

Simpósio de Integração Acadêmica

“Bicentenário da Independência: 200 anos de ciência, tecnologia e inovação no Brasil e 96 anos de contribuição da UFV”

SIA UFV 2022



PinMesh: utilização de Uniform Grid e Flood Fill na resolução de predicados geométricos

Autor: Henrique Campos Padula – Departamento de Informática - Universidade Federal de Viçosa - henrique.padula@ufv.br
Orientador: Salles Viana Gomes de Magalhães – Departamento de Informática - Universidade Federal de Viçosa - salles@ufv.br
Palavras-Chave: Meshe, otimização, localização, grid

Modalidade: Pesquisa | **Área de conhecimento:** Ciências Exatas e Tecnológicas | **Área temática:** Ciência da Computação

Introdução

Uma mesh normalmente é utilizada para representar objetos e regiões em 3D. Dado uma mesh, existem diversas tarefas que podem ser realizadas sobre ela, sendo uma das mais importantes a de localização de pontos dentro do espaço. Contudo, devido a fatores como a existência de vários casos especiais, possíveis erros de arredondamento e grande tamanho de meshes a criação de um algoritmo que realiza tal localização de forma robusta e eficiente é algo não-trivial.

Objetivos

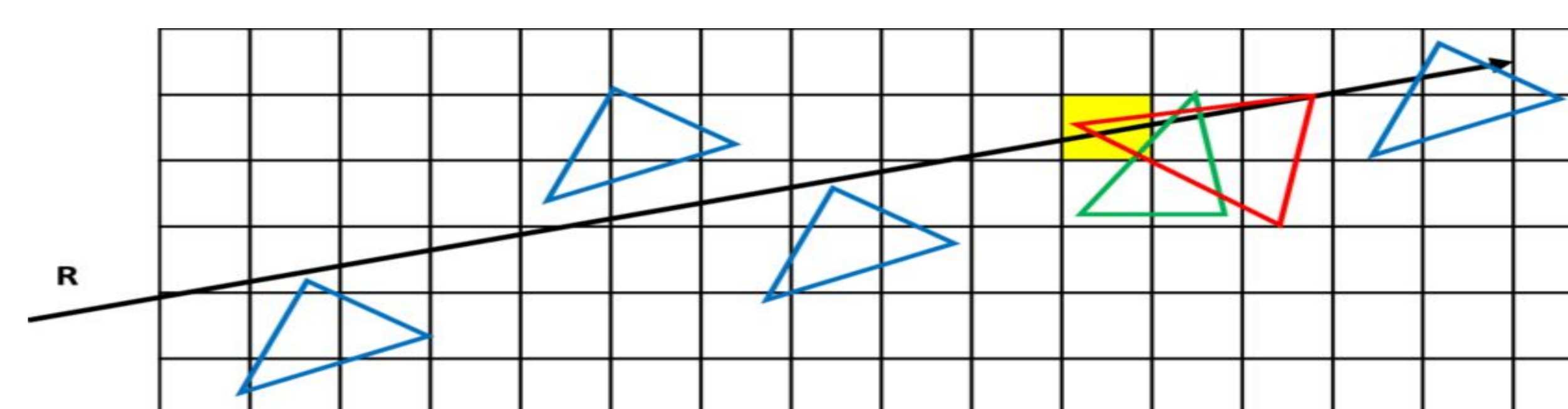
Durante o projeto de iniciação científica, trabalhou-se com o aprimoramento de um algoritmo de localização de pontos criado pelo professor orientador em conjunto com outros alunos em iniciações passadas. Tal aprimoramento baseia-se na indexação de células vazias dentro da figura, evitando que indexações repetidas precisem ser feitas, acelerando ainda mais o algoritmo e sem afetar a sua exatidão e robustez.

