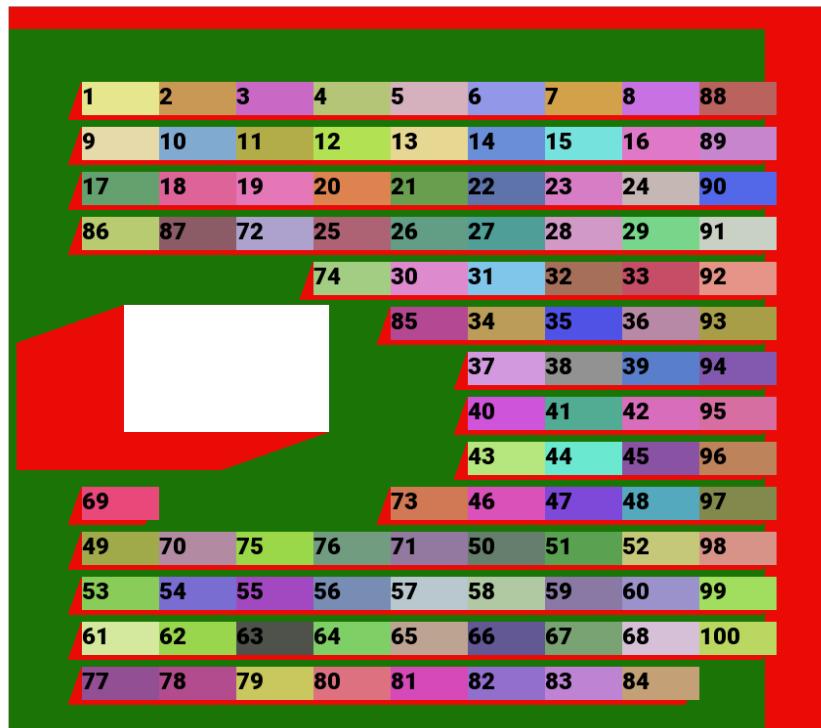
(a) Imagem gerada



(b) Mapa de sombreamento resultante

Fonte: Autoral.

Figura 83 – Sobreposição das placas locadas ao mapa de sombreamento da alocação do Exemplo 2 com espaçamento adicional.



Fonte: Autoral.

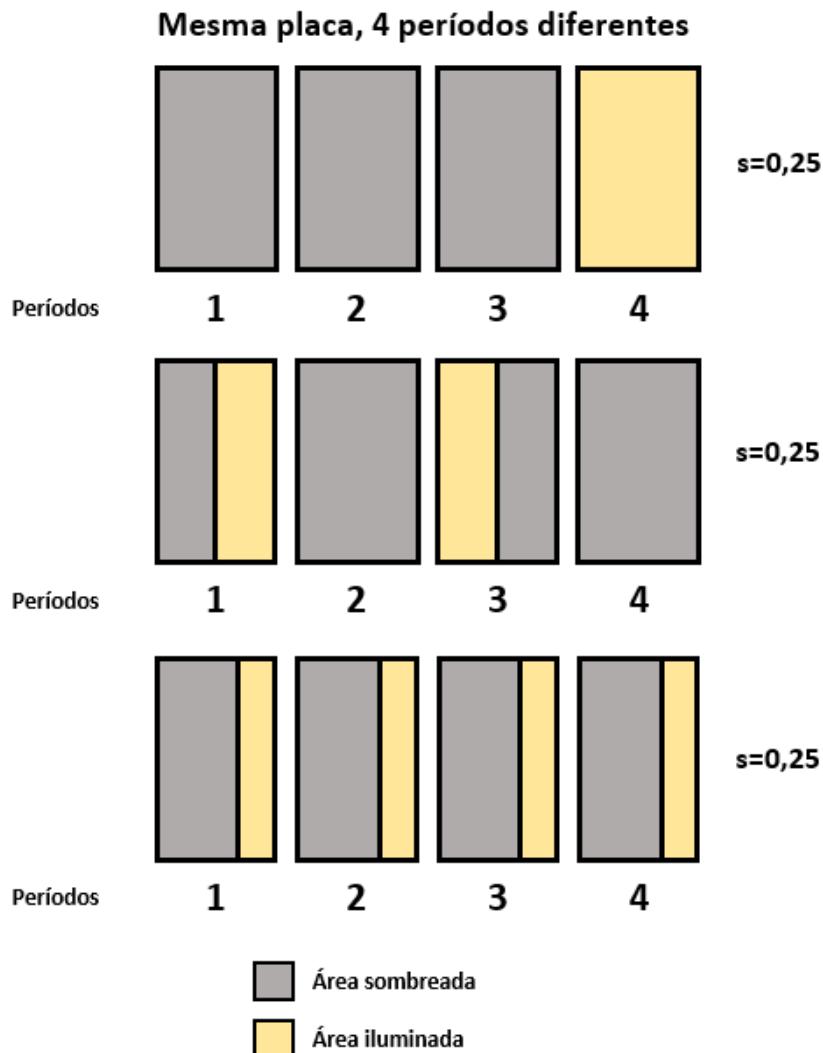
Os resultados obtidos no Exemplo 2 ilustram a problemática do afastamento inter-placas, um dos vários fatores que poderiam ser analisados mais criteriosamente aqui. O sentido para o qual a placa deve estar voltada deve depender da quantidade de dias que o sol se voltará para a face delas. Caso possível, é indicado que elas troquem de sentido ao longo do ano. A inclinação adotada para as placas geralmente é função da latitude do local e este ângulo ideal muda diariamente. Muitos sistemas mais complexos de produção de energia solar utilizam molduras que se voltam automaticamente para o sol ao longo do dia ou que pelo menos permitem um ajuste manual que pode ser feito com uma certa periodicidade. Além disso, caso o posicionamento das placas seja fixo, como os apresentados neste trabalho, é importante a consideração de mais de um dia e horário na definição do espaçamento das placas.

