

Diagrama de Voronoi

INF2604 – Geometria Computacional

Waldemar Celes
celes@inf.puc-rio.br

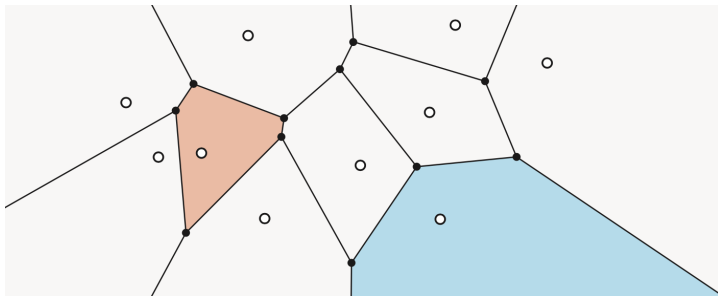
Departamento de Informática, PUC-Rio



Diagrama de Voronoi

Determinação de proximidade

- Regiões de Voronoi delimitam áreas cujo ponto associado é mais próximo



Dado um conjunto de pontos S , a Região de Voronoi de um ponto $\mathbf{p} \in S$

$$\text{Vor}(\mathbf{p}) = \{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{p}\| \leq \|\mathbf{x} - \mathbf{q}\| \quad \forall \quad \mathbf{q} \in S \mid \mathbf{q} \neq \mathbf{p}\}$$

Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011

Diagrama de Voronoi

Diagrama considerando apenas dois pontos **p** e **q**

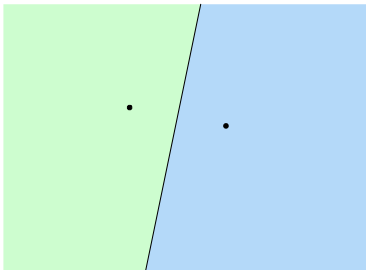


Diagrama de Voronoi

Diagrama considerando apenas dois pontos **p** e **q**

- ▶ Regiões são semiplanos

$$Vor(\mathbf{p}) = H(\mathbf{p}, \mathbf{q}) = \mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{p}\| \leq \|\mathbf{x} - \mathbf{q}\|$$



Regiões de Voronoi

Teorema

- ▶ A região de Voronoi $Vor(\mathbf{p})$ é a interseção de todos os semiplanos $H(\mathbf{p}, \mathbf{q})$, onde \mathbf{q} é qualquer outro ponto de S .



Regiões de Voronoi

Teorema

- ▶ A região de Voronoi $Vor(\mathbf{p})$ é a interseção de todos os semiplanos $H(\mathbf{p}, \mathbf{q})$, onde \mathbf{q} é qualquer outro ponto de S .

Teorema fundamental da geometria discreta

- ▶ A interseção de um conjunto de objetos convexos é um conjunto convexo



Regiões de Voronoi

Teorema

- ▶ A região de Voronoi $Vor(\mathbf{p})$ é a interseção de todos os semiplanos $H(\mathbf{p}, \mathbf{q})$, onde \mathbf{q} é qualquer outro ponto de S .

Teorema fundamental da geometria discreta

- ▶ A interseção de um conjunto de objetos convexos é um conjunto convexo
 - ▶ Prova: considere X_i um conjunto de objetos convexos e X a interseção desses conjuntos. Considere pontos arbitrários \mathbf{p} e \mathbf{q} de X . Por definição de interseção, estes pontos também pertencem a qualquer X_i e, como são convexo, $\overline{\mathbf{p}\mathbf{q}}$ também pertence a X_i e, consequentemente, a X .



Regiões de Voronoi

Teorema

- ▶ A região de Voronoi $Vor(\mathbf{p})$ é a interseção de todos os semiplanos $H(\mathbf{p}, \mathbf{q})$, onde \mathbf{q} é qualquer outro ponto de S .