Metodologia

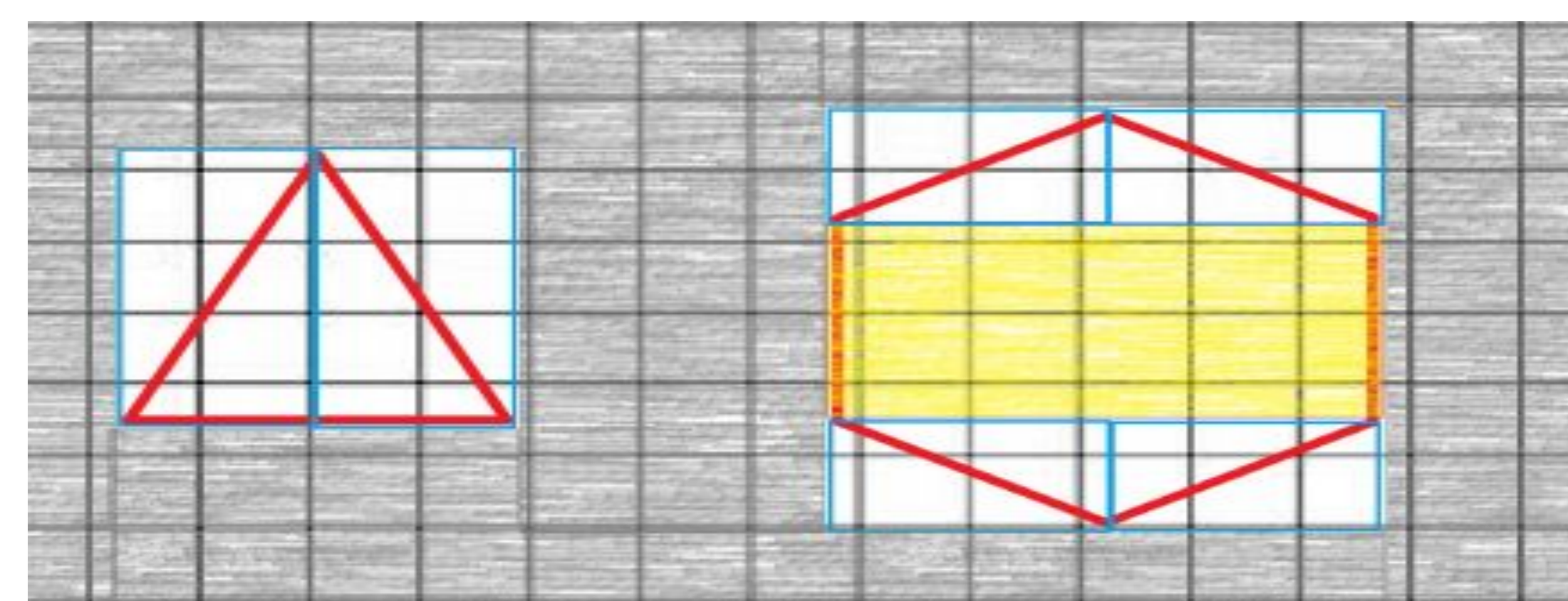
Uma das otimizações no algoritmo consiste na utilização de células de indexação, juntamente com a utilização de bounding boxes para acelerar os cálculos. Inicialmente, se estamos procurando pela interseção entre várias meshes, temos que relacionar cada ponto de uma figura com todos os pontos das outras figuras, gerando um grande número de pares de pontos a serem analisados, sendo assim, uma das otimizações implementadas em trabalhos anteriores consiste em utilizar uma grid tridimensional para a indexação das meshes, diminuindo drasticamente o número de cálculos, visto que apenas arestas que estão presentes em uma mesma célula podem possuir interseção entre si.

Ao se tratar de tetramesh, podemos concluir que não há pontos dentro da figura que não estarão dentro da bounding box, entretanto, para o caso de não tetrameshes, isso não é necessariamente verdade.

