

## Estratégia de cálculo de suporte em manufatura aditiva explorando estrutura de dados

### Support calculation strategy in additive manufacturing exploiting data structure

Henrique Romaniuk Ramalho<sup>1</sup>, Ricardo Dutra da Silva<sup>2</sup>, Rodrigo Minetto<sup>3</sup>, Neri Volpato<sup>4</sup>

#### RESUMO

Uma das etapas de planejamento de processo de manufatura aditiva é a geração de estruturas de suporte, que envolve identificar regiões de um modelo geométrico 3D que precisam de sustentação. Usualmente, as peças são representadas computacionalmente através de uma malha de triângulos desconectados. Desse modo, com intuito de criar estas regiões, é necessário seccionar o modelo, gerando camadas, e aplicar operações booleanas nas listas de contornos de todas as camadas. O objetivo deste estudo é apresentar uma alternativa que diminua a quantidade de operações realizadas para encontrar as regiões de suporte. A proposta é utilizar as características de uma estrutura de dados organizada que preserva informações de vizinhança, para mapear as regiões suspensas e projetá-las nas camadas inferiores. Os resultados iniciais mostraram que é possível diminuir a quantidade de camadas analisadas para aplicar as operações booleanas.

**PALAVRAS-CHAVE:** Estrutura de dados; geração de suporte; manufatura aditiva.

#### ABSTRACT

One of the additive manufacturing process planning steps is the generation of support structures, which involves identifying regions of a 3D geometric model that require support. Usually, the parts are represented computationally through a mesh of disconnected triangles. Therefore, in order to create these regions, it is necessary to section the model, generating layers, and apply boolean operations to the contour lists of all layers. The objective of this study is to present an alternative that reduces the number of operations performed to find support regions. The proposal is to use the characteristics of an organized data structure that preserves neighborhood information, to map the suspended regions and project them into the lower layers. Initial results showed that it is possible to reduce the number of layers analyzed to apply boolean operations.

**KEYWORDS:** Data structure; support generation; additive manufacturing.

## INTRODUÇÃO

O processo de fabricação por manufatura aditiva (AM) consiste na adição sucessiva de material na forma de camadas para produzir uma peça tridimensional (VOLPATO, 2017). Para gerar os dados necessários para a impressão de um objeto, é necessária a realização de um planejamento de processo. Este passo consiste em algumas etapas, como fatiamento, cálculo de suporte para regiões suspensas e cálculo de trajetória de impressão.

<sup>1</sup> Bolsista da Fundação Araucária. Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil. E-mail: henriqueramalho@alunos.utfpr.edu.br. ID Lattes: 3915374566037461.

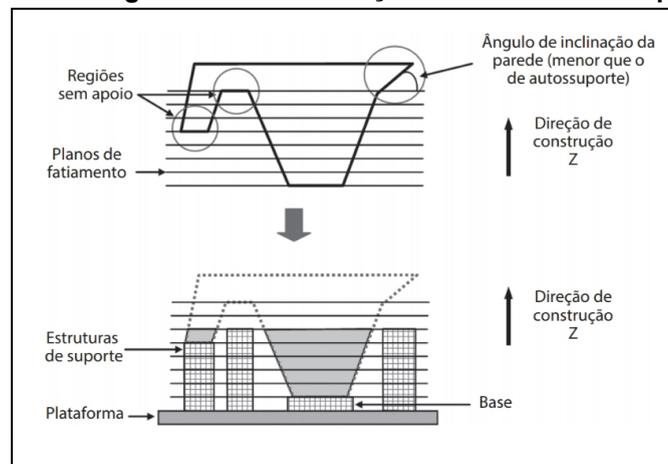
<sup>2</sup> Docente no Departamento Acadêmico de Informática (DAINF). Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil. E-mail: rdsilva@utfpr.edu.br. ID Lattes: 8512085741397097

<sup>3</sup> Docente no Departamento Acadêmico de Informática (DAINF). Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil. E-mail: rminetto@utfpr.edu.br. ID Lattes: 8366112479020867.

<sup>4</sup> Orientador. Docente no Departamento Acadêmico de Mecânica (DAMEC). Universidade Tecnológica Federal do Paraná, Curitiba, Paraná, Brasil. E-mail: nvolpato@utfpr.edu.br. ID Lattes: 8414652619232683.