Teorema fundamental da geometria discreta

- ▶ A interseção de um conjunto de objetos convexos é um conjunto convexo
 - ▶ Prova: considere X_i um conjunto de objetos convexos e X a interseção desses conjuntos. Considere pontos arbitrários \mathbf{p} e \mathbf{q} de X . Por definição de interseção, estes pontos também pertencem a qualquer X_i e, como são convexo, $\overline{\mathbf{p}\mathbf{q}}$ também pertence a X_i e, conseqüentemente, a X .

Teorema

- ▶ Todas as regiões de Voronoi são convexas



Vértices de Voronoi

Considere 3 pontos \mathbf{p} , \mathbf{q} , \mathbf{r}

- ▶ Qual o diagrama de Voronoi?



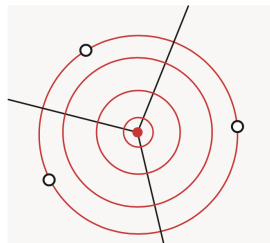
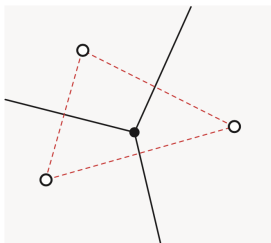
Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011



Vértices de Voronoi

Considere 3 pontos p , q , r

- ▶ Qual o diagrama de Voronoi?
 - ▶ Formado pelas linhas perpendiculares bissetoras dos segmentos \overline{pq} , \overline{qr} , \overline{rp}
 - ▶ Necessariamente se encontram em um único ponto
 - ▶ Este ponto é o centro do círculo que passa pelos 3 pontos (encontro das mediatrizes define o circuncentro do triângulo)



Vértices de Voronoi

Diagrama de Voronoi com 4 pontos

- Pontos cocirculares ou não

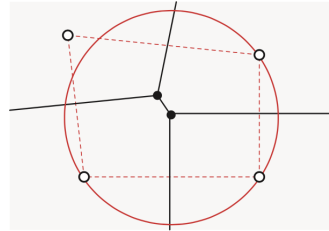
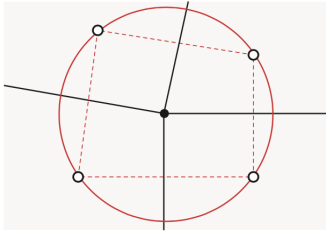


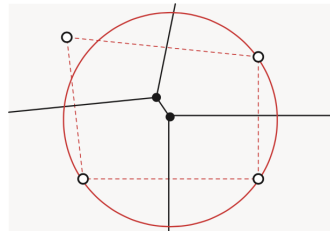
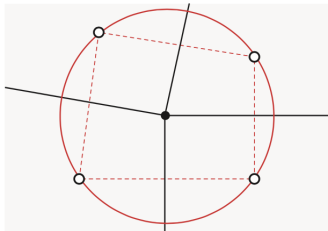
Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011



Vértices de Voronoi

Diagrama de Voronoi com 4 pontos

- ▶ Pontos cocirculares ou não



Conjunto de pontos não degenerados

- ▶ Não existem 4 ou mais vértices cocirculares
- ▶ Todos os vértices de Voronoi tem **grau 3**

Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011

Vértices de Voronoi

Teorema

- ▶ Dado um conjunto de pontos S e seu respectivo diagrama de Voronoi $Vor(S)$, um ponto v é vértice de Voronoi sse existir um círculo com centro em v que passa por 3 pontos de S , e não existe nenhum outro ponto de S interior a este círculo.



Vértices de Voronoi

Teorema

- ▶ Dado um conjunto de pontos S e seu respectivo diagrama de Voronoi $Vor(S)$, um ponto v é vértice de Voronoi sse existir um círculo com centro em v que passa por 3 pontos de S , e não existe nenhum outro ponto de S interior a este círculo.

Exercício

- ▶ Para $n \geq 3$:
 - ▶ Quando o diagrama de Voronoi teria **zero** vértices?



Vértices de Voronoi

Teorema

- ▶ Dado um conjunto de pontos S e seu respectivo diagrama de Voronoi $Vor(S)$, um ponto v é vértice de Voronoi sse existir um círculo com centro em v que passa por 3 pontos de S , e não existe nenhum outro ponto de S interior a este círculo.

