UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

Computación Gráfica

Class 13. Computational Geometry. Convex Hull. Sweep Line Algorithms.

Professor: Eric Biagioli

TRANSFORMATEC

Today

- Convex Hull: Jarvis March, Graham Scam, Removal of interior points, QuickHull
- Sweep line
  - Yet another approach for Convex Hull: Andrew's monotone chain
  - Existence of intersections in a set of N segments
  - Closest pair

<u>**References for the class of today:**</u> (part of the first partial exam)

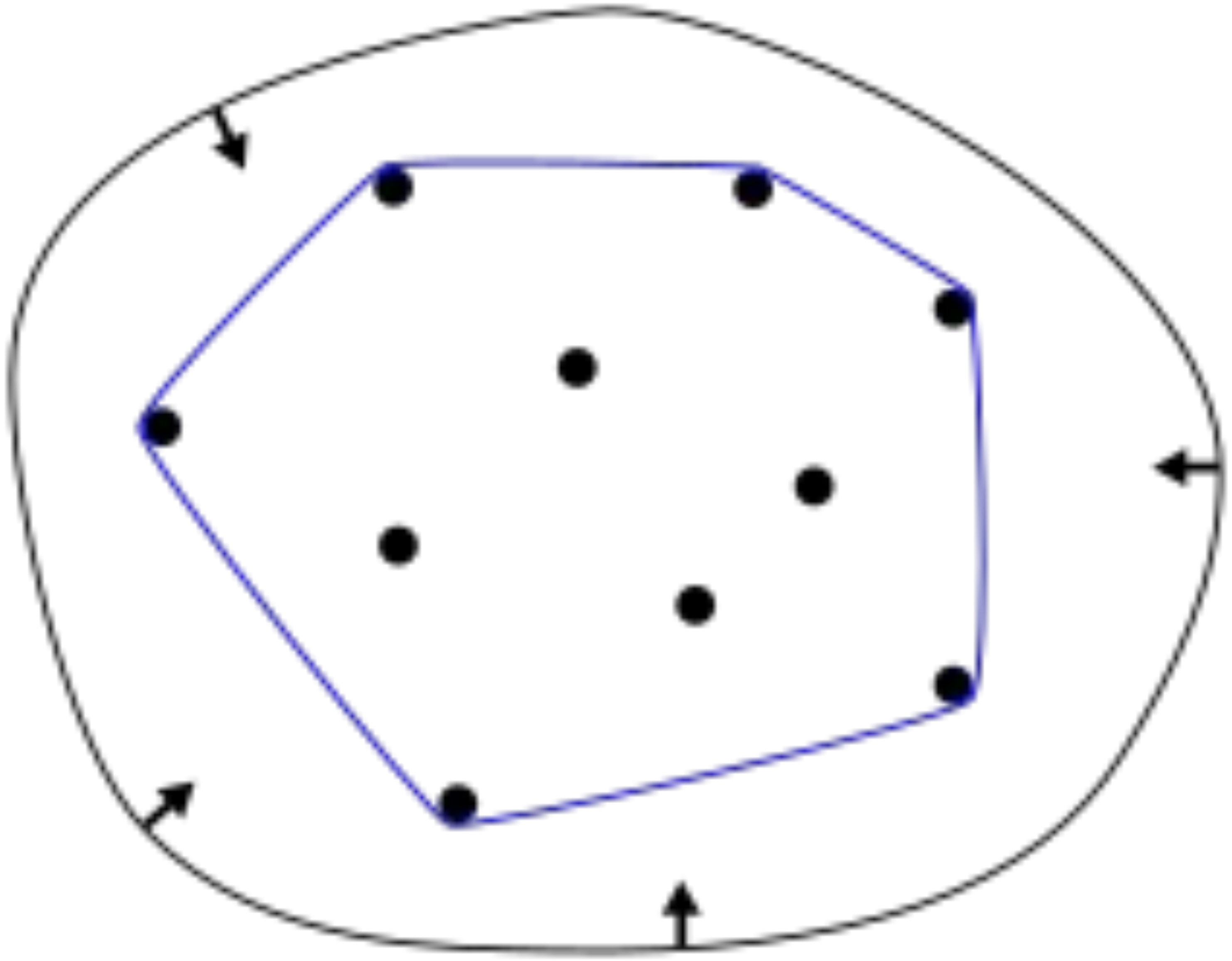
- O’Rourke, J. Computational Geometry in C, second ed. Cambridge University Press, Oct. 1998. → **Section 3.4 (QuickHull)**

- Berg, M. d., Cheong, O., Kreveld, M. v., and Overmars, M. Computational Geometry: Algorithms and Applications, 3rd ed. ed. Springer-Verlag TELOS, Santa Clara, CA, USA, 2008. → **Chapter 1 (Convex Hulls), Section 2.1 (Detect and/or compute line segment intersections in a set of N segments)**

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. Introduction to Algorithms, Third Edition, 3rd ed. The MIT Press, 2009.→ **Section 33.2 (Determining wether any pair of segment intersects), 33.3 (Finding the Convex Hull) (Graham Scan & Jarvis March), 33.4 (Closest pair of points)**

Convex hull: definition

Smallest convex polygon that contains all the given points.