A representação de um modelo geométrico 3D para AM, geralmente é feita através de um conjunto de triângulos não organizados topologicamente, ou seja, uma lista simples de triângulos em que não há conhecimento de arestas e vértices compartilhados entre dois ou mais triângulos. O estágio de geração de suporte se resume em mapear as regiões sem apoio, ou que possuem ângulo de inclinação com a horizontal menor que o de autossuporte - menor ângulo para o qual é possível sustentar a camada superior com o material da camada inferior. Na sequência, deve-se estender os limites dessas regiões até a base da plataforma da máquina de impressão, de forma a criar uma estrutura de sustentação. A Figura 1 ilustra esquematicamente uma vista lateral de uma peça que possui regiões que necessitam de estruturas de suporte e a representação da mesma parcialmente impressa.

Figura 1 - Regiões sem sustentação e estruturas de suporte.



Fonte: (VOLPATO, 2017).

Usualmente, o algoritmo utilizado para o cálculo dessas estruturas percorre todas as camadas de cima para baixo realizando operações booleanas. Como não há informação prévia acerca dessas regiões, faz-se necessário operações lógicas sobre todas as fatias. Áreas obtidas pela diferença entre duas camadas adjacentes da peça, indicam regiões suspensas. Este conjunto de operações tem como entrada um conjunto de camadas (fatias) compostas por contornos 2D. Como saída, a etapa de cálculo de suporte retorna um conjunto de camadas, contendo contornos que representam os limites das estruturas de sustentação. Essa metodologia, todavia, apresenta um elevado tempo de processamento à medida que o número de fatias cresce. Neste sentido, este artigo apresenta uma alternativa para o cômputo das estruturas de sustentação, com intuito de que não seja necessária a realização de operações booleanas em todas as camadas, através de um mapeamento prévio por meio de uma malha triangular organizada topologicamente. Para alcançar este objetivo, fez-se o uso da estrutura de dados topológica *Doubly Connected Edge List* (DCEL) (BERG, 2008). Essa estrutura fornece informações de vizinhança entre arestas e vértices. Desse modo, foi possível mapear regiões suspensas conjuntas, através de uma navegação na superfície, percorrendo triângulos adjacentes e conferindo suas respectivas inclinações.

## METODOLOGIA

A partir de uma malha de um modelo 3D no formato STereoLithography (STL), a geometria tridimensional é traduzida para a DCEL através da biblioteca Computational Geometry Algorithms Library (CGAL<sup>5</sup>). A estrutura de dados é composta por faces, vértices e arestas. Ramalho et al. (2022) apresentam de forma mais detalhada o funcionamento e os métodos utilitários da estrutura de dados utilizada.

Com a finalidade de determinar as regiões de suporte, utilizou-se a estratégia de busca em largura na superfície da malha. Para isso, cada face triangular da malha foi tratada como um vértice em um grafo, e cada par de meia-aresta, corresponde a uma aresta no grafo. Portanto, uma superfície com  $n$  triângulos e  $3n$  meia-arestas é possível ser traduzida como um grafo  $G = (V, E)$ , em que  $V$  é o conjunto de vértices do grafo e  $E$  é o conjunto de arestas. Ainda,  $|V| = n$  e  $|E| = \frac{3}{2}n$ .

Dada uma malha e um ângulo limite de autossuporte, o algoritmo itera sobre a lista de triângulos procurando por uma face que possui inclinação menor do que o ângulo fornecido. Em seguida, realiza-se a busca em largura, tendo como vértice inicial a face encontrada. Durante a execução da rotina, marca-se todas as faces alcançáveis, a partir do início, com um mesmo rótulo, pois pertencem a uma mesma região. O algoritmo continua, até que toda a lista de triângulos seja percorrida. A seguir, é possível visualizar os pseudocódigos das funções de mapeamento e de busca, respectivamente.

---

### Algoritmo 1: MAPEAMENTO

---

**Entrada:** Grafo  $G(V, E)$ , Ângulo de autossuporte  $\alpha$   
**Saída:** Conjunto  $S \subset V$  com cada  $s \in S$  rotulado

```
1 início
2    $r \leftarrow 0$ 
3   para  $v \in V$  faça
4     se  $\text{FACESSUSPENSA}(v, \alpha)$  e  $\text{NÃOVISITADO}(v)$  então
5       BUSCA( $G, v, r, \alpha$ )
6        $r \leftarrow r + 1$ 
7     fim
8   fim
9 fim
```

---

Fonte: Autoria própria (2023).

