# Círculo Mínimo

## INF2604 – Geometria Computacional

Waldemar Celes
celes@inf.puc-rio.br

Departamento de Informática, PUC-Rio

# Determinação de círculo mínimo

Problema

- Dado um conjunto de pontos no plano $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$, achar o **círculo mínimo** envolvente
  - Determinar centro $\mathbf{c}$ e raio $r$ do círculo

# Determinação de círculo mínimo

Problema

- Dado um conjunto de pontos no plano $P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$, achar o **círculo mínimo** envolvente
  - Determinar centro $\mathbf{c}$ e raio $r$ do círculo

Casos especiais:

- $n = 1$: $\mathbf{c} = \mathbf{p}_1$ e $r = 0$

# Determinação de círculo mínimo

Problema

- Dado um conjunto de pontos no plano $P = \{\mathbf{p}_1, ..., \mathbf{p}_n\}$, achar o **círculo mínimo** envolvente
  - Determinar centro $\mathbf{c}$ e raio $r$ do círculo

Casos especiais:

- $n = 1$: $\mathbf{c} = \mathbf{p}_1$ e $r = 0$
- $n = 2$: $\mathbf{c} = \frac{\mathbf{p}_1 + \mathbf{p}_2}{2}$ e $r = \frac{\|\mathbf{p}_1 - \mathbf{p}_2\|}{2}$

# Determinação de círculo mínimo

Caso $n = 3$:

*L.H. de Figueiredo, P.C.P. Carvalho, "Notas de Geometria Computacional", IMPA*

*Wikipedia, "Pontos, linhas e círculos associados a um triângulo"*

# Determinação de círculo mínimo

Caso $n = 3$:

- Pontos colineares
  - Recai no caso $n = 2$

---

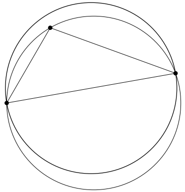
*L.H. de Figueiredo, P.C.P. Carvalho, "Notas de Geometria Computacional", IMPA*

*Wikipedia, "Pontos, linhas e círculos associados a um triângulo"*

# Determinação de círculo mínimo

Caso $n = 3$:

- Pontos colineares
  - Recai no caso $n = 2$
- Pontos não colineares
  - Com ângulo obtuso: recai no caso $n = 2$



*L.H. de Figueiredo, P.C.P. Carvalho, "Notas de Geometria Computacional", IMPA*

*Wikipedia, "Pontos, linhas e círculos associados a um triângulo"*

# Determinação de círculo mínimo

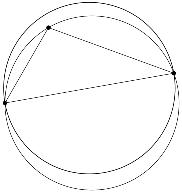
Caso $n = 3$:

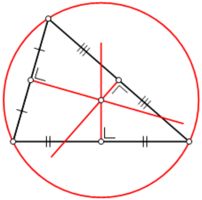
- Pontos colineares
  - Recai no caso $n = 2$
- Pontos não colineares
  - Com ângulo obtuso: recai no caso $n = 2$
  - Sem ângulo obtuso: circuncírculo de triângulo
    - Determinação de **c**: encontro das mediatrizes

$$\|\mathbf{p}_1 - \mathbf{c}\|^2 = \|\mathbf{p}_2 - \mathbf{c}\|^2 = \|\mathbf{p}_3 - \mathbf{c}\|^2$$

    - Determinação de $r$

$$r = \|\mathbf{c} - \mathbf{p}_i\|, \quad \text{com} \quad i = 1, 2, \text{ou} 3$$





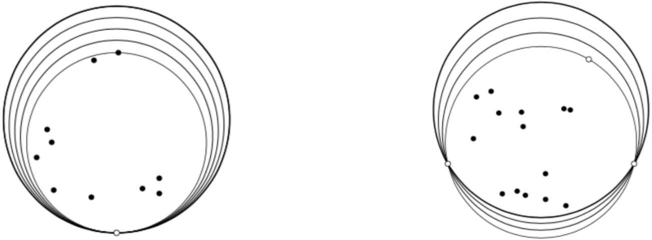
*L.H. de Figueiredo, P.C.P. Carvalho, "Notas de Geometria Computacional", IMPA*

