Fecho Convexo

Gabriela Moreira, Luiz Gustavo Eburneo

gabrielamoreira05@gmail.com,
botucacontato@gmail.com

Departamento de Ciência da Computação
Centro de Ciências e Tecnológicas
Universidade do Estado de Santa Catarina

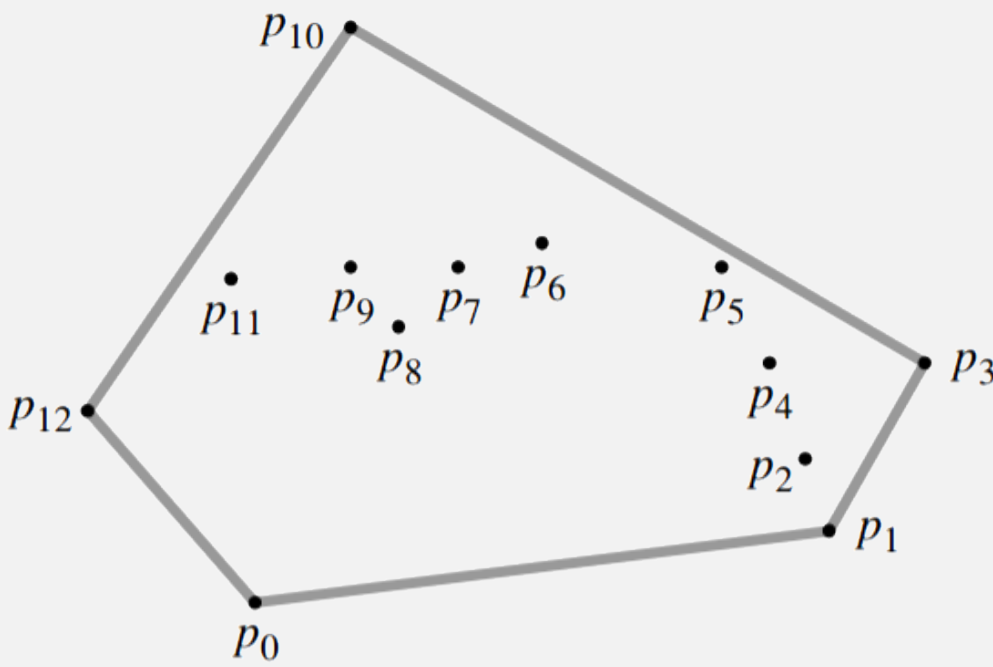
25 de Outubro de 2017

Convex Hull

Encontrar o **menor** polígono convexo que contém todos os pontos de um determinado conjunto.

Polígono Convexo

Os valores dos ângulos internos são menores que 180°.



Aplicações

- Encontrar caminhos
- GIS (Geographic Information System)
- Casamento de padrão visual
- Jogos

Convex Hull: Graham-Scan

GRAHAM-SCAN(Q)

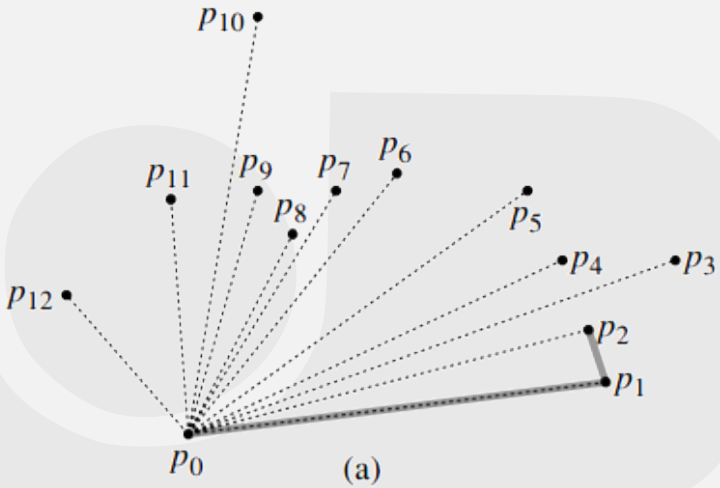
1 let p_0 be the point in Q with the minimum y-coordinate,
or the leftmost such point in case of a tie
2 let $\langle p_1, p_2, \ldots, p_m \rangle$ be the remaining points in Q,
sorted by polar angle in counterclockwise order around p_0
(if more than one point has the same angle, remove all but
the one that is farthest from p_0)
3 PUSH(p_0, S)
4 PUSH(p_1, S)
5 PUSH(p_2, S)
6 **for** $i \leftarrow 3$ **to** m
7 **do while** the angle formed by points NEXT-TO-TOP(S), TOP(S),
and p_i makes a nonleft turn
8 **do** POP(S)
9 PUSH(p_i, S)
10 **return** S