4.3 Sobre o coeficiente de sombreamento

O uso do coeficiente de sombreamento, que é de certa forma uma média dos valores da matriz do mapa de sombreamento, é bastante útil para determinar quantitativamente as áreas sombreadas, mas é um valor que perde um pouco do significado do mapa de sombreamento. Num exemplo onde temos 4 períodos representativos do ano todo, um valor de 3 em um dos índices

do mapa de sombreamento indica que aquela área representada pelo *pixel* está sombreada em 75% das horas de produção do sistema. Já um coeficiente de sombreamento de 0,25, equivalente a $1 - 0,75$, não indica necessariamente que o módulo está sombreado em 75% das horas de produção, mas sim que o somatório das áreas de superfície sombreada ao longo dos períodos equivalem a 75% das áreas de superfície total ao longo dos períodos, como mostra a Figura 84.

Figura 84 – Exemplo de módulos com o mesmo coeficiente de sombreamento em períodos diferentes.



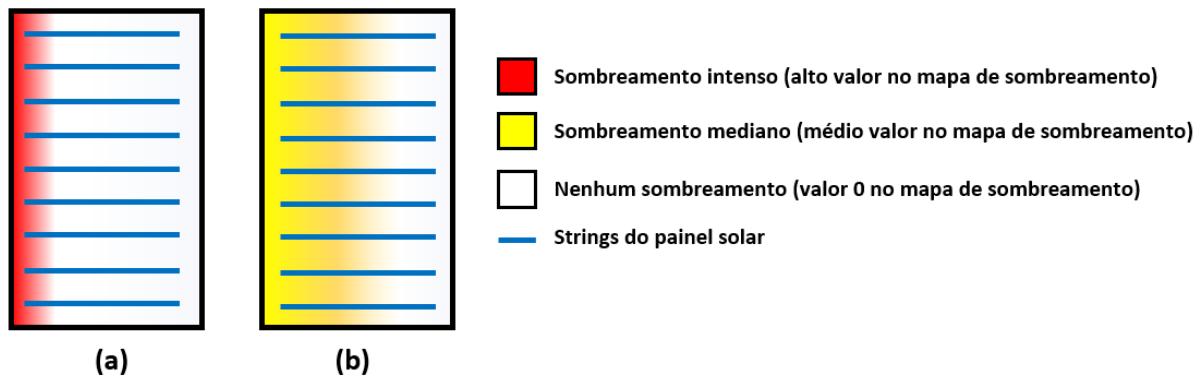
Fonte: Autoral.

Para complementar o entendimento dos resultados, é necessário que o usuário avalie duas das desvantagens desta generalização provocada pelo uso do coeficiente de sombreamento: a heterogeneidade da captação do arranjo, e a heterogeneidade da radiação ao longo do dia.

