

Triangulação de Delaunay com Quad-Edge

Geometria Computacional

Leonardo Seperuelo Duarte

Instituto Nacional de Matemática Pura e Aplicada
Departamento de Computação Gráfica - IMPA

26 de novembro de 2008

Tópicos

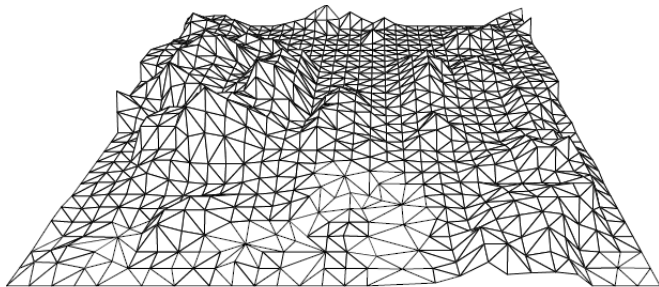
- 1 **Introdução**
 - Motivação
 - Definição
 - Primitivas Geométricas
- 2 **Quad-Edge**
 - Estrutura
 - Operadores Topológicos
 - Dualidade
- 3 **Algoritmo**
 - Divisão e Conquista
- 4 **Resultado**
- 5 **Conclusão**

Tópicos

- 1 **Introdução**
 - **Motivação**
 - Definição
 - Primitivas Geométricas
- 2 Quad-Edge
 - Estrutura
 - Operadores Topológicos
 - Dualidade
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

Motivação

- Interpolação de dados discretos para aproximação de terrenos.



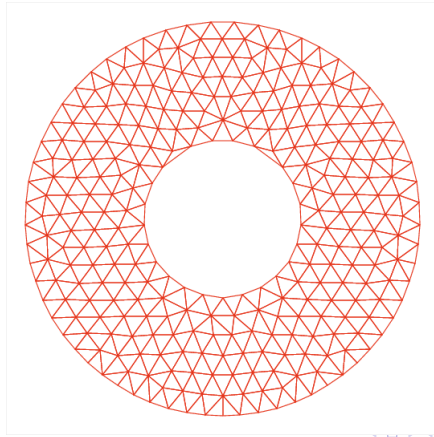
Motivação

- Localização de centros de serviços.



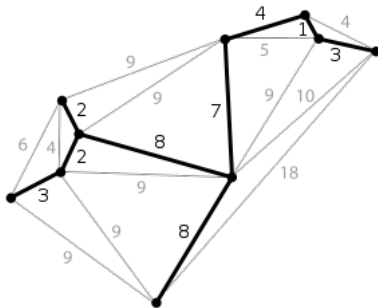
Motivação

- Construção de malhas para aplicação de FEM.



Motivação

- Obtenção direta de resultados como, EMST, PMP, GG e o diagrama de Voronoi.

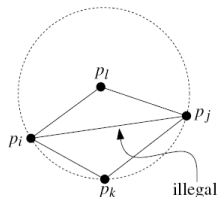
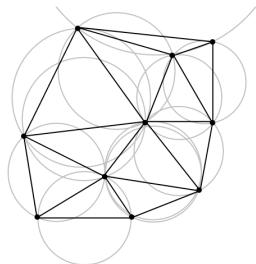


Tópicos

- 1 **Introdução**
 - Motivação
 - **Definição**
 - Primitivas Geométricas
- 2 Quad-Edge
 - Estrutura
 - Operadores Topológicos
 - Dualidade
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

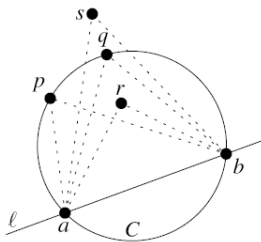
Triangulação de Delaunay (DT)

Para um conjunto S de pontos no plano, $DT(S)$ garante que o circuncírculo de qualquer face de $DT(S)$ não possui pontos em seu interior.



Triangulação de Delaunay (DT)

DT(S) maximiza o ângulo mínimo entre os ângulos dos triângulos de delaunay, evitando assim, triângulos finos, que não são ideais para aplicações de interpolação.



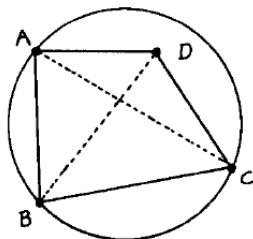
$$\angle arb > \angle apb = \angle aqb > \angle asb.$$

Tópicos

- 1 **Introdução**
 - Motivação
 - Definição
 - **Primitivas Geométricas**
- 2 Quad-Edge
 - Estrutura
 - Operadores Topológicos
 - Dualidade
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

Teste InCircle

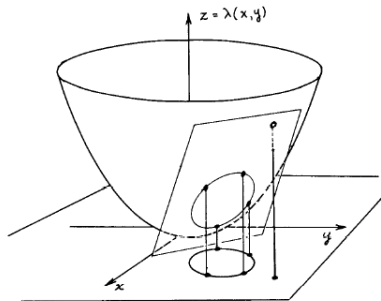
- Principal primitiva geométrica para o cálculo de DT(S).
- $\text{InCircle}(A, B, C, D)$ é verdadeiro \Leftrightarrow o ponto D está no interior do círculo orientado definido por ABC.