*Wikipedia, "Pontos, linhas e círculos associados a um triângulo"*

# Determinação de círculo mínimo

Problema geral

- O círculo mínimo que envolve $P = \{\mathbf{p}_1, ..., \mathbf{p}_n\}$, com $n > 1$, tem, obrigatoriamente, 2 ou 3 pontos de contato com P
  - Dois pontos de $P$ diametralmente opostos, ou
  - Três pontos de $P$ formando um triângulo agudo



L.H. de Figueiredo, P.C.P. Carvalho, "Notas de Geometria Computacional", IMPA

# Determinação de círculo mínimo

Algoritmo força bruta

- Considere todas os pares de pontos
  - Verifique se círculo diametral envolve todos os demais pontos
  - Guarde o menor círculo
- Considere todas as triplas de pontos
  - Verifique se circuncírculo do triângulo envolve todos os demais pontos
  - Guarde o menor círculo

# Determinação de círculo mínimo

Algoritmo força bruta

- Considere todas os pares de pontos
  - Verifique se círculo diametral envolve todos os demais pontos
  - Guarde o menor círculo
- Considere todas as triplas de pontos
  - Verifique se circuncírculo do triângulo envolve todos os demais pontos
  - Guarde o menor círculo

Tempo esperado

# Determinação de círculo mínimo

Algoritmo força bruta

- Considere todas os pares de pontos
  - Verifique se círculo diametral envolve todos os demais pontos
  - Guarde o menor círculo
- Considere todas as triplas de pontos
  - Verifique se circuncírculo do triângulo envolve todos os demais pontos
  - Guarde o menor círculo

Tempo esperado

$$n\left(\binom{n}{2} + \binom{n}{3}\right) = O(n^4)$$

Lembrando que:

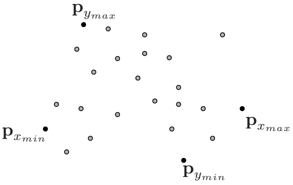
$$\binom{n}{k} = \frac{n!}{(n-k)!k!}$$

# Determinação de círculo envolvente

Algoritmo baseado em heurística

- Não resulta no círculo mínimo
- Usado como forma simples de determinar círculo envolvente

# Determinação de círculo envolvente




Algoritmo baseado em heurística

- Não resulta no círculo mínimo
- Usado como forma simples de determinar círculo envolvente

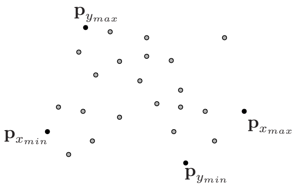
Algoritmo

- Ache os dois pares de pontos: $\{\mathbf{p}_{x_{min}}, \mathbf{p}_{x_{max}}\}$, $\{\mathbf{p}_{y_{min}}, \mathbf{p}_{y_{max}}\}$

# Determinação de círculo envolvente

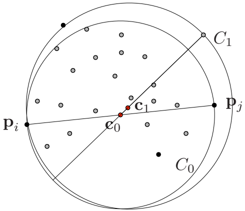
Algoritmo baseado em heurística

- Não resulta no círculo mínimo
- Usado como forma simples de determinar círculo envolvente




Algoritmo

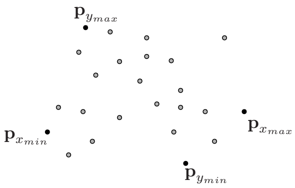
- Ache os dois pares de pontos: $\{\mathbf{p}_{x_{min}}, \mathbf{p}_{x_{max}}\}$, $\{\mathbf{p}_{y_{min}}, \mathbf{p}_{y_{max}}\}$
- Escolha par mais distante: $\mathbf{p}_i, \mathbf{p}_j$
- Considere o círculo diametral: $\mathbf{c} = \frac{\mathbf{p}_i + \mathbf{p}_j}{2}$ e $r = \frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{2}$
- Para cada ponto $\mathbf{p}_k$, faz $\vec{d} = \mathbf{p}_k - \mathbf{c}$; se $\|d\| > r$:
  - $\mathbf{c} = \mathbf{c} + \frac{\|d\| - r}{2}\hat{d}$ e $r = \frac{\|d\| + r}{2}$



# Determinação de círculo envolvente

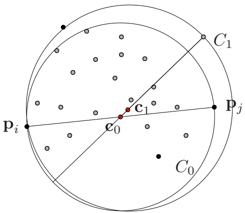
Algoritmo baseado em heurística

- Não resulta no círculo mínimo
- Usado como forma simples de determinar círculo envolvente