Exercício

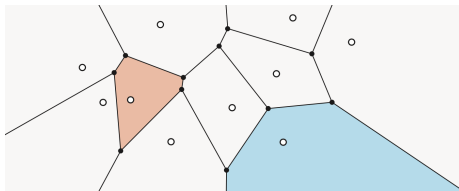
- ▶ Para $n \geq 3$:
 - ▶ Quando o diagrama de Voronoi teria **zero** vértices?
 - ▶ Quando o diagrama de Voronoi teria **um** vértice?



Arestas de Voronoi

Tipos de arestas

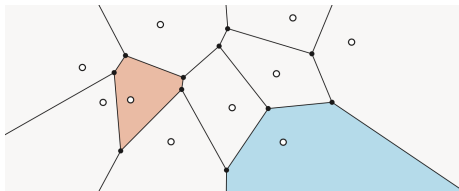
- ▶ Finitas
- ▶ Semiinfinitas



Arestas de Voronoi

Tipos de arestas

- ▶ Finitas
- ▶ Semiinfinitas



Quando todos pontos são colineares

- ▶ Arestas são infinitas
- ▶ Arestas são desconectadas

Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011

Diagrama de Voronoi

Relação linear entre número de vértices e número de arestas

- ▶ Dados S com $n \geq 3$ pontos, o diagrama de Voronoi de S , $Vor(S)$:
 - ▶ Tem no máximo $2n - 5$ vértices
 - ▶ Tem no máximo $3n - 6$ arestas



Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011



Diagrama de Voronoi

Relação linear entre número de vértices e número de arestas

- ▶ Dados S com $n \geq 3$ pontos, o diagrama de Voronoi de S , $Vor(S)$:
 - ▶ Tem no máximo $2n - 5$ vértices
 - ▶ Tem no máximo $3n - 6$ arestas
 - ▶ Prova:
 - ▶ Similar a grafo planar onde vértice no infinito é representado

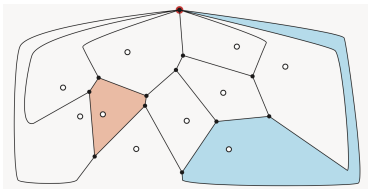


Diagrama de Voronoi

Prova:

- ▶ Fórmula de Euler:

$$V - E + F = 2$$
$$(v + 1) - e + n = 2$$

- ▶ Grau dos vértices: ≥ 3

$$3(v + 1) \leq 2e$$

- ▶ Substituindo a Fórmula de Euler nesta inequação
 - ▶ Vértices

$$3(v + 1) \leq 2(n + v + 1 - 2) \quad \therefore \quad v \leq 2n - 5$$

- ▶ Arestas

$$3(2 + e - n) \leq 2e \quad \therefore \quad e \leq 3n - 6$$



Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Dado um diagrama inicial
- ▶ Acrescenta um ponto **p** no diagrama
 - ▶ Localiza a região de Voronoi $Vor(p_1)$ que contém **p**
 - ▶ Acha bissetora de $\overline{pp_1}$
 - ▶ Acha interseção com arestas da região: x_1 e x_2
 - ▶ Acha bissetora de $\overline{pp_2}$
 - ▶ Até fechar região $Vor(p)$