Teste InCircle

- $\text{InCircle}(A, B, C, D)$ é equivalente ao determinante abaixo.
- Pontos abaixo do plano, que intercepta o parabolóide, são projetados no interior do círculo x - y em R^2 .

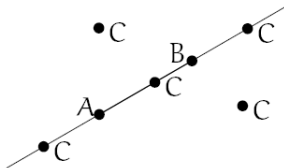
$$\mathcal{D}(A, B, C, D) = \begin{vmatrix} x_A & y_A & x_A^2 + y_A^2 & 1 \\ x_B & y_B & x_B^2 + y_B^2 & 1 \\ x_C & y_C & x_C^2 + y_C^2 & 1 \\ x_D & y_D & x_D^2 + y_D^2 & 1 \end{vmatrix} > 0.$$



Teste Counterclockwise (CCW)

- $CCW(A, B, C)$ determina se o ponto C está à esquerda do segmento AB .
- Para pontos colineares o resultado é falso.

$$\begin{vmatrix} x_A & y_A & 1 \\ x_B & y_B & 1 \\ x_C & y_C & 1 \end{vmatrix} > 0$$



Tópicos

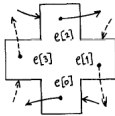
- 1 Introdução
 - Motivação
 - Definição
 - Primitivas Geométricas
- 2 Quad-Edge
 - **Estrutura**
 - Operadores Topológicos
 - Dualidade
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

Estruturas para DT

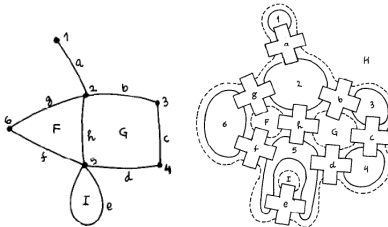
- Lista de polígonos
 - Cada um com as coordenadas de todos os seus vértices.
 - Redundância de informações e ineficiente na busca pela vizinhança.
- Lista de vértices e faces
 - Vértices armazenados separadamente e faces apontando para os vértices.
 - Diminui a redundância, porém a vizinhança de um vértice ainda tem que consultar toda a lista de faces.
- As duas estruturas acima precisam de informações topológicas de adjacências.

Estrutura da Quad-Edge

- Cada aresta formada por 4 componentes.



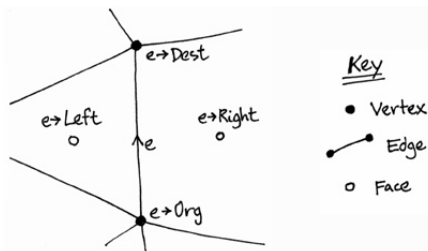
- Seus vértices extremos e suas faces adjacentes.



Tópicos

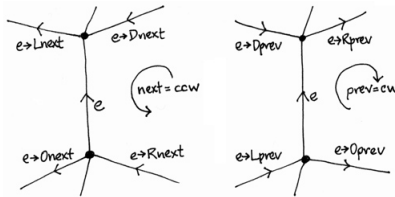
- 1 Introdução
 - Motivação
 - Definição
 - Primitivas Geométricas
- 2 **Quad-Edge**
 - Estrutura
 - **Operadores Topológicos**
 - Dualidade
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

Operadores Topológicos



- $e \rightarrow \text{Org}()$ retorna o vértice origem da aresta e .
- $e \rightarrow \text{Dest}()$ retorna o vértice destino da aresta e .
- $e \rightarrow \text{Left}()$ retorna a face à esquerda da aresta e .
- $e \rightarrow \text{Right}()$ retorna a face à direita da aresta e .
- Todas as operações com custo $O(1)$.

Operadores Topológicos



- Sentido CCW

- Lnext()/Rnext() aresta com mesma face à esquerda/direita.
- Onext()/Dnext() aresta com mesmo vértice origem/destino.

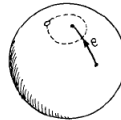
- Sentido CW

- Lprev()/Rprev() aresta com mesma face à esquerda/direita.
- Oprev()/Dprev() aresta com mesmo vértice origem/destino.

Operadores Topológicos

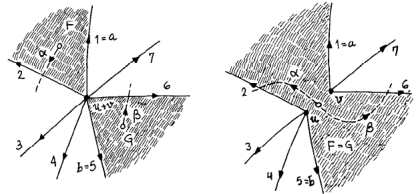
- MakeEdge()

- $Org \leftarrow s1$ e $Dest \leftarrow s2$.
- $Left \leftarrow face$ e $Right \leftarrow face$.