Algoritmo

- Ache os dois pares de pontos: $\{\mathbf{p}_{x_{min}}, \mathbf{p}_{x_{max}}\}$, $\{\mathbf{p}_{y_{min}}, \mathbf{p}_{y_{max}}\}$
- Escolha par mais distante: $\mathbf{p}_i, \mathbf{p}_j$
- Considere o círculo diametral: $\mathbf{c} = \frac{\mathbf{p}_i + \mathbf{p}_j}{2}$ e $r = \frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{2}$
- Para cada ponto $\mathbf{p}_k$, faz $\vec{d} = \mathbf{p}_k - \mathbf{c}$; se $\|d\| > r$:
  - $\mathbf{c} = \mathbf{c} + \frac{\|d\| - r}{2}\hat{d}$ e $r = \frac{\|d\| + r}{2}$

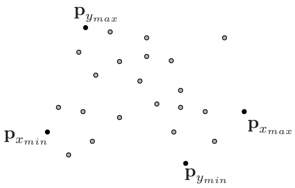



Tempo Esperado:

# Determinação de círculo envolvente

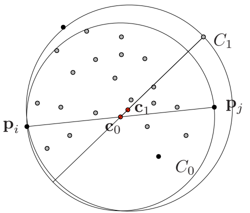
Algoritmo baseado em heurística

- Não resulta no círculo mínimo
- Usado como forma simples de determinar círculo envolvente




Algoritmo

- Ache os dois pares de pontos: $\{\mathbf{p}_{x_{min}}, \mathbf{p}_{x_{max}}\}$, $\{\mathbf{p}_{y_{min}}, \mathbf{p}_{y_{max}}\}$
- Escolha par mais distante: $\mathbf{p}_i, \mathbf{p}_j$
- Considere o círculo diametral: $\mathbf{c} = \frac{\mathbf{p}_i + \mathbf{p}_j}{2}$ e $r = \frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{2}$
- Para cada ponto $\mathbf{p}_k$, faz $\vec{d} = \mathbf{p}_k - \mathbf{c}$; se $\|d\| > r$:
  - $\mathbf{c} = \mathbf{c} + \frac{\|d\| - r}{2}\hat{d}$ e $r = \frac{\|d\| + r}{2}$




Tempo Esperado: $O(n)$

# Determinação de círculo mínimo

Algoritmo incremental randômico

- Permutação randômica de $\mathbf{p}_1, \ldots, \mathbf{p}_n$
- Considere $P_i = \{\mathbf{p}_1, \ldots, \mathbf{p}_i\}$
- Considere $C_i$ como o círculo mínimo de $P_i$

# Determinação de círculo mínimo

Algoritmo incremental randômico

- Permutação randômica de $\mathbf{p}_1, \ldots, \mathbf{p}_n$
- Considere $P_i = \{\mathbf{p}_1, \ldots, \mathbf{p}_i\}$
- Considere $C_i$ como o círculo mínimo de $P_i$

**Lema**: para $2 < i < n$, temos:

- Se $\mathbf{p}_i \in C_{i-1}$, então $C_i = C_{i-1}$
- Se $\mathbf{p}_i \notin C_{i-1}$, então $\mathbf{p}_i$ está em contato com $C_i$

# Determinação de círculo mínimo

**Algoritmo**: `MinCircle` $(\{\mathbf{p}_1, ..., \mathbf{p}_n\})$

1. Faça uma permutação em $\mathbf{p}_1, ..., \mathbf{p}_n$ de $P$
2. Inicialize $C_2$ considerando $\{\mathbf{p}_1, \mathbf{p}_2\}$
3. Para $i = 3, ..., n$ faça:
   - Se $p_i \in C_{i-1}$ então: $C_i = C_{i-1}$
   - Senão: $C_i =$ `MinCircleWithPoint` $(\{\mathbf{p}_1, ..., \mathbf{p}_{i-1}\}, \mathbf{p}_i)$
4. Retorna $C_n$