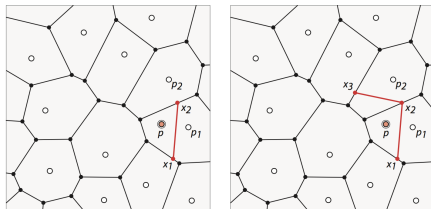


Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011

Construção do diagrama de Voronoi

Algoritmo incremental

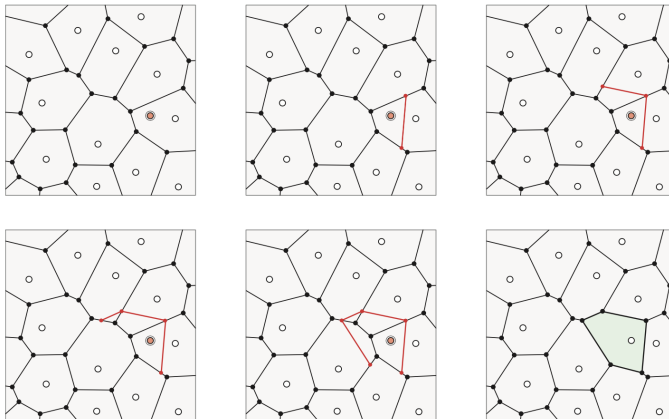


Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011

Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Complexidade $O(n^2)$



Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Complexidade $O(n^2)$
 - ▶ Busca da região que contém ponto **p** pode ser melhorada?



Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Complexidade $O(n^2)$
 - ▶ Busca da região que contém ponto **p** pode ser melhorada?
 - ▶ Sim, busca topológica fazendo inserções com coerência espacial



Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Complexidade $O(n^2)$
 - ▶ Busca da região que contém ponto **p** pode ser melhorada?
 - ▶ Sim, busca topológica fazendo inserções com coerência espacial
 - ▶ Construção de $Vor(\mathbf{p})$ é sempre local?



Construção do diagrama de Voronoi

Algoritmo incremental

- ▶ Complexidade $O(n^2)$
 - ▶ Busca da região que contém ponto **p** pode ser melhorada?
 - ▶ Sim, busca topológica fazendo inserções com coerência espacial
 - ▶ Construção de $Vor(\mathbf{p})$ é sempre local?
 - ▶ Provavelmente na prática (diminuindo a complexidade)
 - ▶ Contra-exemplo?



Diagrama de Voronoi

Grafo dual

- ▶ Arcos conectam pontos próximos
- ▶ Pontos cujas regiões de Voronoi compartilham uma aresta

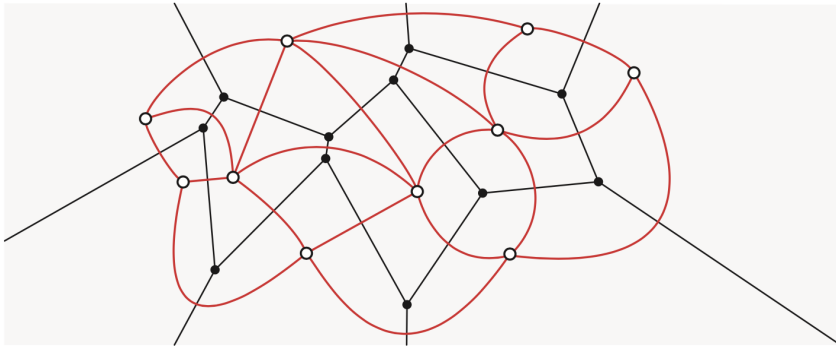


Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011

Diagrama de Voronoi

Retificação dos arcos em arestas conectando pontos

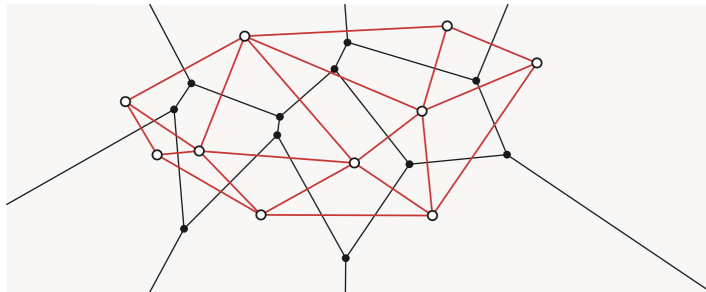
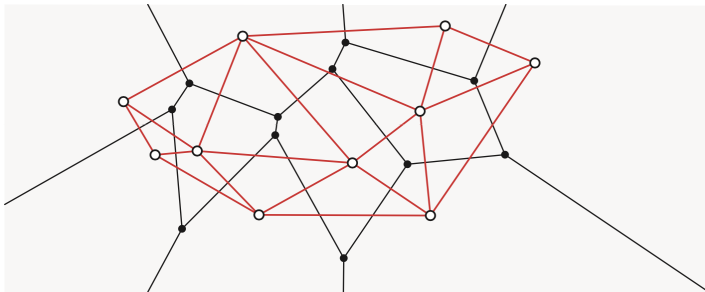