Em módulos menos modernos, com células arranjadas em *strings* e sem diodos de *by-pass*, ou em arranjos onde os módulos não possuem *by-pass* individualmente, a ocorrência de

sombra é, de certa forma, mais importante que a área total sombreada. Tomando o exemplo da Figura 85, onde ambas as placas têm o mesmo coeficiente de sombreamento, em um módulo que não possui estes dispositivos de proteção, com *strings* transversais, qualquer sombreamento incidindo longitudinalmente nelas as desligaria completamente, visto que pelo menos uma célula de cada *strings* seria desligada, e como estão ligadas em série, todas as outras células na *string*. Neste caso, a placa (a) está em desvantagem: por mais que a área sombreada seja menor, esta área perdura por mais tempo, como indicado pelo alto valor no mapa de sombreamento, e portanto, ela passaria mais tempo sem produzir. Não obstante, a placa (a) tem um maior potencial de produzir *hotspots*, visto que o sombreamento é mais concentrado.

Figura 85 – Exemplo de placas com o mesmo coeficiente de sombreamento em diferentes manchas de sombreamento.



Fonte: Autoral.

Portanto, é importante que o usuário conheça os dispositivos de proteção presentes no modelo de módulo adotado, como também a sua disposição ao longo do mesmo. Placas e arranjos mais modernos, com estes dispositivos, tendem a ter uma produção mais homogênea.

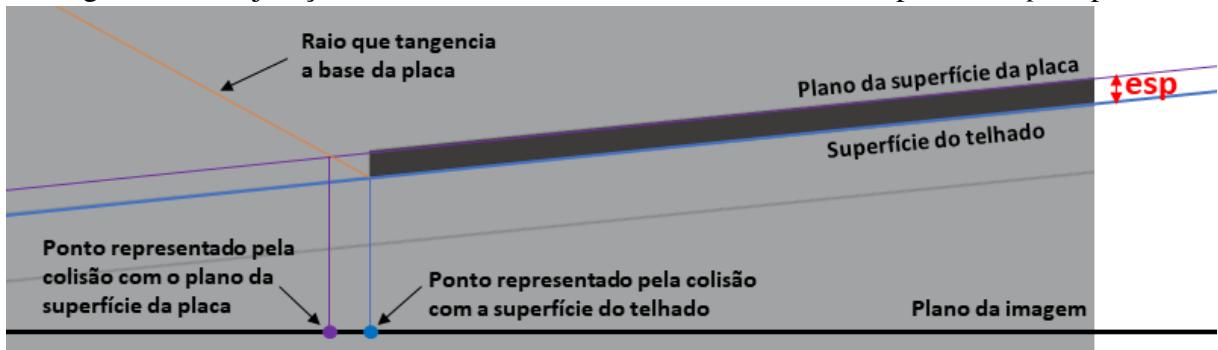
Uma outra desvantagem da metodologia utilizada para classificar o sombreamento neste trabalho é a heterogeneidade da radiação ao longo do dia. Enquanto o modelo de traçado de raios aplicado é eficiente em determinar quantitativamente o sombreamento de uma área, não se obtém quantitativamente o contrário: a quantidade de radiação solar que atinge as placas quando elas estão iluminadas, ou seja, a irradiância. Ao longo do dia, o nível de radiação solar varia, sendo maior próxima ao meio dia, e menor ao nascer e ao pôr do sol. De certa forma, pode-se dizer que o sombreamento em um momento de maior irradiância é mais prejudicial ao sistema, já que a potência que a placa deixa de produzir é maior. Porém, pela continuidade dos valores campo escalar produzido, e da própria natureza da trajetória solar, isto só seria um

problema em uma alocação de muitas placas, próximo ao fisicamente possível pela região viável, visto que as sombras nesses horários são as menores, e se concentram nas bordas da edificação. Na metodologia aplicada, estas placas seriam as últimas a serem locadas.

4.4 Sobre a espessura do módulo

As imagens produzidas e o mapa de sombreamento são resultado da colisão de raios traçados na superfície da região viável, como por exemplo um telhado. Porém, a placa produzirá energia a partir dos raios que recebe em sua superfície, e não na sua base, como é representado nas imagens geradas. Como podemos analisar a partir da Figura 86, a desconsideração da espessura da placa, como ocorre no programa produzido, majora a ocorrência das sombras, criando uma margem que favorece a segurança.

Figura 86 – Majoração do sombreamento ao se desconsiderar a espessura esp da placa.



Fonte: Autoral.