Convex hull: 3 approaches and one optimization

- Jarvis March
  - Take one point known to belong to the convex hull, and compute (in time linear in the number of points) the next point in the convex hull.
- Graham Scan
  - Take the leftmost point P, and traverse the N points counter-clockwise, centered at P.
- Optimization: interior points removal
- QuickHull

Convex hull: Jarvis march

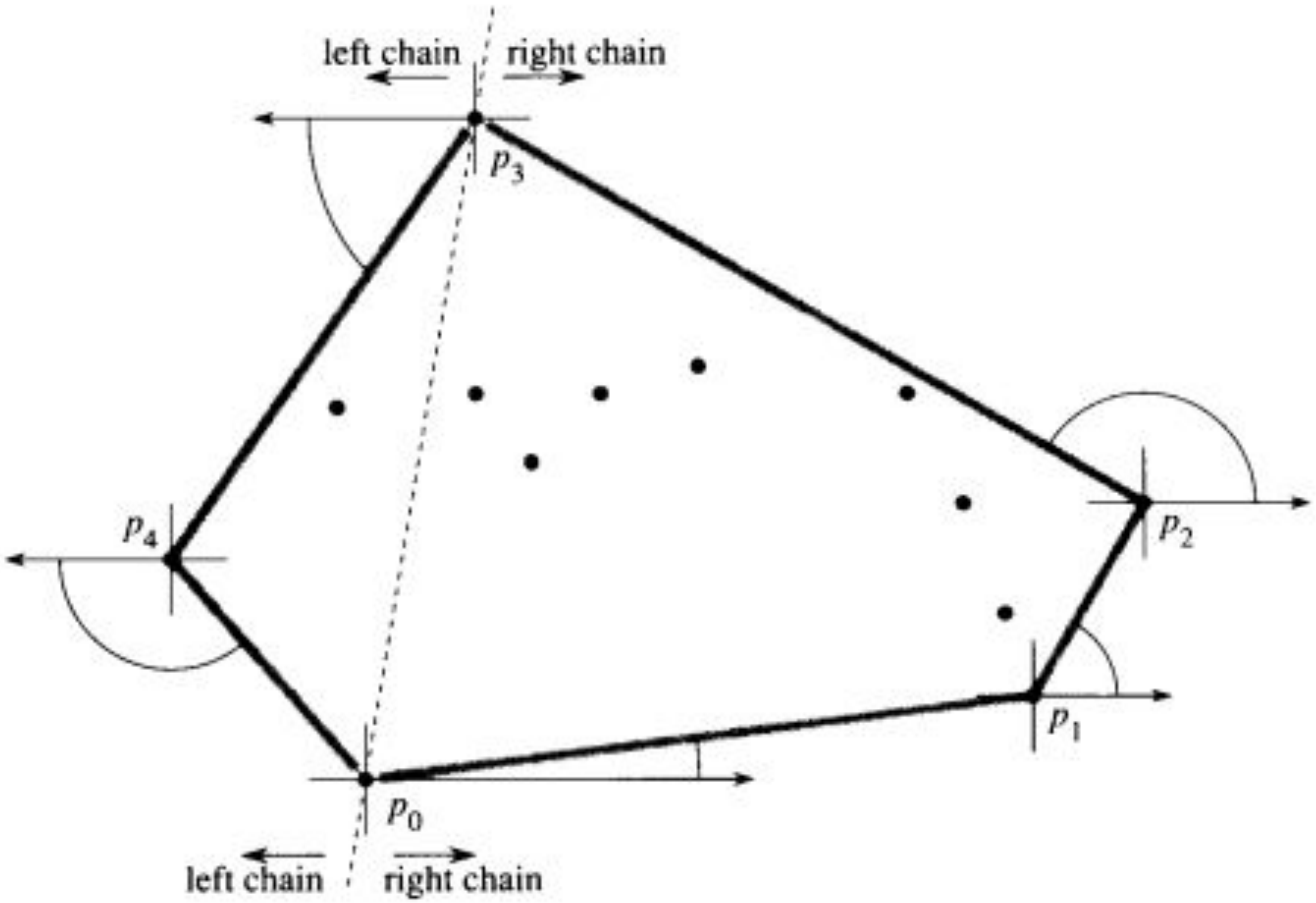
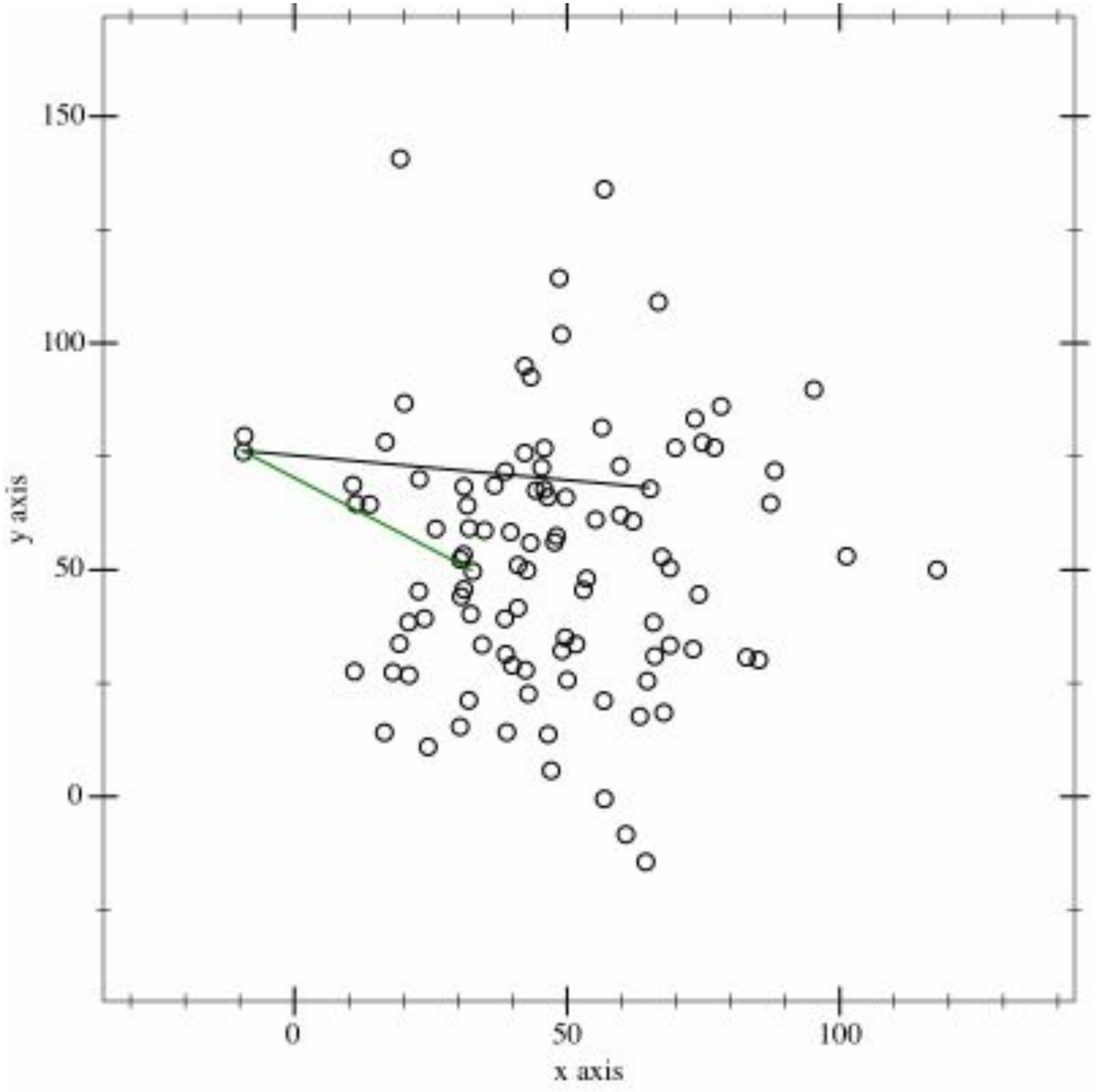