No Algoritmo 1, a linha 2 inicializa a variável de rótulo de regiões. Em seguida, na linha 3, é definido o laço de repetição que realizará o mapeamento e rotulação dos triângulos visitados.

---

<sup>5</sup> CGAL, **The Computational Geometry Algorithms Library**, <http://www.cgal.org/>, Acesso em: Agosto, 2023.

---

**Algoritmo 2: BUSCA**

---

**Entrada:** Grafo  $G(V, E)$ , Vértice  $v$ , Rótulo  $r$ , Ângulo de autossuporte  $\alpha$

**Saída:** Conjunto  $R \subset V$  com cada  $x \in R$  rotulado

```

1 início
2   Fila  $F$ 
3   ADICIONAR( $F, v$ )
4   enquanto NÃOVAZIA( $F$ ) faça
5     Vértice  $s \leftarrow$  REMOVER( $F$ )
6     para  $u \in$  VIZINHOS( $s$ ) faça
7       se NÃOVISITADO( $u$ ) e FACESUSPENSA( $u, \alpha$ ) então
8         INSERE( $F, u$ )
9         ROTULAR( $u, r$ )
10      fim
11    fim
12  fim
13 fim
    
```

---

Fonte: Autoria própria (2023).

No Algoritmo 2, a implementação da busca em largura é feita com uma fila. Enquanto todas as faces visitadas não forem removidas da fila, o algoritmo continua explorando as faces adjacentes até que sejam rotulados todos os triângulos de uma mesma região. Para isso, a linha 6 se encarrega de visitar a vizinhança de cada face e expandir a busca, caso o triângulo adjacente não tenha sido visitado e seja suspenso.

Concluído o mapeamento de todas as regiões, é necessário a realização de um procedimento de extração dos contornos das partes rotuladas da superfície. Para tanto, beneficiou-se das características topológicas da malha e conhecimentos de vizinhança para navegar até as faces de borda das regiões, e iterar entre as arestas, mantendo-se na região de fronteira. A coordenada  $z$  do contorno foi suprimida, de forma a tornar um contorno bidimensional. A seguir, é possível visualizar o pseudocódigo do algoritmo de extração de contornos.

---

**Algoritmo 3: CONTORNOS**

---

**Entrada:** Lista de regiões  $R$

**Saída:** Conjunto  $S$  de contornos

```

1 início
2    $S \leftarrow []$ 
3   para  $r \in R$  faça
4      $C \leftarrow []$ 
5      $t \leftarrow$  TRIÂNGULODEFRONTIEIRA( $r$ )
6      $H \leftarrow$  ARESTDEFRONTIEIRA( $t$ )
7      $HS \leftarrow H$ 
8     repita
9       Point  $P \leftarrow$  ORIGEM( $H$ )
10      ADICIONAR( $C, P$ )
11       $H \leftarrow$  PRÓXIMAARESTA( $H$ )
12      enquanto NÃOÉTRIÂNGULODEFRONTIEIRA(FACE( $H$ )) faça
13         $H \leftarrow$  PRÓXIMAARESTA(ARESTAGÊMEA( $H$ ))
14      fim
15    até  $H = HS$ ;
16    ADICIONAR( $S, C$ )
17  fim
18 fim
    
```

---

Fonte: Autoria própria (2023).

No Algoritmo 3, o percorrimento das regiões ocorre por meio das meias-arestas promovidas pela malha organizada. Na linha 8, inicia-se a iteração nas faces da região de suporte. Em um primeiro momento, o ponto de origem da aresta em questão é salvo, em seguida, a próxima aresta de uma mesma face é acessada. Enquanto a face desta aresta recebida em iteração não fizer fronteira com a parte não suspensa da malha, a variável  $H$  continua sendo atualizada. Por fim, ao voltar à aresta de partida, o ciclo é fechado e o contorno é salvo.