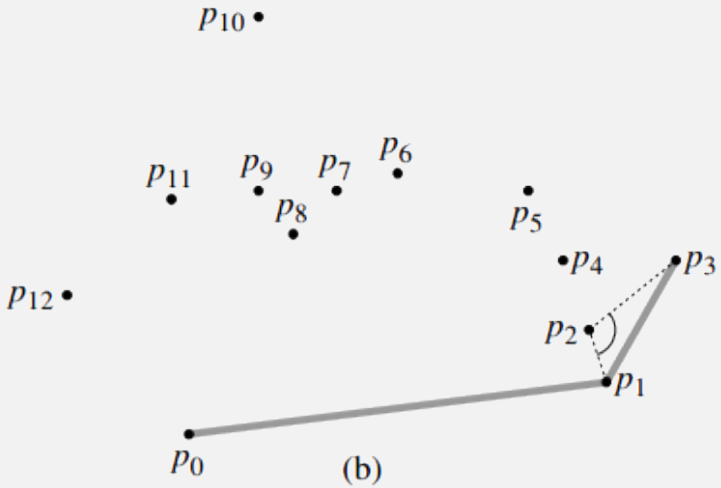
Cross Product

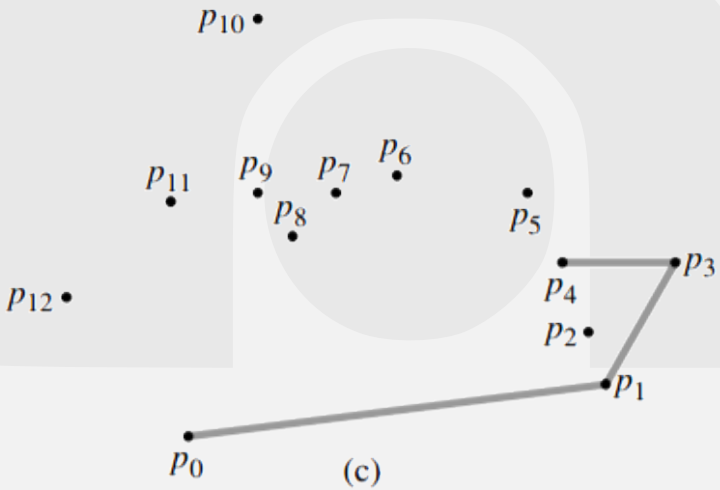
$$p_1 \times p_2 = \det \begin{pmatrix} x_1 & x_2 \\ y_1 & y_2 \end{pmatrix} = x_1 y_2 - x_2 y_1$$

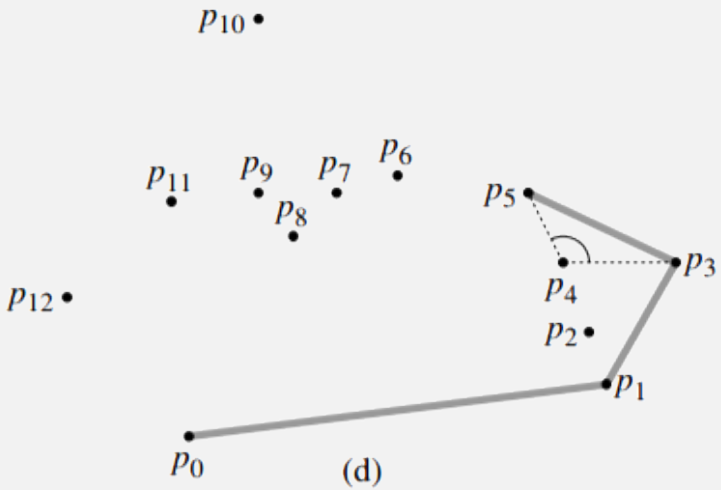
- Se > 0: *clockwise*
- Se < 0: *counterclockwise*
- Se $= 0$: colineares