O principal motivo por trás da decisão de se utilizar a superfície da região viável, majorando o sombreamento, é a de se poder utilizar os resultados para qualquer modelo de placa após a sua execução. Caso fosse necessário testar a edificação para uma placa com uma espessura diferente da anterior, os resultados anteriores não seriam válidos, principalmente se a nova espessura for menor. Como o traçado de raios e a produção do mapa de sombreamento são as etapas mais custosas da metodologia apresentada, o programa para a alocação das placas utiliza um resultado que representa bem a ocorrência de sombras, com uma pequena margem que favorece a segurança, e que pode ser utilizado para múltiplos modelos de placa. O modelo tridimensional gerado pelo programa após a alocação considera a espessura esp das placas inserida pelo usuário.

4.5 Sobre o desempenho computacional

Como foi comentado na Seção 4.1, a execução do traçado de raios nos exemplos se deu em um tempo que justifica a sua aplicação, visto que para a produção de imagens dos mesmos horários, em visitas a campo, seriam necessárias pelo menos 28 horas (7 horas por dia de observação), em dias espaçados ao longo de 9 meses. Ainda assim, o tempo de execução do Exemplo 1 é muito próximo desse limite, porém algumas otimizações foram aplicadas já neste exemplo e no Exemplo 2. Um comparativo resumido das execuções dos exemplos, quanto ao traçado de raios, pode ser encontrado no Quadro 1. A seguir serão levantadas justificativas em relação ao tempo de execução do programa.

Quadro 1 – Comparativo entre as execuções do traçado de raios dos Exemplos 1 e 2

| | Ex 1 Esquerda | Ex 1 Direita | Ex 2 Laje |
|-----------------------------|------------------|-----------------|--------------|
| Tempo Total de Execução | 18h 56min | 7h 36min | 7h 52min |
| Máquina Utilizada | 1 | 1 | 1 |
| Vértices | 134 | 134 | 104 |
| Faces Triangulares | 216 | 216 | 172 |
| Etapas de Execução | 4 | 2 | 1 |
| Imagens Geradas | 32 | 32 | 32 |
| Multiprocessamento | Sim | Sim | Sim |
| Pré-mapeamento | Não | Sim | Sim |
| Densidade de Pixels | 50 | 50 | 50 |
| Quantidade de Pixels/Imagem | 950000 | 950000 | 1207500 |
| Pixels calculados/Minuto | 26700,56 | 66666,67 | 81864,41 |

Fonte: Autoral.

O uso da linguagem de programação *Python* se deu pela popularidade da linguagem e sua difusão no ensino introdutório de programação nos cursos de engenharia. Sabe-se, porém, que esta não é uma linguagem de alto desempenho computacional, como C++ ou Fortran. A portabilidade do algoritmo para estas linguagens pode melhorar o tempo de execução do código.

Os processadores atuais costumam ter múltiplos núcleos, e um código que permite a execução das tarefas com multiprocessamento pode ter melhorias significativas de desempenho. No presente trabalho, foi utilizado multiprocessamento no algoritmo de traçado de raios. Esta tarefa é fundamentalmente apropriada para o multiprocessamento, visto que a execução de um *pixel* não depende da execução do *pixel* anterior ou de nenhum outro. Desta forma, cada núcleo pode ficar com a tarefa de produzir um *pixel* de cada vez, e retornar os resultados na ordem que

forma a imagem. Não obstante, a tarefa de traçado de raios pode ser facilmente associada ao uso de GPU's, sendo que há várias GPU's no mercado atualmente que possuem tecnologias nativas que aceleram algoritmos de traçado de raios.

Já a tarefa de inserção das placas não é fundamentalmente apropriada para a abordagem do multiprocessamento: não se pode verificar a inserção de placas para vários *pixels* simultaneamente, visto que a inserção de uma placa em um deles afetaria a possibilidade de inserção em outro pixel. Em outras palavras, a verificação da inserção das placas precisa ser em série, diferentemente da criação dos *pixels* no traçado de raios, que pode ocorrer em paralelo.

A verificação de múltiplos arranjos de placas solares, para que se obtenha o arranjo com o maior coeficiente de sombreamento, pode ser paralelizado, ou seja, macroscopicamente, o problema da alocação de placas pode se beneficiar do multiprocessamento. Não foi dado foco à otimização do tempo de execução da alocação das placas porque este tempo já era significativamente menor em relação ao traçado de raios, na modalidade alinhado.