Diagrama de Voronoi

Retificação dos arcos em arestas conectando pontos



- ▶ Arestas não se cruzam
- ▶ Subdivisão resultante é uma triangulação de S
- ▶ Triangulação é de **Delaunay**

Figura extraída de Discrete and Computational Geometry, Devadoss and Rourke, 2011

Triangulação de Delaunay

Algoritmo incremental

- ▶ Explora dualidade com diagrama de Voronoi
 - ▶ Suponha uma triangulação de Delaunay $Del(S_k)$
 - ▶ Adiciona **p**
 - ▶ Marca triângulos cujo circuncírculo contém **p**
 - ▶ Elimina diagonais do polígono marcado
 - ▶ Insere arestas de **p** aos vértices desse polígono

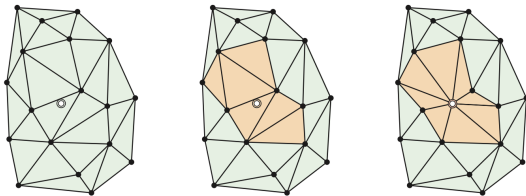
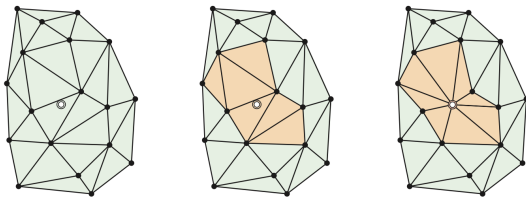


Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011

Triangulação de Delaunay

Algoritmo incremental

- ▶ Explora dualidade com diagrama de Voronoi
 - ▶ Suponha uma triangulação de Delaunay $Del(S_k)$
 - ▶ Adiciona \mathbf{p}
 - ▶ Marca triângulos cujo circuncírculo contém \mathbf{p}
 - ▶ Elimina diagonais do polígono marcado
 - ▶ Insere arestas de \mathbf{p} aos vértices desse polígono



- ▶ $Vor(\mathbf{p})$ é delimitado pelas bissetoras de cada ponto vizinho
- ▶ Arestas representam arcos retificados dessas bissetoras



Figura extraída de *Discrete and Computational Geometry*, Devadoss and Rourke, 2011



Diagrama de Voronoi Centroidal

CVT – Centroidal Voronoi Tessellation

- ▶ Pontos geradores \mathbf{p}_i representam **centros de massas** das correspondentes regiões de Voronoi $Vor(\mathbf{p}_i)$
 - ▶ Regiões tendem a ser hexaedros regulares

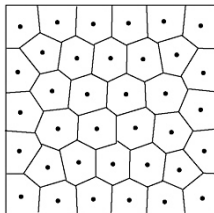


Diagrama de Voronoi Centroidal

Algoritmo de Lloyd

- ▶ Dado um conjunto inicial de geradores S
 - ▶ Gera o diagrama de Voronoi correspondente
- ▶ Itera até convergência
 - ▶ Move cada ponto gerador para o centro de massa da sua região
 - ▶ Regera o diagrama de Voronoi considerando os novos pontos

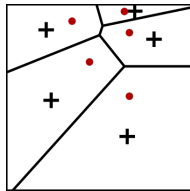


Diagrama de Voronoi Centroidal

Algoritmo de Lloyd

- ▶ Dado um conjunto inicial de geradores S
 - ▶ Gera o diagrama de Voronoi correspondente
- ▶ Itera até convergência
 - ▶ Move cada ponto gerador para o centro de massa da sua região
 - ▶ Regera o diagrama de Voronoi considerando os novos pontos