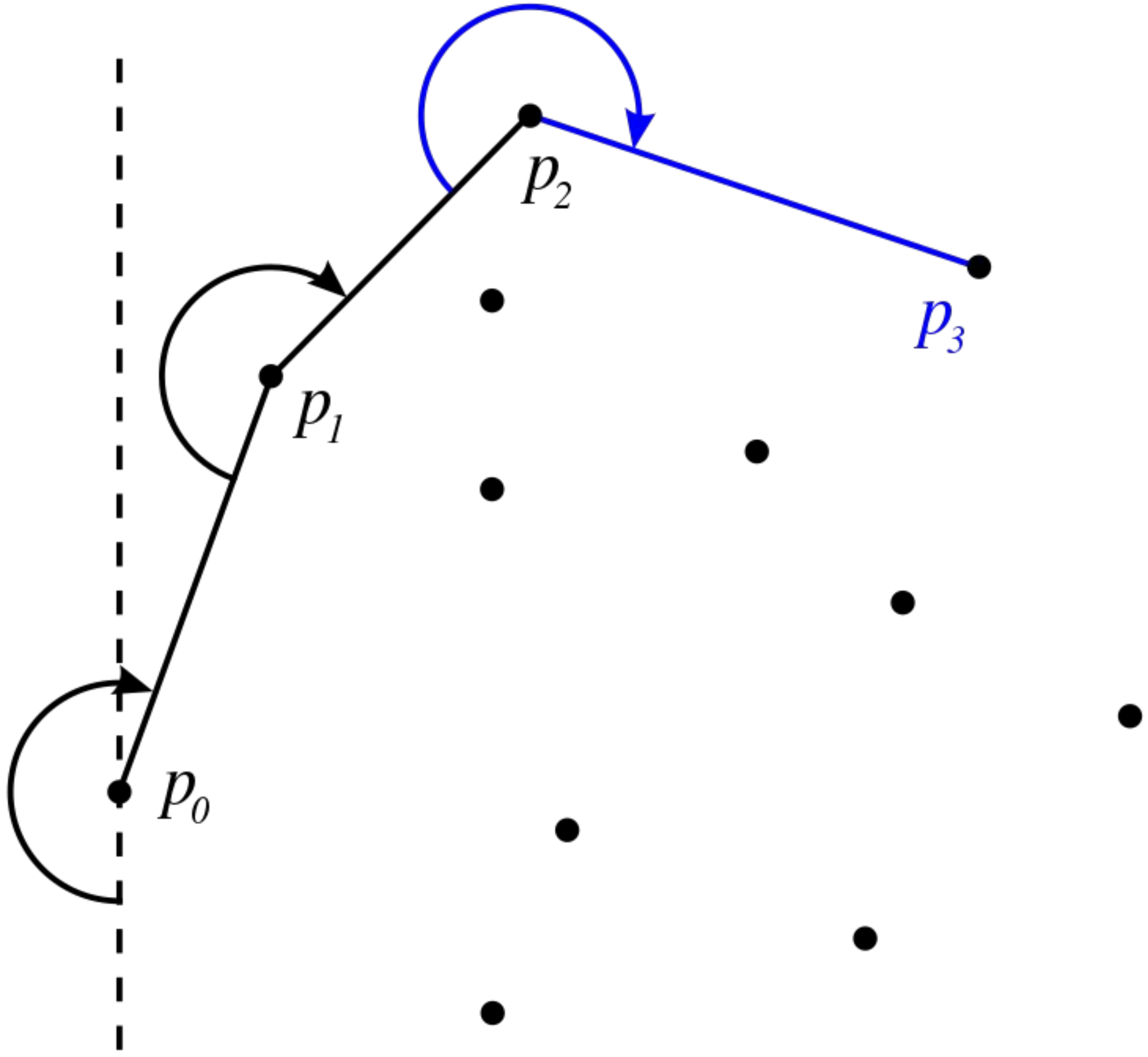


Figure 35.10 The operation of Jarvis's march. The first vertex chosen is the lowest point p_0. The next vertex, p_1, has the least polar angle of any point with respect to p_0. Then, p_2 has the least polar angle with respect to p_1. The right chain goes as high as the highest point p_3. Then, the left chain is constructed by finding least polar angles with respect to the negative x-axis.