Outro ponto de melhoria em relação ao código desenvolvido é a otimização do uso de memória: da forma como está, o algoritmo cria muitas cópias de variáveis, e vários passos podem se mostrar desnecessários a partir das especificidades do problema. Um exemplo de implementação que é possibilitada apenas pela especificidade do problema e que trouxe uma melhoria significativa no código é o pré mapeamento de objetos citado anteriormente na Seção 4.1. Como todas as imagens produzidas no traçado de raios colocam o modelo na mesma posição, e a tela emite os raios dos mesmos pontos, o primeiro raio, que vai da tela ao modelo tridimensional, colide sempre com os mesmos objetos, em todas as imagens. O que o pré mapeamento de objetos faz é guardar o índice do triângulo atingido pelo primeiro raio em uma matriz, e para as imagens subsequentes, o programa não vai mais traçar o primeiro raio, que é o raio que sai da tela, e sim apenas o segundo raio, que é o que vai do modelo tridimensional até a direção do sol. O segundo raio precisa ser traçado em cada imagem, visto que a posição do sol muda para cada uma delas. A melhoria é significativa porque para todas as imagens, o primeiro raio não é traçado, então não há iteração entre todos os triângulos da cena, e o segundo raio só é traçado na região viável, onde o primeiro raio detectou uma colisão que já está salva, diminuindo bastante o número de iterações que o programa precisa performar.

5 CONSIDERAÇÕES FINAIS

Tendo em vista que a metodologia aplicada e os resultados obtidos contemplam os objetivos específicos descritos na Seção 1.1.2, é possível afirmar que a utilização de traçado de raios e de heurísticas de empacotamento são uma alternativa viável para a alocação automática de placas solares em telhados. Houve sucesso na aplicação e adaptação do algoritmo tradicional do traçado de raios para o problema proposto, e as simplificações adotadas na heurística de empacotamento conseguem contemplar os casos mais comuns de alocação, considerando, entre outros fatores, a inclinação do telhado, a orientação e alinhamento das placas e o ângulo do plano de trabalho com o norte. Não obstante, as metodologias aplicadas foram validadas, tanto pela literatura, como se deu no caso da alocação das placas, como na construção do mapa de sombreamento, a partir de observações em campo, que é justamente a metodologia que este trabalho visa simplificar.

O desempenho do programa, por mais que não seja o ideal, é suficiente para substituir os métodos manuais de análise do sombreamento em edificações, permitindo uma análise mais prática e rápida, diminuindo os prazos de projetos neste escopo. O uso dos dados retornados pelo programa desenvolvido enriquece o embasamento teórico da locação das placas, tornando-a mais assertiva. A locação passa a não ser mais feita “no olho”, e sim por um software que atesta que aquela posição tomada é a melhor entre as possíveis.

Os resultados do programa ainda exigem uma avaliação humana em casos específicos, e não são suficientes para uma completa análise da viabilidade do sistema, visto que não quantifica a radiação que atinge a placa, e consequentemente, a energia gerada. O retorno sobre o investimento ainda precisa ser analisado separadamente, manualmente ou com o uso de algum *software*. No entanto, é possível afirmar que a metodologia utilizada representa uma evolução em relação à literatura sobre o assunto, tanto na associação entre o sombreamento e o posicionamento das placas, como nas tecnologias utilizadas para sua implementação. Algoritmos de traçado de raios já têm sido utilizados em modelagem e renderização há tempos, mas a possibilidade de se executar estes algoritmos em computadores pessoais é uma realidade recente. Além disso, os aprimoramentos desenvolvidos em relação às heurísticas de empacotamento clássicas, e principalmente a desenvolvida por Dür e Nowak (2008), visando o problema específico trazido por este trabalho, são um passo importante para a obtenção de resultados úteis na prática.

Trabalhos futuros

Algumas melhorias à metodologia adotada por este trabalho podem ser levantadas em trabalhos complementares. Acerca do desempenho computacional do código, alguns pontos de melhoria são:

- aprimoramento dos algoritmos implementados e do uso da memória;
- portabilidade para uma linguagem de alto desempenho;
- uso de processamento por GPU;
- utilização de multiprocessamento distribuído.

O custo computacional da heurística implementada é diretamente proporcional à densidade de pixel γ_p e à quantidade de dias usados na criação do mapa de sombreamento. Um estudo detalhado desses dois parâmetros pode levar a valores que produzam o melhor equilíbrio entre a acuidade do modelo e o tempo de execução.