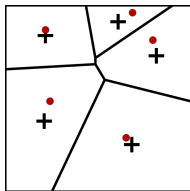
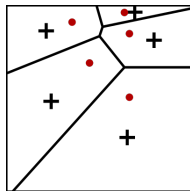


Diagrama de Voronoi Centroidal

Algoritmo de Lloyd

- ▶ Dado um conjunto inicial de geradores S
 - ▶ Gera o diagrama de Voronoi correspondente
- ▶ Itera até convergência
 - ▶ Move cada ponto gerador para o centro de massa da sua região
 - ▶ Regera o diagrama de Voronoi considerando os novos pontos

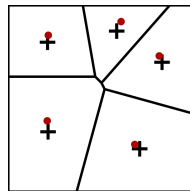
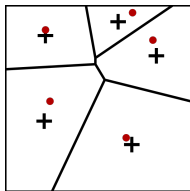
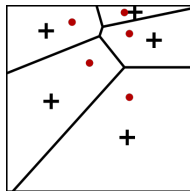


Diagrama de Voronoi Centroidal

Algoritmo de Lloyd

- ▶ Dado um conjunto inicial de geradores S
 - ▶ Gera o diagrama de Voronoi correspondente
- ▶ Itera até convergência
 - ▶ Move cada ponto gerador para o centro de massa da sua região
 - ▶ Regera o diagrama de Voronoi considerando os novos pontos

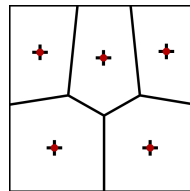
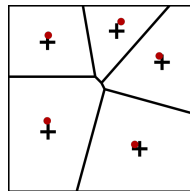
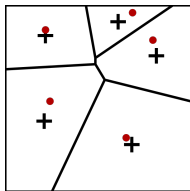
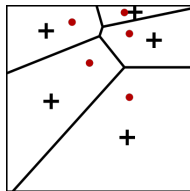


Diagrama de Voronoi discretizado

Diagrama de Voronoi em um domínio discretizado

- ▶ Exemplo: resolução de uma tela



Diagrama de Voronoi discretizado

Diagrama de Voronoi em um domínio discretizado

- ▶ Exemplo: resolução de uma tela

Métodos

- ▶ Construção por renderização
- ▶ Construção por dilatação iterativa



Diagrama de Voronoi discretizado

Construção por renderização de cones

- ▶ Vértices dos cones posicionados nos pontos geradores
- ▶ Vista ortográfica de cima
- ▶ Teste de visibilidade (*depth test*) garante corretude



Diagrama de Voronoi discretizado

Construção por renderização de cones

- ▶ Vértices dos cones posicionados nos pontos geradores
- ▶ Vista ortográfica de cima
- ▶ Teste de visibilidade (*depth test*) garante corretude

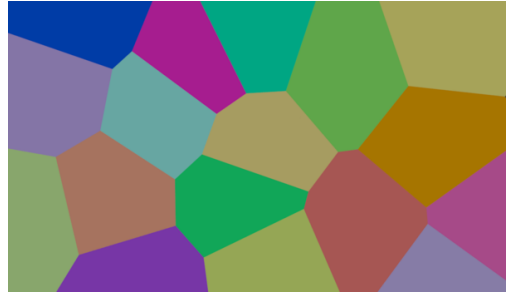


Diagrama de Voronoi discretizado

Construção por dilatação iterativa

- ▶ Cada ponto gerador é preenchido no domínio discreto
- ▶ A cada iteração, visita todos os pixels não preenchidos
 - ▶ Preenche pixel com atributo de vizinho preenchido, se houver
 - ▶ Repete o processo até que todos os pixels estejam preenchidos

