

Relatório Inclusão de Ponto em Polígonos

Bruno Lippert e Pedro Chem

Escola Politécnica – Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Porto Alegre – RS – Brasil

1. Introdução

Este relatório tem por objetivo demonstrar os resultados obtidos no trabalho de inclusão de pontos em polígonos para os diversos algoritmos implementados. O problema consiste em, dado um polígono carregado de um arquivo, gerar o *Convex Hull* deste polígono e a partir de um conjunto de pontos gerados aleatoriamente, definir se estes pontos estão inclusos no polígono ou não.

2. Algoritmos implementados

Para realizar o trabalho, foram implementados os três algoritmos definidos: Inclusão por força bruta, Inclusão por força bruta com faixas e *Convex Hull*. Para o funcionamento do programa foi disponibilizado 4 opções de combinações destes algoritmos, para assim, medir suas eficiências, são elas: *Convex Hull* + Força bruta, *Convex Hull* + Força bruta com faixas, apenas Força bruta e apenas Força bruta com faixas, na tabela 1 podemos visualizar os tempos que cada combinação tomou para ser executada e quantas vezes as funções *HaIntersec* e *ProdVetorial* foram chamadas para 200, 2000 e 20000 pontos aleatórios gerados.

2.1. Convex Hull

O algoritmo implementado para gerar o Convex Hull funciona da seguinte forma:

1. Encontrar os vértices do polígono com o menor e o maior valor no eixo Y.
2. Adicionar o menor vértice do eixo Y na lista de vértices do Convex Hull e definir este ponto como o ponto atual.
3. A partir do ponto atual, indo sempre para a direita, adicionar o vértice do polígono que forme o menor ângulo horizontal com o ponto atual a lista de vértices do Convex Hull, e definir este vértice como o ponto atual.
4. Repetir o passo 3 até que chegue no maior vértice do eixo Y.
5. Repetir o passo 3 mas considerando o lado esquerdo ao invés do direito.
6. Finalizar ao chegar novamente no vértice com o menor valor no eixo Y.

2.2. Algoritmo de faixas

O algoritmo das faixas divide o polígono em faixas horizontais. Para implementar algoritmo, utilizamos uma lista de objetos do tipo faixa, onde cada objeto faixa armazena uma lista de arestas, estas correspondem a arestas do polígono.

Após estipular um número de faixas e a distância entre elas, iteramos sobre cada aresta do polígono e a adicionamos na lista de arestas da(s) faixa(s) que a aresta pertence.

	200 pontos		
Algoritmo	Tempo (s)	Ha intersec	Prod vetorial
Convex Hull + força bruta	0	167937	5042
Convex Hull + força bruta faixas	0	21917	5080
força bruta	0,015	436200	0
força bruta faixas	0	46386	0
	2000 pontos		
Algoritmo	Tempo (s)	Ha intersec	Prod vetorial
Convex Hull + força bruta	0,093	1583406	48514
Convex Hull + força bruta faixas	0,015	213190	48364
força bruta	0,219	4362000	0
força bruta faixas	0,031	432270	0
	20000 pontos		
Algoritmo	Tempo (s)	Ha intersec	Prod vetorial
Convex Hull + força bruta	0,844	16189563	486421
Convex Hull + força bruta faixas	0,2	2113259	482542
força bruta	2,172	43620000	0
força bruta faixas	0.312	4372506	0

Tabela 1. Comparação de algoritmos de inclusão

2.3. Algoritmo de detecção de intersecção

O algoritmo que verifica se há intersecção por força bruta, simplesmente testa se um ponto gera intersecção com todas os vértices do polígono, sem exceção. O algoritmo que verifica intersecção por meio das faixas, testa o ponto apenas com as arestas do polígono que estão na mesma faixa que ele, para registrar as arestas que passam em cada faixa foi criado um *vector* de objetos faixas que contem uma lista de objetos Aresta.

3. Conclusão

Ao realizar as comparações dos algoritmos pode-se ver claramente que usar o *Convex Hull* juntamente com o algoritmo das faixas é a solução mais otimizada para desempenho, utilizando menos cada função e levando menos tempo para processar os pontos.