Outras melhorias não relacionadas ao desempenho do programa, mas que seriam igualmente interessantes, são:

- a possibilidade de se utilizar módulos com orientação arbitrária, não só horizontal ou vertical;
- uma interface que possibilitasse a manipulação do modelo geométrico tridimensional sem a necessidade da utilização de programas de terceiros, como o *Blender*;
- uso de outros algoritmos de otimização para a alocação dos módulos, ou de inteligência artificial.

O programa também pode ser modificado de forma a auxiliar a automação de sistemas de produção solar que alternam a orientação e a angulação dos módulos, como também se beneficiar de dados obtidos por estes sistemas para validar o seu próprio modelo ou enriquecer a obtenção de valores para a trajetória solar e de irradiação. A utilização de algum valor de radiação para o raio que atinge os módulos para cada hora do dia, ou o uso de outros algoritmos, como de radiosidade, em substituição ao traçado de raios, poderiam permitir uma avaliação mais completa do sistema.

REFERÊNCIAS

- AKENINE-MÖLLER, T.; HAINES, E.; HOFFMAN, N. **Real-Time Rendering**. Natick: A K Peters, Ltd., 2008. 1027 p. ISBN 978-1-56881-424-7.
- ALMEIDA, H. A. de; ALMEIDA, E. d. C. V. Potencial da energia solar fotovoltaica no semiárido nordestino. **Concilium**, [S. l.], v. 22, n. 2, p. 197–210, 2022.
- ANEEL - Agência Nacional de Energia Elétrica. **Brasil supera marca de 5 GW de capacidade instalada de usinas fotovoltaicas com geração centralizada**. 2022. Disponível em: <<https://www.gov.br/aneel/pt-br/assuntos/noticias/2022/brasil-supera-marcade-5-gw-de-capacidade-instalada-de-usinas-fotovoltaicas-com-geracao-centralizada>>. Acesso em: 12 jun. 2022.
- ANEEL - Agência Nacional de Energia Elétrica. **Brasil ultrapassa marca de 10 GW em micro e minigeração distribuída**. 2022. Disponível em: <<https://www.gov.br/aneel/pt-br/assuntos/noticias/2022/brasil-ultrapassa-marcade-10-gw-em-micro-e-minigeracao-distribuida>>. Acesso em: 12 jun. 2022.
- ANEEL - Agência Nacional de Energia Elétrica. **Capacidade de geração de energia no Brasil supera os 180 GW**. 2022. Disponível em: <<https://www.gov.br/aneel/pt-br/assuntos/noticias/2022/capacidade-de-geracao-de-energia-no-brasil-supera-os-180-gw>>. Acesso em: 11 jul. 2022.
- ASSOCIAÇÃO NACIONAL DE CRUZEIROS. **Declinação do Sol**. 2022. Disponível em: <<https://www.ancruzeiros.pt/ancdrp/declinação-do-sol#:~:text=Declinação do%20do%20Sol,%20a,isto%20o%20ponto%20da%20embarcação>>. Acesso em: 07 out. 2022.
- AUTODESK. **Perguntas Frequentes Sobre a Descontinuação do Ecotect Analysis**. 2016. Disponível em: <<https://knowledge.autodesk.com/pt-br/search/result/caas/sfdarticles/sfdarticles/PTB/Ecotect-Analysis-Discontinuation-FAQ.html>>. Acesso em: 14 jun. 2022.
- AZÉVEDO, R. O. **Análise de Fatores Determinantes na Viabilidade Econômica de Investimentos em Energia Solar e Eólica**. 107 f. Dissertação (Mestrado em Engenharia de Produção) — UFPB, João Pessoa-PB, 2020.
- BIRGIN, E. G.; LOBATO, R. D.; MORABITO, R. An effective recursive partitioning approach for the packing of identical rectangles in a rectangle. **Journal of the Operational Research Society**, v. 61, n. 2, p. 306–320, Fevereiro 2010.
- BOTSCH, M. et al. **Polygon Mesh Processing**. Natick: A K Peters/CRC Press, 2010. v. 1. 230 p. ISBN 978-1-56881-426-1.
- BOURGES, B. Improvement in solar declination computation. **Solar Energy**, v. 35, n. 4, p. 367–369, Fevereiro 1985.
- BÜHLER, A. J. Princípios básicos da conversão fotovoltaica, histórico e status atual desta tecnologia. In: SEMINÁRIO INTERNACIONAL DE CONSTRUÇÕES SUSTENTÁVEIS, 5., 2016, Farroupilha. Farroupilha: Instituto Federal de Educação, Ciência e Tecnologia - RS, 2016. Disponível em: <https://www.imed.edu.br/Uploads/Alexandre%20B%C3%A7hler_Energia%20Solar%20Fotovoltaica_Situa%C3%A7%C3%A3o%20Atual%20e%20Perspectiva%20para%20o%20Brasil.pdf>. Acesso em: 13 jun. 2022.