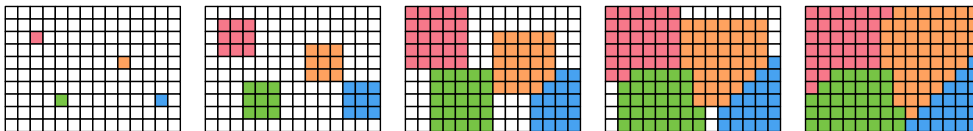
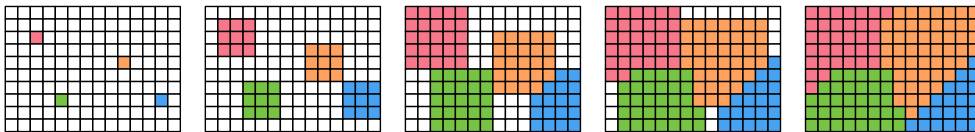


Diagrama de Voronoi discretizado

Construção por dilatação iterativa

- ▶ Cada ponto gerador é preenchido no domínio discreto
- ▶ A cada iteração, visita todos os pixels não preenchidos
 - ▶ Preenche pixel com atributo de vizinho preenchido, se houver
 - ▶ Repete o processo até que todos os pixels estejam preenchidos



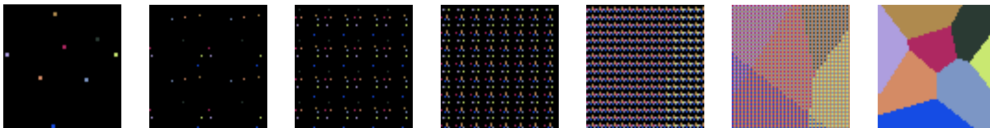
$O(n)$ iterações

- ▶ Números de iterações proporcional ao número de pixels

Algoritmo *jump flooding*

Propagação completa com $\log n$ iterações

- ▶ Mas pode apresentar erros
 - ▶ Pequenos e contornáveis na maior parte dos casos



Algoritmo *jump flooding*

Preenchimento convencional

$$(x, y) \longrightarrow (x + i, y + j) \quad | \quad i, j \in \{-1, 0, 1\}$$



Algoritmo *jump flooding*

Preenchimento convencional

$$(x, y) \longrightarrow (x + i, y + j) \quad | \quad i, j \in \{-1, 0, 1\}$$

Preenchimento com saltos (*jumps*)

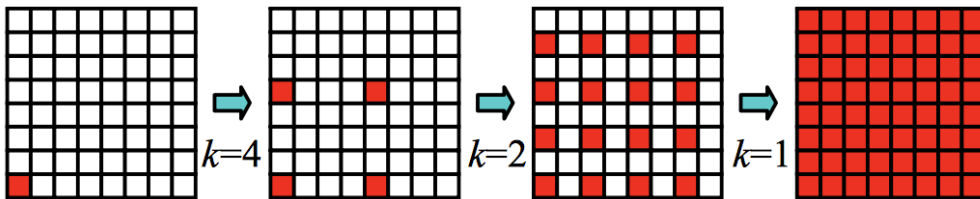
$$(x, y) \longrightarrow (x + i, y + j) \quad | \quad i, j \in \{-k, 0, k\}$$



Algoritmo *jump flooding*

Estratégia

- ▶ Reduzir o passo à metade a cada iteração
 - ▶ Cada pixel recebe informações de 8 outras localizações
 - ▶ Mais a sua própria informação
 - ▶ Cada pixel armazena a informação de menor distância



Algoritmo *jump flooding*

Algoritmo

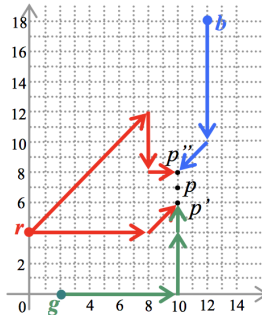
- ▶ Início
 - ▶ Semente com informações $[s, pos(s)]$
 - ▶ Não semente com informações $[nil, nil]$
- ▶ A cada iteração
 - ▶ Informações dos pixels são atualizadas
 - ▶ Escolhe-se a informação da menor distância