Convex Hull: Graham-Scan

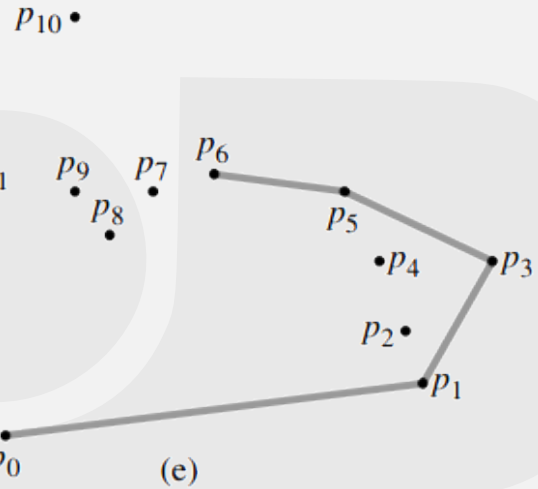


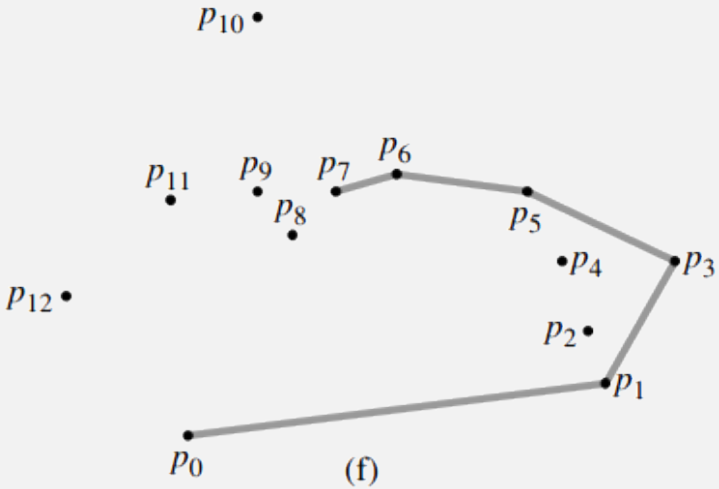


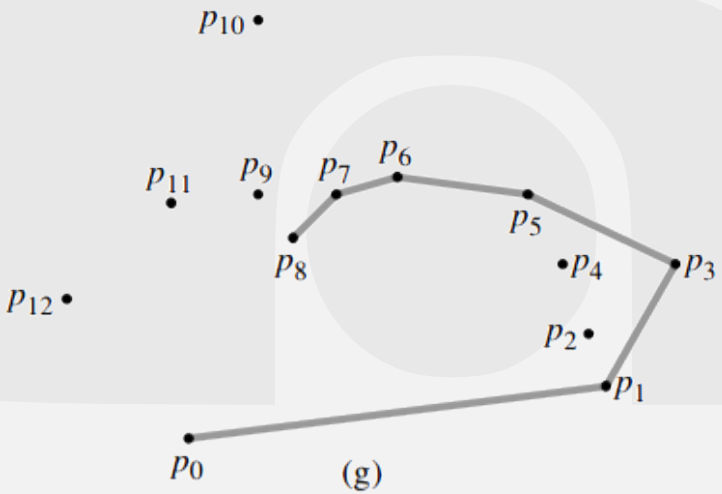


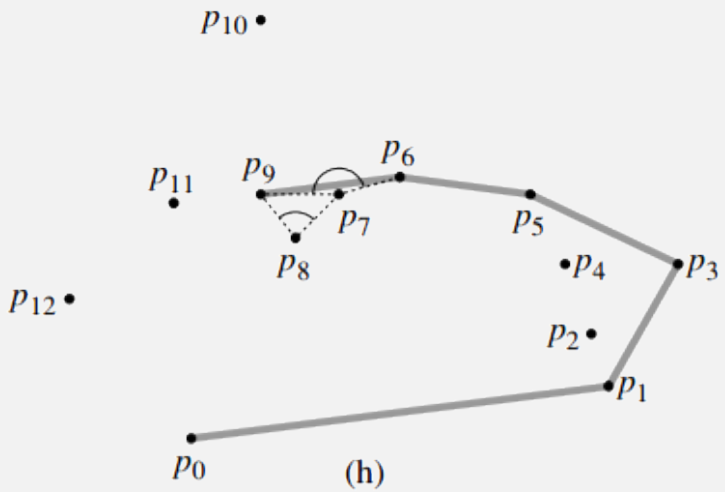


Convex Hull: Graham-Scan