CARNEIRO, J. A. O. Apoio às aulas teóricas da UC de Electromagnetismo B do Mestrado Integrado em Engenharia Civil, **Módulos Fotovoltaicos – Características e Associações**. Azurém: [s.n.], 2010. Disponível em: <http://repository.uminho.pt/bitstream/1822/16961/1/Módulos%20Fotovoltaicos_Características%20e%20Associações.pdf>. Acesso em: 13 jun. 2022.

CHAUDHARY, A. S.; CHATURVEDI, D. K. Observing hotspots and power loss in solar photovoltaic array under shading effects using thermal imaging camera. **International Journal of Electrical Machines and Drives**, Dayalbagh, v. 3, n. 1, p. 15–23, out. 2017. Disponível em: <https://www.researchgate.net/profile/Akash-Chaudhary-6/publication/325896762_Observing_Hotspots_and_Power_Loss_in_Solar_Photo voltaic_Array_Under_Shading_Effects_Using_Thermal_Imaging_Camera/links/5b2b8400a6fdcce657340f4b/Observing-Hotspots-and-Power-Loss-in-Solar-Photo voltaic-Array-Under-Shading-Effects-Using-Thermal-Imaging-Camera.pdf>. Acesso em: 13 jun. 2022.

COLLISCHONN, W.; DORNELLES, F. **Hidrologia para Engenharia e Ciências Ambientais**. Porto Alegre: ABRH, 2013. v. 1. 350 p. ISBN 978-85-8868-634-2.

Deutsche Gesellschaft für Sonnenenergie. **Planning and Installing Photovoltaic Systems: A Guide for Installers, Architects and Engineers**. [S.l.]: Earthscan, 2008. (Planning and Installing Series). ISBN 9781844074426.

DÜR, M.; NOWAK, N. Packing solar cells on a roof. **Optim Eng**, Darmstadt, v. 10, p. 397–408, set. 2008.

GOMES, J.; VELHO, L. **Fundamentos da Computação Gráfica**. Rio de Janeiro: IMPA, 2015. 603 p. ISBN 978-1-4822-2941-7.

GÓMEZ, J. M. R.; CARLESSO, F.; E, V. L.; DA SILVA, L. A irradiância solar: conceitos básicos. **Revista Brasileira de Ensino de Física**, São Paulo, v. 40, n. 3, 2018.

HOLTZ, G. C. D. C. **Traçado automático de envoltórias de esforços em estruturas planas utilizando um algoritmo evolucionário**. Dissertação (Mestrado) — PUC-Rio de Janeiro, Rio de Janeiro, 2005.

HOPPER, E.; TURTON, B. C. H. An empirical investigation of meta-heuristic and heuristic algorithms for a 2d packing problem. **European Journal of Operational Research**, v. 128, n. 1, p. 34–57, Janeiro 2001.

HORICATCHER. **Meteonorm**. 2022. Disponível em: <<https://meteonorm.com/en/product/horicatcher>>. Acesso em: 14 jun. 2022.

ILSE, K. *et al.* Techno-economic assessment of soiling losses and mitigation strategies for solar power generation. **Joule**, v. 3, n. 10, p. 2303–2321, 2019.

JOHANNES, O. **Exact Optimization Algorithms for the Aggregation of Spatial Data**. Dissertação (Doutorado) — Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn, 2020.

KLUGE, R. A.; TEZOTTO-ULIANA, J. V.; DA SILVA, P. P. M. Aspectos fisiológicos e ambientais da fotossíntese. **Revista Virtual de Química**, v. 7, n. 1, p. 56–73, 2015.

MACHADO, C. T.; MIRANDA, F. S. Energia solar fotovoltaica: Uma breve revisão. **Revista Virtual de Química**, v. 7, n. 1, p. 126–143, 2015.