- Splice(Edge* a, Edge* b)

- Se $a \rightarrow Org = b \rightarrow Org$ separa em dois pedaços.
- Se $a \rightarrow Org \neq b \rightarrow Org$ combina em apenas um.



Tópicos

- 1 Introdução
 - Motivação
 - Definição
 - Primitivas Geométricas
- 2 Quad-Edge
 - Estrutura
 - Operadores Topológicos
 - **Dualidade**
- 3 Algoritmo
 - Divisão e Conquista
- 4 Resultado
- 5 Conclusão

Diagrama de Voronoi

- Dual geométrico do Diagrama de Delaunay.
 - Faces de Voronoi \leftrightarrow Vértices de Delaunay.
 - Vértices de Voronoi \leftrightarrow Faces de Delaunay.
 - Arestas de Voronoi \leftrightarrow Arestas de Delaunay.

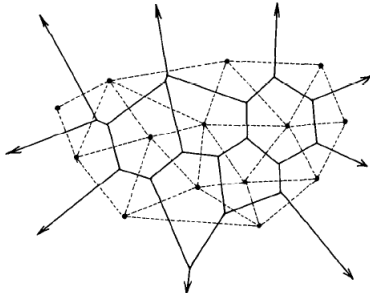
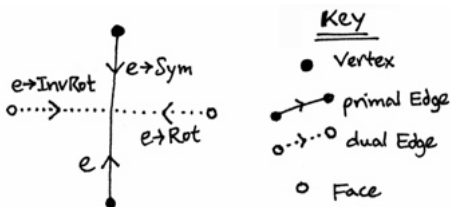


Diagrama de Voronoi

- Estrutura já considera a dualidade.
- Cálculo de Voronoi junto com o cálculo de Delaunay.
 - $e \rightarrow \text{Sym}()$ aresta no sentido oposto.
 - $e \rightarrow \text{Rot}()$ aresta dual da direita para a esquerda.
 - $e \rightarrow \text{InvRot}()$ aresta dual da esquerda para a direita.



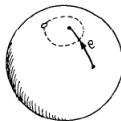
Tópicos

- 1 Introdução
 - Motivação
 - Definição
 - Primitivas Geométricas
- 2 Quad-Edge
 - Estrutura
 - Operadores Topológicos
 - Dualidade
- 3 Algoritmo**
 - Divisão e Conquista**
- 4 Resultado
- 5 Conclusão

Caso Base

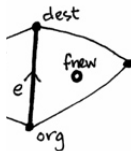
- Ordenação lexicográfica.

- $S = 2$, apenas MakeEdge().



- $S = 3$.

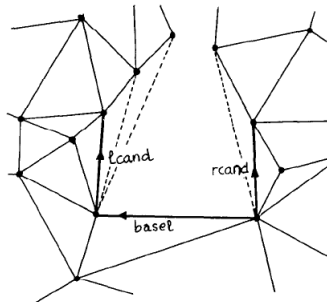
- $a \leftarrow \text{MakeEdge}()$.
- $b \leftarrow \text{MakeEdge}()$.
- $c \leftarrow \text{Connect}(b, a)$.
- Caso CCW(a, b, c), nova face à esquerda.



- $S > 3$, chama o algoritmo para $\frac{S}{2}$ e $\frac{S}{2}$

Merge

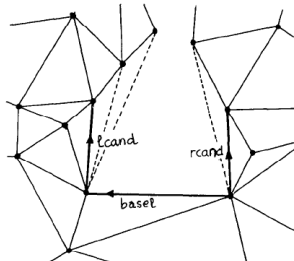
- $[ldo, ldi] \leftarrow \text{delaunay}(L)$
- $[rdi, rdo] \leftarrow \text{delaunay}(R)$
- Encontrar a aresta *basel*.
- $\text{basel} \leftarrow \text{Connect}(\text{rdi} \rightarrow \text{Sym}, \text{ldi})$.



Merge

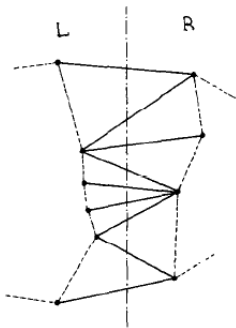
- *Merge loop*

- *lcand* é válida \rightarrow verifica *InCircle* para *lcand* e *basel*.
- Caso não respeite, *lcand* e sua *Left* são destruídas.
- Os passos feitos para *lcand* são feitos para *rcand*.
- Caso *lcand* e *rcand* sobrevivam, *InCircle* decide:
 - *Connect(rcand, basel \rightarrow Sym)*
 - *Connect(basel \rightarrow Sym, lcand \rightarrow Sym)*