Convex hull: Jarvis march





Convex hull: Jarvis march

What is the complexity of the Jarvis march?

Convex hull: Jarvis march

What is the complexity of the Jarvis march?

→ nh, where n is the number of points in the initial set and h is the number of points in the convex hull

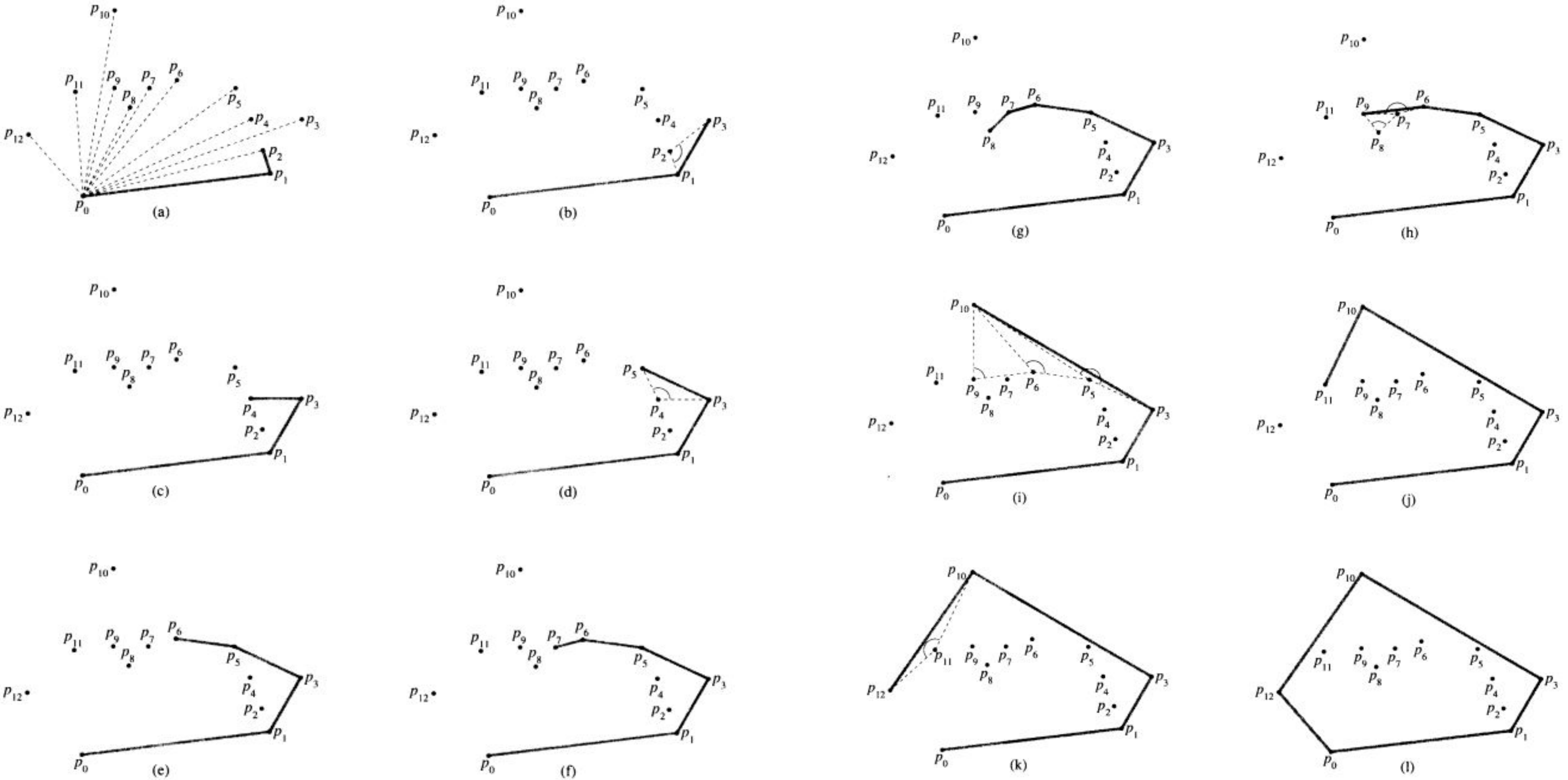
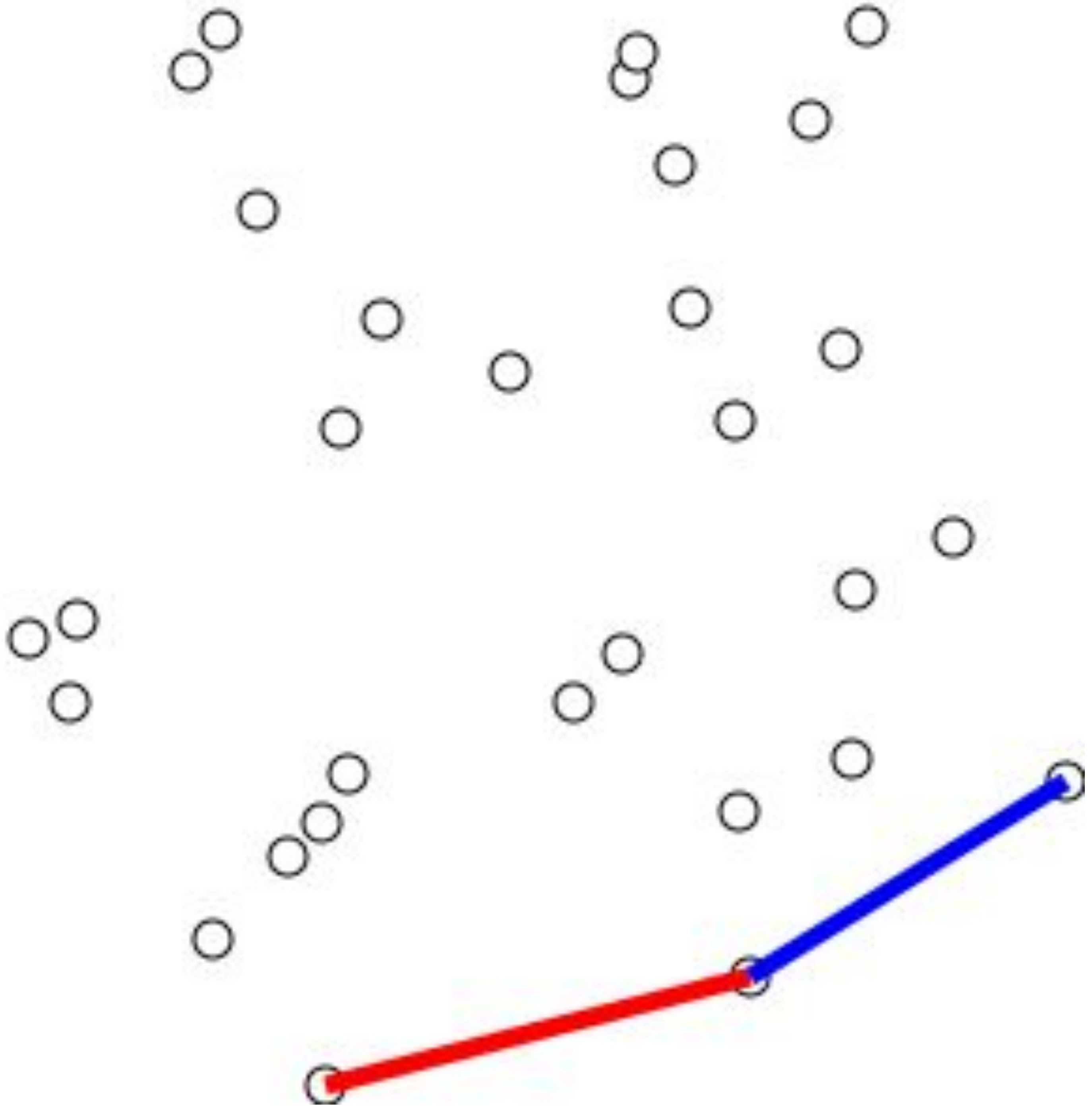
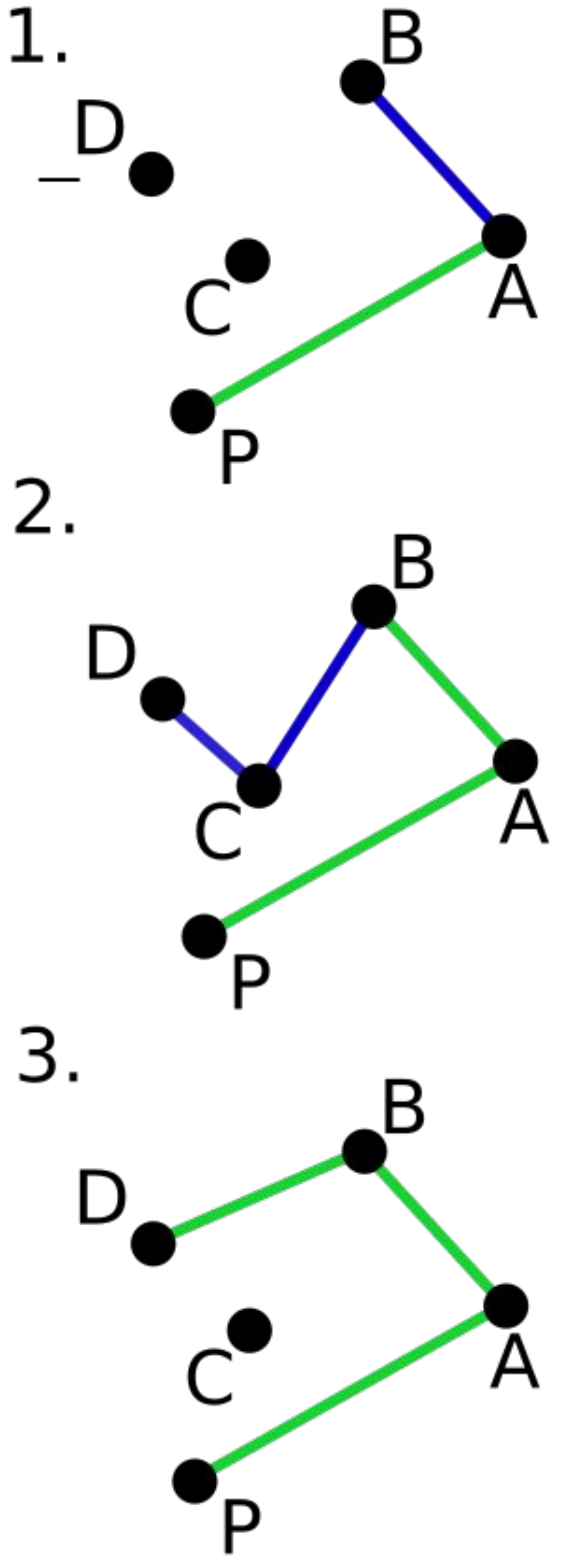