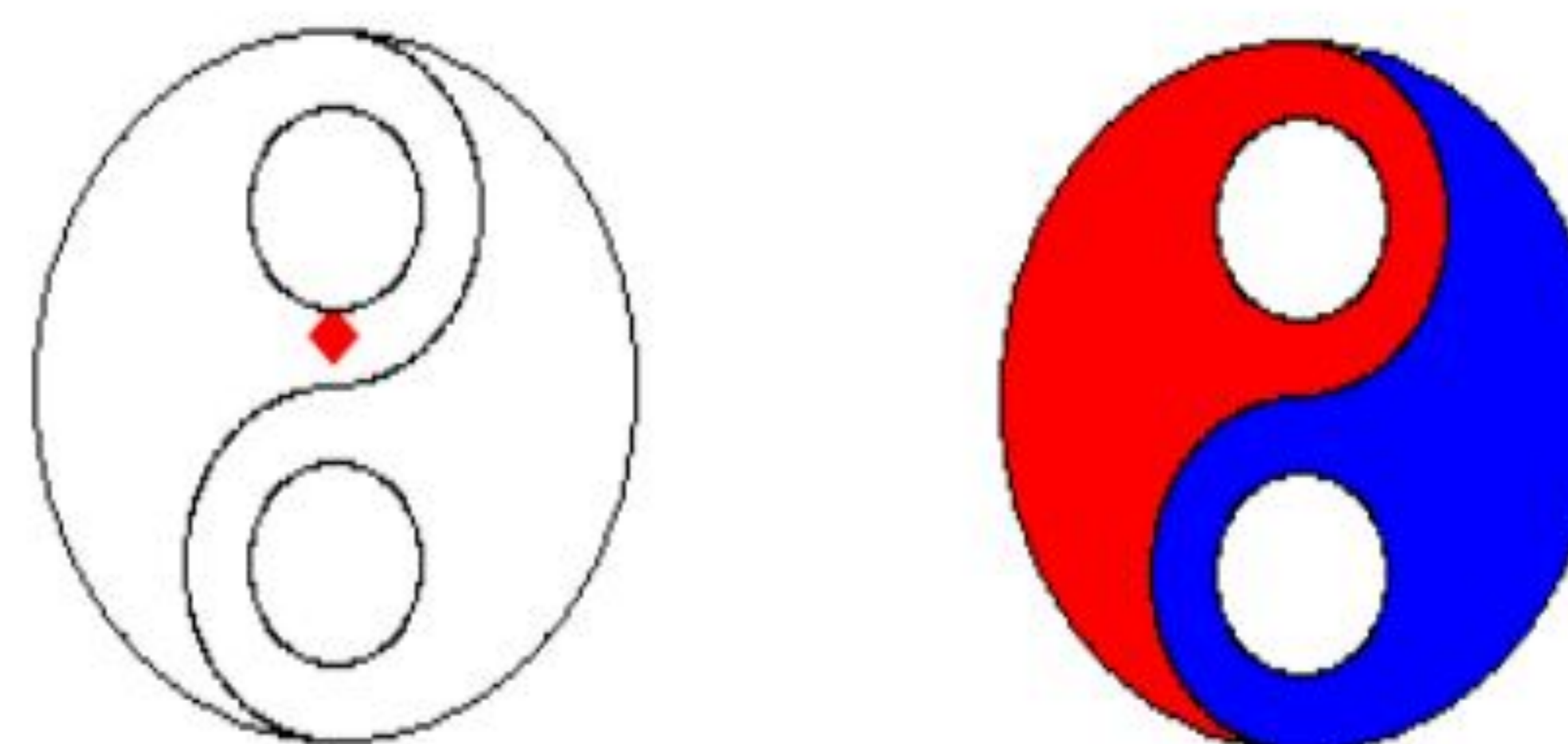
Nesse contexto, propomos uma melhoria, que consiste na utilização do flood fill para marcar as células vazias. Temos que, se um ponto está dentro de uma célula vazia (sem triângulos), toda a célula estará dentro da mesma região, e consequentemente, podemos concluir que as células vizinhas vazias também estarão dentro dessa região. Com isso, podemos acelerar o cálculo, pois não precisaremos localizar a mesma célula várias vezes.



Uso de grade para acelerar testes de interseção com um segmento



Exemplo de representação no espaço bidimensional



Antes e depois da aplicação do algoritmo de flood fill

Conclusão

Por meio da utilização do algoritmo de flood fill, juntamente com a estruturação de grid criada anteriormente, conseguimos alcançar uma melhoria no desempenho do algoritmo, sem alterar a sua precisão e robustez

Bibliografia

S. V. G. de Magalhães, “Exact and parallel intersection of 3D triangular meshes”, tese de doutoramento, Rensselaer Polytechnic Institute, Troy, NY, USA, 2017.
M. de Matos Menezes, S. V. G. Magalhães, M. A. de Oliveira, W. R. Franklin e R. E. de Oliveira Bauer Chichorro, “Fast parallel evaluation of exact geometric predicates on GPUs”, J. Computer Aided Design, n.o 103285, J. Peiro, S. Shontz e R. Viertel, eds., set. de 2022, Special Issue: 28th International Meshing Roundtable: Mesh Modeling for Simulations and Visualization. doi: <https://doi.org/10.1016/j.cad.2022.103285>. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0010448522000616>.