# Determinação de círculo mínimo

**Algoritmo**: `MinCircleWithPoint` $(\{\mathbf{p}_1, ..., \mathbf{p}_n\}, \mathbf{q})$

1. Inicialize $C_1$ considerando $\{\mathbf{p}_1, \mathbf{q}\}$
2. Para $j = 2, ..., n$ faça:
   - Se $p_j \in C_{j-1}$ então: $C_j = C_{j-1}$
   - Senão: $C_j =$ `MinCircleWith2Points` $(\{\mathbf{p}_1, ..., \mathbf{p}_{j-1}\}, \mathbf{p}_j, \mathbf{q})$
3. Retorna $C_n$

# Determinação de círculo mínimo

**Algoritmo**: `MinCircleWithPoint` $(\{\mathbf{p}_1, ..., \mathbf{p}_n\}, \mathbf{q})$

1. Inicialize $C_1$ considerando $\{\mathbf{p}_1, \mathbf{q}\}$
2. Para $j = 2, ..., n$ faça:
   - Se $p_j \in C_{j-1}$ então: $C_j = C_{j-1}$
   - Senão: $C_j =$ `MinCircleWith2Points` $(\{\mathbf{p}_1, ..., \mathbf{p}_{j-1}\}, \mathbf{p}_j, \mathbf{q})$
3. Retorna $C_n$

**Algoritmo**: `MinCircleWith2Points` $(\{\mathbf{p}_1, ..., \mathbf{p}_n\}, \mathbf{q}_1, \mathbf{q}_2)$

1. Inicialize $C_0$ considerando $\{\mathbf{q}_1, \mathbf{q}_2\}$
2. Para $k = 1, ..., n$ faça:
   - Se $p_k \in C_{k-1}$ então: $C_k = C_{k-1}$
   - Senão: $C_k =$ circuncírculo de $\mathbf{p}_k, \mathbf{q}_1, \mathbf{q}_2$
3. Retorna $C_n$

# Determinação de círculo mínimo

Tempo esperado

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$
- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{i=2}^{n} O(i)p$
  - Onde $p$ é a propabilidade da cláusula *senão* ser executada

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$
- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{i=2}^{n} O(i)p$
  - Onde $p$ é a propabilidade da cláusula *senão* ser executada

Determinação de $p$

- Considere o problema inverso:
  - Ao remover um ponto de $P_i$, qual a probabilidade de remover do contorno?

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$
- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{i=2}^{n} O(i)p$
  - Onde $p$ é a propabilidade da cláusula *senão* ser executada

Determinação de $p$

- Considere o problema inverso:
  - Ao remover um ponto de $P_i$, qual a probabilidade de remover do contorno?
  - Logo: $p = \frac{2}{i}$

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$
- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{i=2}^{n} O(i)p$
  - Onde $p$ é a propabilidade da cláusula *senão* ser executada

Determinação de $p$

- Considere o problema inverso:
  - Ao remover um ponto de $P_i$, qual a probabilidade de remover do contorno?
  - Logo: $p = \frac{2}{i}$

Então:

- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{2}^{n} O(i)\frac{2}{i} = O(n)$

# Determinação de círculo mínimo

Tempo esperado

- `MinCircleWith2Points` $\longrightarrow O(n)$
- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{i=2}^{n} O(i)p$
  - Onde $p$ é a propabilidade da cláusula *senão* ser executada

Determinação de $p$

- Considere o problema inverso:
  - Ao remover um ponto de $P_i$, qual a probabilidade de remover do contorno?
  - Logo: $p = \frac{2}{i}$

Então:

- `MinCircleWithPoint` $\longrightarrow O(n) + \sum_{2}^{n} O(i)\frac{2}{i} = O(n)$

De forma similar, chegamos a:

- `MinCircle` $\longrightarrow O(n)$

# Permutação de um conjunto

Como fazer a permutação dos pontos?

# Permutação de um conjunto

Como fazer a permutação dos pontos?

**Algoritmo**: `RandomPermutation` $(A[1...n])$

1. Para k = n,...,2 faça:
   - $r =$ `random` $(1, k)$
   - $A[k] \leftrightarrow A[r]$

# Permutação de um conjunto

Como fazer a permutação dos pontos?

**Algoritmo**: `RandomPermutation` $(A[1...n])$

1. Para k = n,...,2 faça:
   - $r =$ `random` $(1, k)$
   - $A[k] \leftrightarrow A[r]$

Tempo esperado: $O(n)$