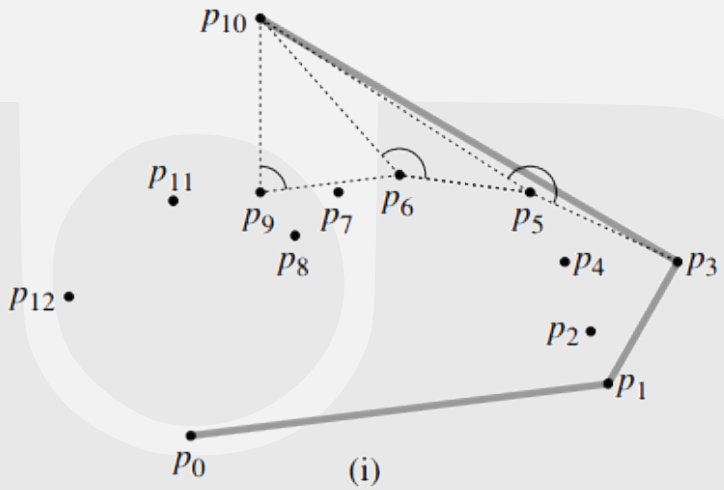




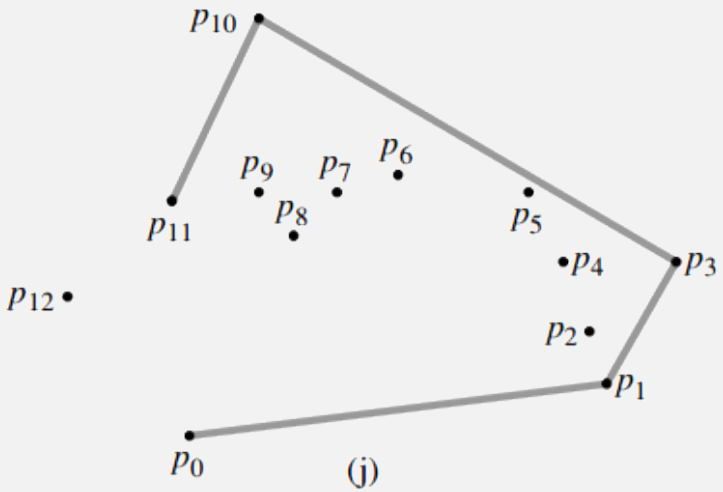




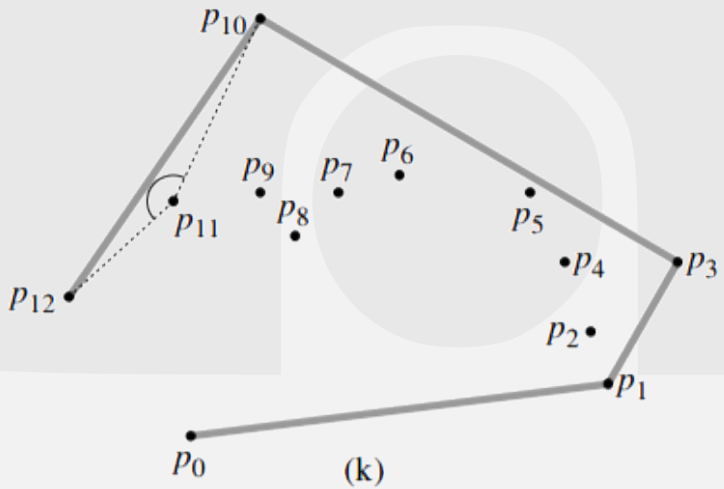
Convex Hull - Graham-Scan



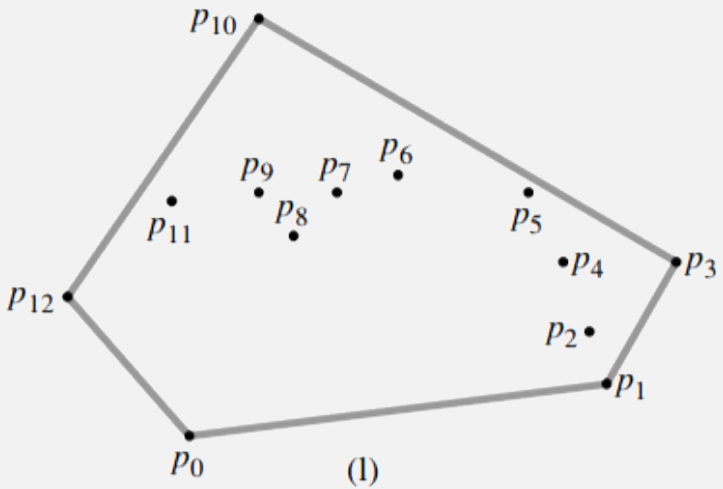
(i)



(j)