Figure 25.7 The execution of GRAHAM-SCAN on the set Q of Figure 25.6. The

Convex hull: Graham Scan





Convex hull: Graham Scan

```
GRAHAM-SCAN(Q)
let p0 be the point in Q with the minimum y-coordinate,
        or the leftmost such point in case of a tie
let ⟨p1, p2, ..., pm⟩ be the remaining points in Q,
        sorted by polar angle in counterclockwise order around p0
        (if more than point has the same angle, remove all but
        the one that is farthest from p0)
top[S] ← 0
PUSH(p0, S)
PUSH(p1, S)
PUSH(p2, S)
for i ← 3 to m
    do while the angle formed by points NEXT-TO-TOP(S),
                TOP(S), and pi makes a nonleft turn
           do POP(S)
       PUSH(S, pi)
return S
```

Convex hull: Graham Scan

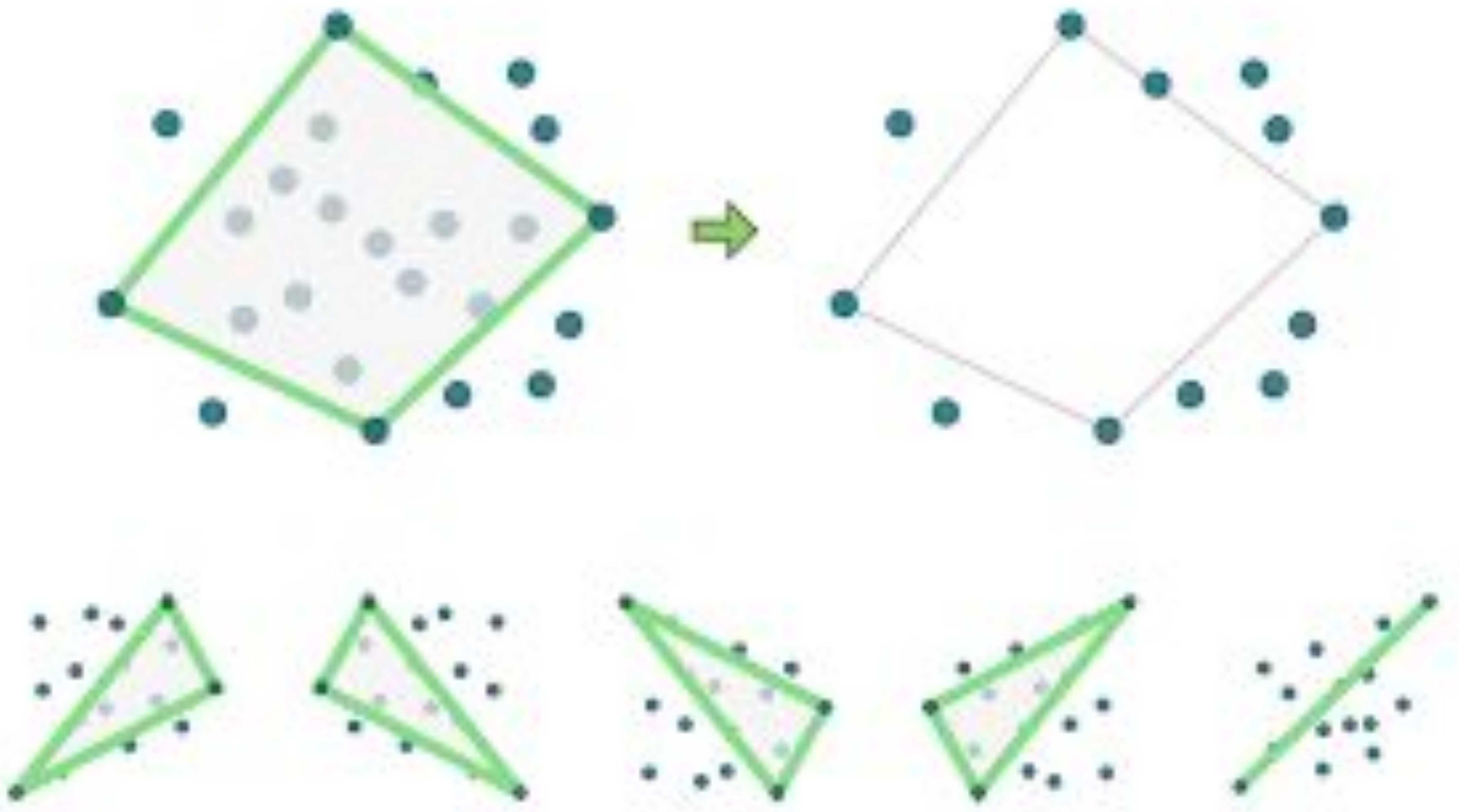
What is the complexity of the Graham scan?