- MARINOSKY, D. L.; SALAMONI, I. T.; RÜTHER, R. Pré-dimensionamento de sistema solar fotovoltaico: Estudo de caso do edifício sede do crea-sc. In: I CONFERÊNCIA LATINO-AMERICANA DE CONSTRUÇÃO SUSTENTÁVEL, 1., 2004, São Paulo. São Paulo, 2004.
- MELO, D. L. d. A. **Um pipeline para renderização fotorrealística de tempo real com ray tracing para a realidade aumentada**. Dissertação (Mestrado) — UFPE, Recife, 2012.
- MERTENS, K. **Photovoltaics: Fundamentals, Technology and Practice**. [S.l.]: John Wiley & Sons, 2014. v. 1. ISBN 9781118634165.
- NASCIMENTO, C. A. **Princípio de Funcionamento da Célula Fotovoltaica**. Dissertação (Especialização em Fontes Alternativas de Energia) — Universidade Federal de Lavras, Lavras, MG, 2004.
- OLIVEIRA, J. F.; JÚNIOR, A. N.; SILVA, E.; CARRAVILLA, M. A. A survey on heuristics for the two-dimensional rectangular strip packing problem. **Pesquisa Operacional**, Rio de Janeiro, v. 36, n. 2, p. 197–226, 2016.
- OLIVEIRA, O. A. M. **Efficient Heuristics for Two-Dimensional Cutting and Packing Problems**. Tese (Doutorado) — FEUP, Porto, 2019.
- PARKER, S. G. *et al.* Optix, a general purpose ray tracing engine. **ACM Transactions on Graphics**, New York, v. 29, n. 4, p. 1–13, 2010.
- PEREIRA, G. C. **Estudo Sobre A Substituição Do Silício Pelo Grafeno Em Placas Solares Fotovoltaicas**. Dissertação (Graduação) — Universidade de Taubaté - Departamento de Engenharia Mecânica e Elétrica, Taubaté, 2019.
- PVPERFORMANCE. **Sun Position**. 2022. Disponível em: <<https://pvpmc.sandia.gov/modeling-steps/1-weather-design-inputs/sun-position/>>. Acesso em: 07 out. 2022.
- PVSYST. **Features – Shading Scene**. 2022. Disponível em: <<https://www.pvsyst.com/features/>>. Acesso em: 14 jun. 2022.
- RAKOTONIRAINY, R. G.; VUUREN, J. H. v. Improved metaheuristics for the two-dimensional strip packing problem. **Applied Soft Computing**, Matieland, v. 92, n. 1, 2020.
- SERATIUK, A. R. **Análise do Sombreamento em Sistemas de Proteção Solar Fotovoltaica Integrados à Arquitetura – Estudo de Caso para Fachadas de Edificação na Cidade de Curitiba**. Monografia (Especialização em Construções Sustentáveis) — UTFPR, Curitiba, 2016.
- SHIRLEY, P.; MARSCHNER, S. **Fundamentals of Computer Graphics**. Boca Raton: CRC Press, 2015. v. 4. 721 p. ISBN 978-1-4822-2941-7.
- SOLAR PATHFINDER. **Solar Pathfinder**. 2022. Disponível em: <<https://www.solarpathfinder.com/PF>>. Acesso em: 14 jun. 2022.
- SOULAYMAN, S. Comments on solar azimuth angle. **Renewable Energy**, v. 123, n. 1, p. 294–300, Agosto 2018.
- SUNEARTHTOOLS. **Sun Position**. 2022. Disponível em: <https://www.sunearthtools.com/dp/tools/pos_sun.php>. Acesso em: 07 jul. 2022.

TORTELLI, C. O. **O Efeito do Sombreamento na Eficiência do Sistema Fotovoltaico do Escritório Verde da UTFPR.** Monografia (Especialização em Construções Sustentáveis) — UTFPR, Curitiba, 2016.

VALADARES, E. C.; MOREIRA, A. M. Ensinando física moderna no segundo grau: Efeito fotoelétrico, laser e emissão de corpo negro. **Caderno Brasileiro de Ensino de Física**, Florianópolis, v. 15, n. 2, 1998. Acesso em: 13 jun. 2022.

VELHO, L.; GOMES, J. **Sistemas Gráficos 3D.** Rio de Janeiro: IMPA, 2007. 336 p. ISBN 85-244-0167-2.

WÄSCHER, G.; HAUBNER, H.; SCHUMANN, H. An improved typology of cutting and packing problems. **European Journal of Operational Research**, Magdeburg, v. 183, n. 1, p. 1109–1130, 2007.

ZOMER, C. D. **Método de Estimativa da Influência do Sombreamento Parcial na Geração Energética de Sistemas Solares Fotovoltaicos Integrados em Edificações.** Tese (Doutorado) — UFSC, Florianópolis, 2014.