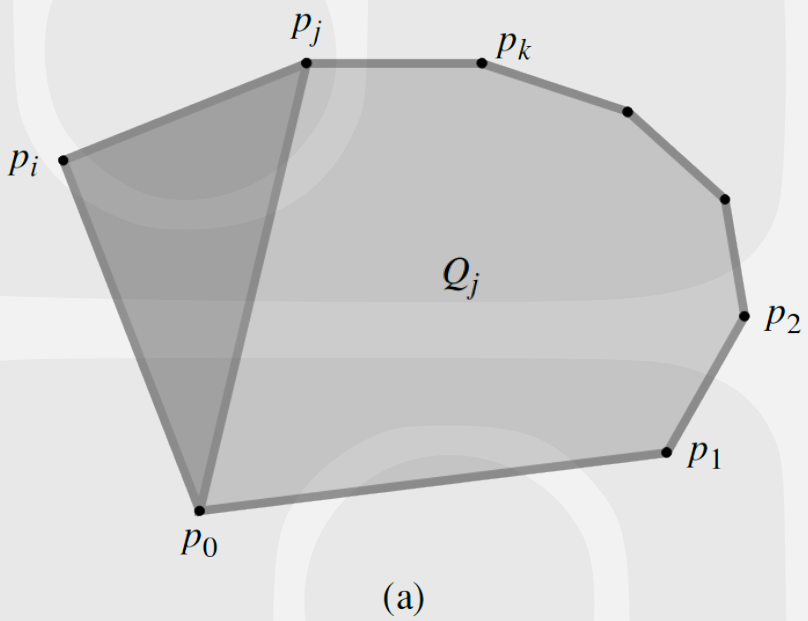


(k)

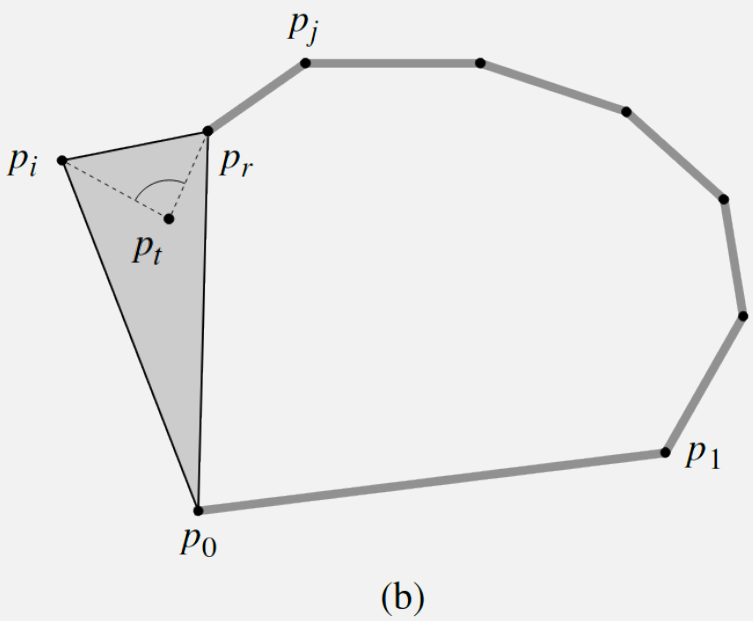


(l)

Convex Hull: Graham-Scan - Corretude



(a)



(b)

Convex Hull: Graham-Scan

GRAHAM-SCAN(Q)

1 let p_0 be the point in Q with the minimum y-coordinate,
or the leftmost such point in case of a tie
2 let $\langle p_1, p_2, \ldots, p_m \rangle$ be the remaining points in Q,
sorted by polar angle in counterclockwise order around p_0
(if more than one point has the same angle, remove all but
the one that is farthest from p_0)
3 PUSH(p_0, S)
4 PUSH(p_1, S)
5 PUSH(p_2, S)
6 **for** $i \leftarrow 3$ **to** m
7 **do while** the angle formed by points NEXT-TO-TOP(S), TOP(S),
and p_i makes a nonleft turn
8 **do** POP(S)
9 PUSH(p_i, S)
10 **return** S

$\mathcal{O}(n \log n)$

Método incremental.

$\mathcal{O}(n \log n)$

Método incremental.

$\mathcal{O}(n \log n)$

Método dividir e conquistar.

$\mathcal{O}(n \log n)$

Método incremental.

$\mathcal{O}(n \log n)$

Método dividir e conquistar.

$\mathcal{O}(n \log h)$

Método podar e pesquisar.

$\mathcal{O}(n \log n)$

Método incremental.

$\mathcal{O}(n \log n)$

Método dividir e conquistar.

$\mathcal{O}(n \log h)$

Método podar e pesquisar.

$\mathcal{O}(n^2)$

QuickHull

Fim =)