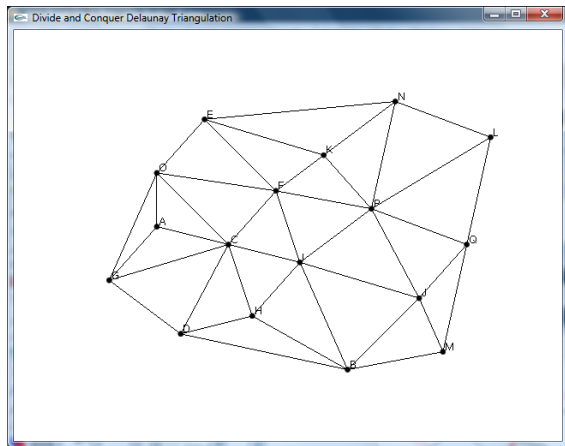


Final

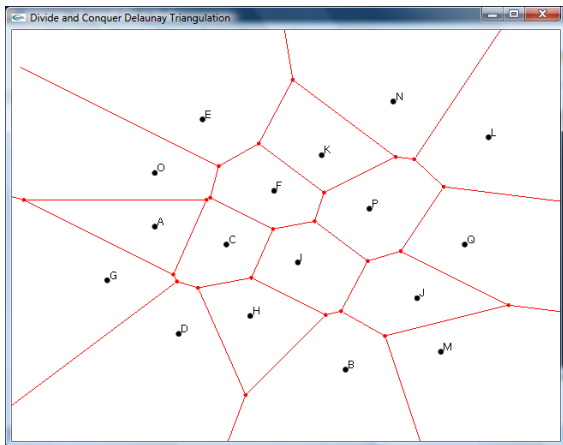
- Quando *lcand* e *rcand* não são mais válidas.
- O algoritmo termina retornando *le* e *re*



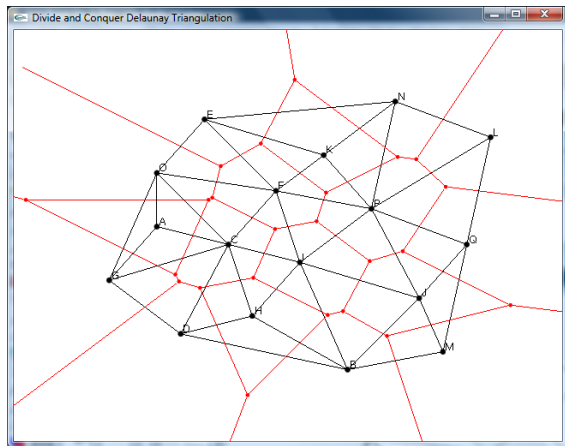
Delaunay



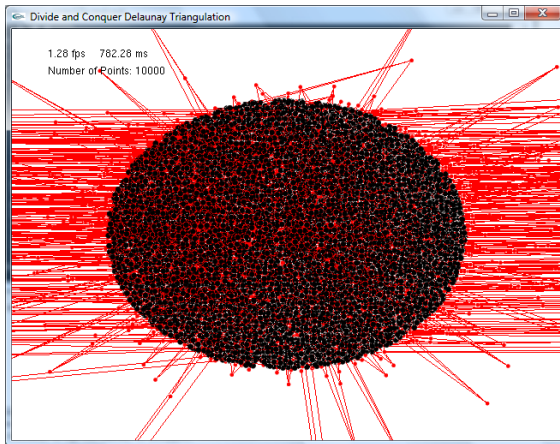
Voronoi



Resultado



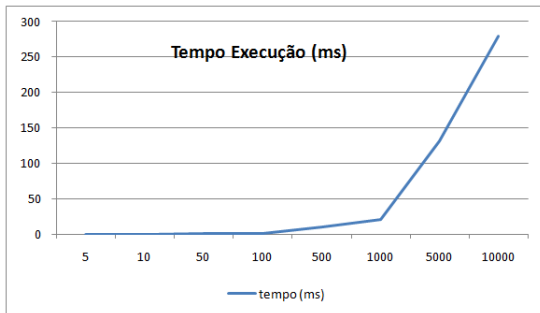
Resultado



Tempo de Execução

- Complexidade da $O(n \log(n))$.

Divisão e Conquista								
# pontos	5	10	50	100	500	1000	5000	10000
tempo (ms)	0,07838	0,1521	0,8946	1,7186	10,582	20,751	130,91	278,77



Conclusão

- Estrutura Quad-Edge de grande utilidade, armazenando o dual.
- Possibilidade de cálculo de Delaunay e Voronoi juntos.
- Complexidade $O(n \log(n))$.
- Informações topológicas em $O(1)$.
- Algoritmo de divisão e conquista fica compacto e simples.
- Maior dificuldade em entender e utilizar a estrutura topológica Quad-Edge.