Convex hull: Graham Scan

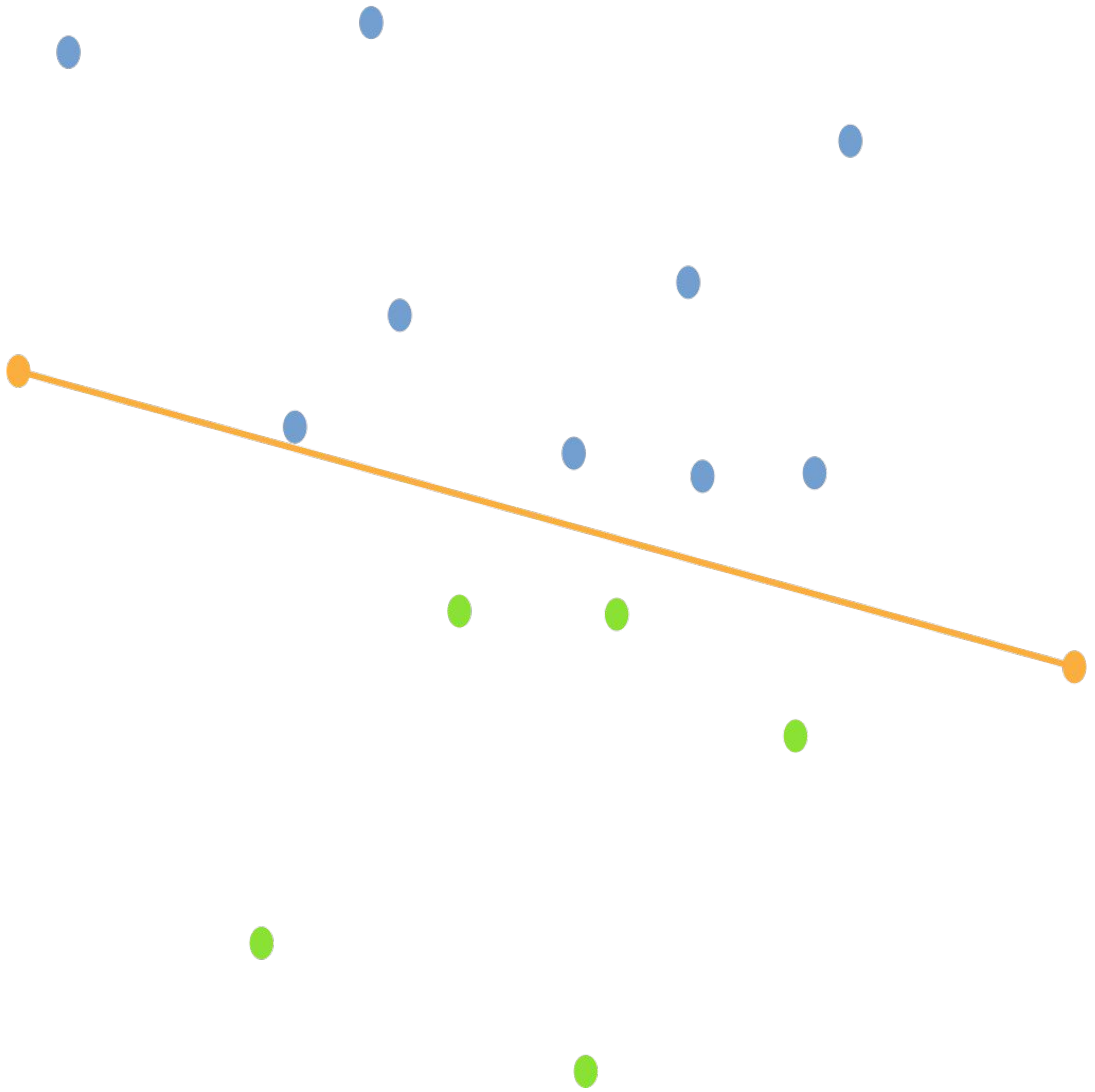
What is the complexity of the Graham scan?