Algoritmo *jump flooding*

Possíveis erros

- ▶ No exemplo, p deveria ser r
 - ▶ Mas p' e p'' não tem r como menor distância



Algoritmo *jump flooding*

Variações visando reduzir erros

- ▶ $JFA + 1$
 - ▶ Uma passada extra com passo igual a 1 no final



Algoritmo *jump flooding*

Variações visando reduzir erros

- ▶ $JFA + 1$
 - ▶ Uma passada extra com passo igual a 1 no final
- ▶ $JFA + 2$
 - ▶ Duas passadas extras no final com passos 2 e 1



Algoritmo *jump flooding*

Variações visando reduzir erros

- ▶ $JFA + 1$
 - ▶ Uma passada extra com passo igual a 1 no final
- ▶ $JFA + 2$
 - ▶ Duas passadas extras no final com passos 2 e 1
- ▶ $JFA + \log n$ ou JFA^2
 - ▶ $\log n$ passadas adicionais



Algoritmo *jump flooding*

Variações visando reduzir erros

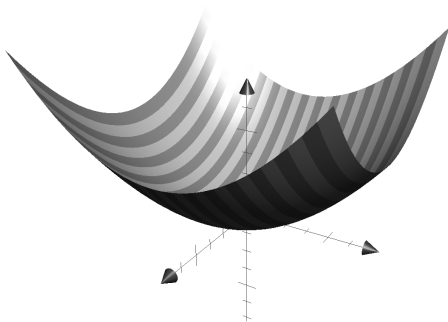
- ▶ $JFA + 1$
 - ▶ Uma passada extra com passo igual a 1 no final
- ▶ $JFA + 2$
 - ▶ Duas passadas extras no final com passos 2 e 1
- ▶ $JFA + \log n$ ou JFA^2
 - ▶ $\log n$ passadas adicionais
- ▶ $JFA + 2$ sementes (ou $+n$ sementes)
 - ▶ Armazena os 2 (ou n) mais próximos
 - ▶ Aumenta uso de memória
 - ▶ Aumenta processamento para escolha do mínimo



Relação Delaunay e Fecho Convexo 3D

Equação de um parabolóide

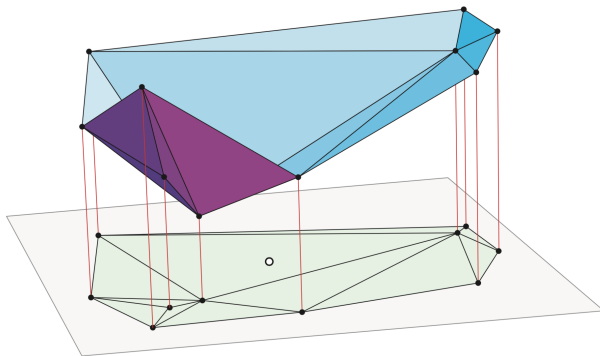
$$z = x^2 + y^2$$



Relação Delaunay e Fecho Convexo 3D

Teorema

- Dado um conjunto de pontos S no plano xy , não cocirculares, a triangulação de Delaunay $Del(S)$ é exatamente a projeção no plano xy das faces inferiores do fecho convexo dos pontos $(\mathbf{x}_i, \mathbf{y}_i, \mathbf{x}_i^2 + \mathbf{y}_i^2)$.



Relação Delaunay e Fecho Convexo 3D

Complexidade computacional

- ▶ Determinação de fecho convexo 3D: $O(n \log n)$
 - ▶ Consequentemente, triangulação de Delaunay 2D: $O(n \log n)$



Relação Delaunay e Fecho Convexo 3D

Complexidade computacional

- ▶ Determinação de fecho convexo 3D: $O(n \log n)$
 - ▶ Consequentemente, triangulação de Delaunay 2D: $O(n \log n)$
-

Teorema se aplica para dimensões maiores

- ▶ A triangulação de Delaunay 3D pode ser obtida pela projeção no espaço xyz do fecho convexo 4D dos pontos $(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i, \mathbf{x}_i^2 + \mathbf{y}_i^2 + \mathbf{z}_i^2)$.