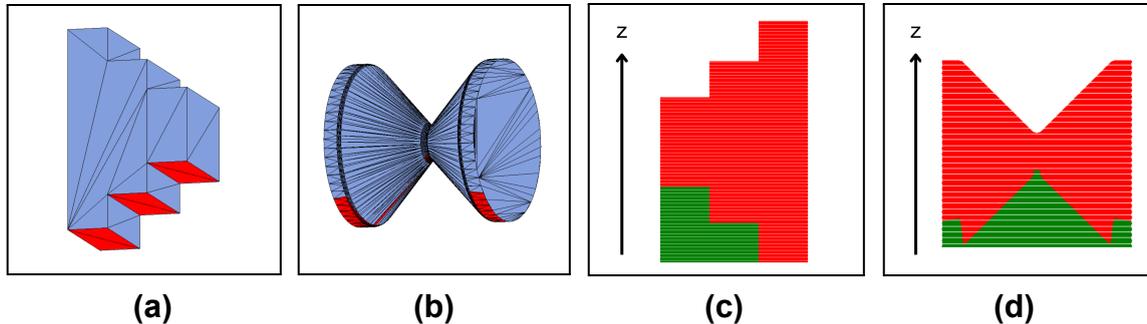
Após o mapeamento dos locais suspensos e extração dos contornos é feita a projeção dos contornos 2D. Primeiro, percorreu-se as regiões de suporte de forma a analisar a maior altura, no eixo  $z$ , de ocorrência de um local suspenso, de modo a encontrar a primeira camada de projeção - a camada com maior altura  $z$  que possui suporte. Assim, inicia-se as operações booleanas de diferença de todos os contornos 2D de suporte com os contornos de limite de peça, em seguida, a união dos contornos resultantes. Vale ressaltar que, o conjunto de contornos inicial, antes de realizar as operações booleanas, de uma camada, é sempre o resultado final da anterior. Este método se repete até que todos os contornos de suporte sejam projetados da camada válida mais alta até a primeira camada.

A implementação do método foi realizada no *software* Rapid Prototyping Process Planning (RP3), desenvolvido pelo Núcleo de Manufatura Aditiva e Ferramental (NUFER) da UTFPR-CT. E algumas peças foram submetidas ao cálculo do suporte.

## RESULTADOS

As Figuras 2(a) e 2(b) ilustram dois modelos 3D de duas peças onde é possível visualizar o mapeamento das regiões que precisam de suporte (faces em vermelho). As Figuras 2(c) e 2(d) mostram as estruturas de sustentação geradas em uma vista lateral. Nestas é possível ver os contornos (em vermelho) criados na etapa de fatiamento e os contornos referentes a estrutura de suporte (em verde). No exemplo da Figura 2(a), pelo método tradicional são necessárias 909 iterações booleanas para se calcular o suporte e no método proposto somente 287. Já no exemplo da Figura 2(b), pelo método tradicional são necessárias 273 iterações booleanas para se calcular o suporte e no método proposto somente 114.

Figura 2 - Modelos 3D com indicação das regiões suspensas (a) e (b) e vista lateral de camadas e estruturas de sustentação (c) e (d).



Fonte: Autoria própria (2023)

## CONCLUSÃO

A implementação da estratégia de mapeamento e projeção de contornos de suporte apresentou resultados iniciais favoráveis, pois houve diminuição da quantidade de camadas analisadas do modelo. No entanto, é necessária a realização de tomadas de tempo, para que se verifique se a diminuição da quantidade de análises não é compensada pela rotina de mapeamento prévio.

Um aprimoramento da metodologia apresentada ainda é necessário para permitir o mapeamento de regiões suspensas pontiagudas formadas por faces com inclinação menor que o ângulo de autossuporte.

## Agradecimentos

Agradecimentos à Fundação Araucária pela bolsa de iniciação científica concedida e aos professores Neri Volpato, Ricardo Dutra da Silva, e Rodrigo Minetto pelas orientações durante o decorrer do período.

## Conflito de interesse

Não há conflito de interesse.

## REFERÊNCIAS

VOLPATO, Neri. **Manufatura Aditiva: Tecnologias e Aplicações da Impressão 3D**. São Paulo: Blucher, 2017.

BERG, Mark de et al. **Computational Geometry: Algorithms and Applications**. 3rd ed. Santa Clara, CA, USA: Springer-Verlag TELOS, 2008.

RAMALHO, Henrique. **Comparação de duas metodologias para fatiamento de malhas triangulares para manufatura aditiva**. XXVII Seminário de Iniciação Científica e Tecnológica da UTFPR, 2022.