→ dominated by the ordering of the points, so n log n

Interior points removal: Akl–Toussaint heuristic

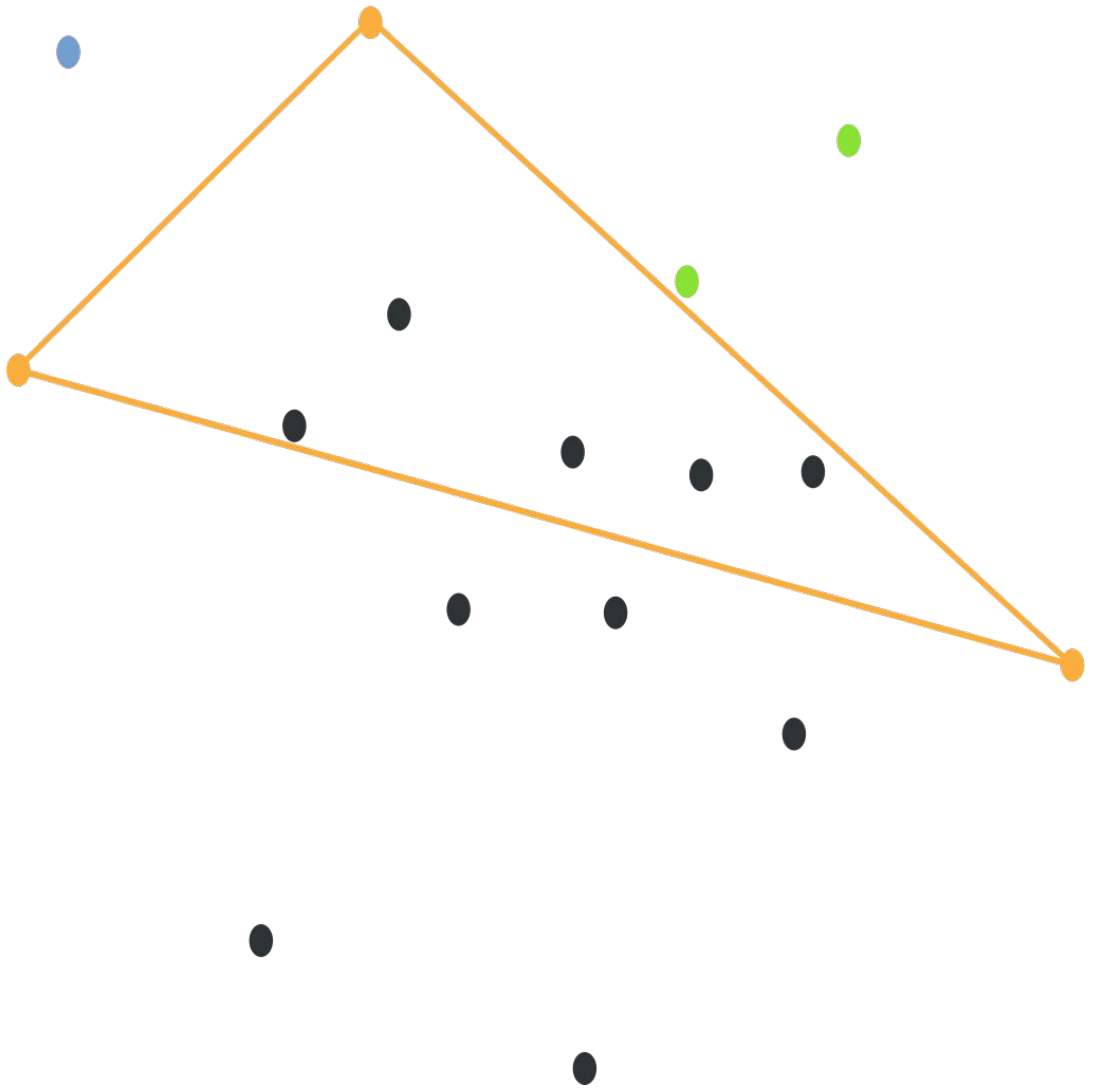


Convex hull: QuickHull



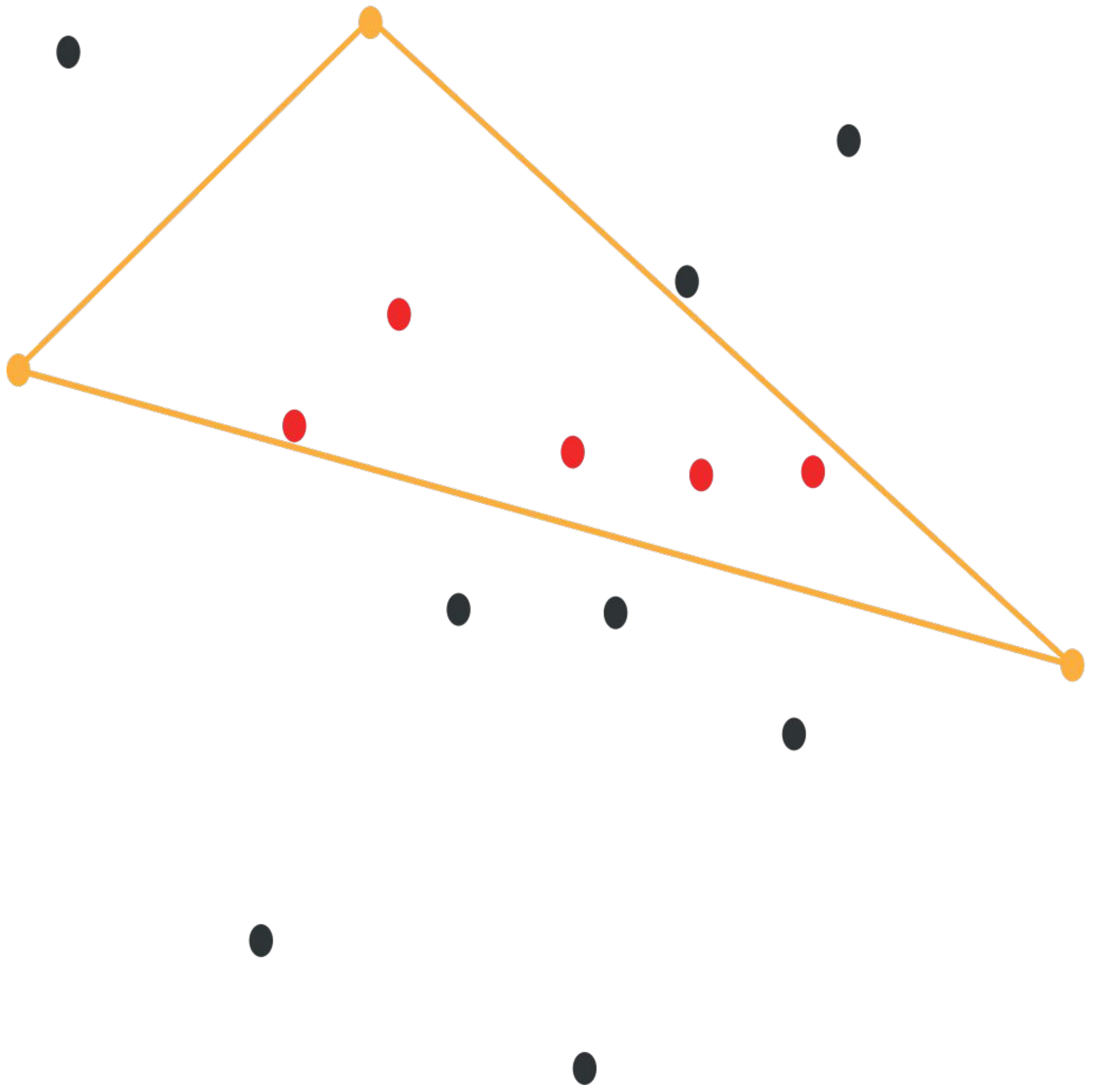
Step 1: Divide the points into two subsets using a line.

Convex hull: QuickHull



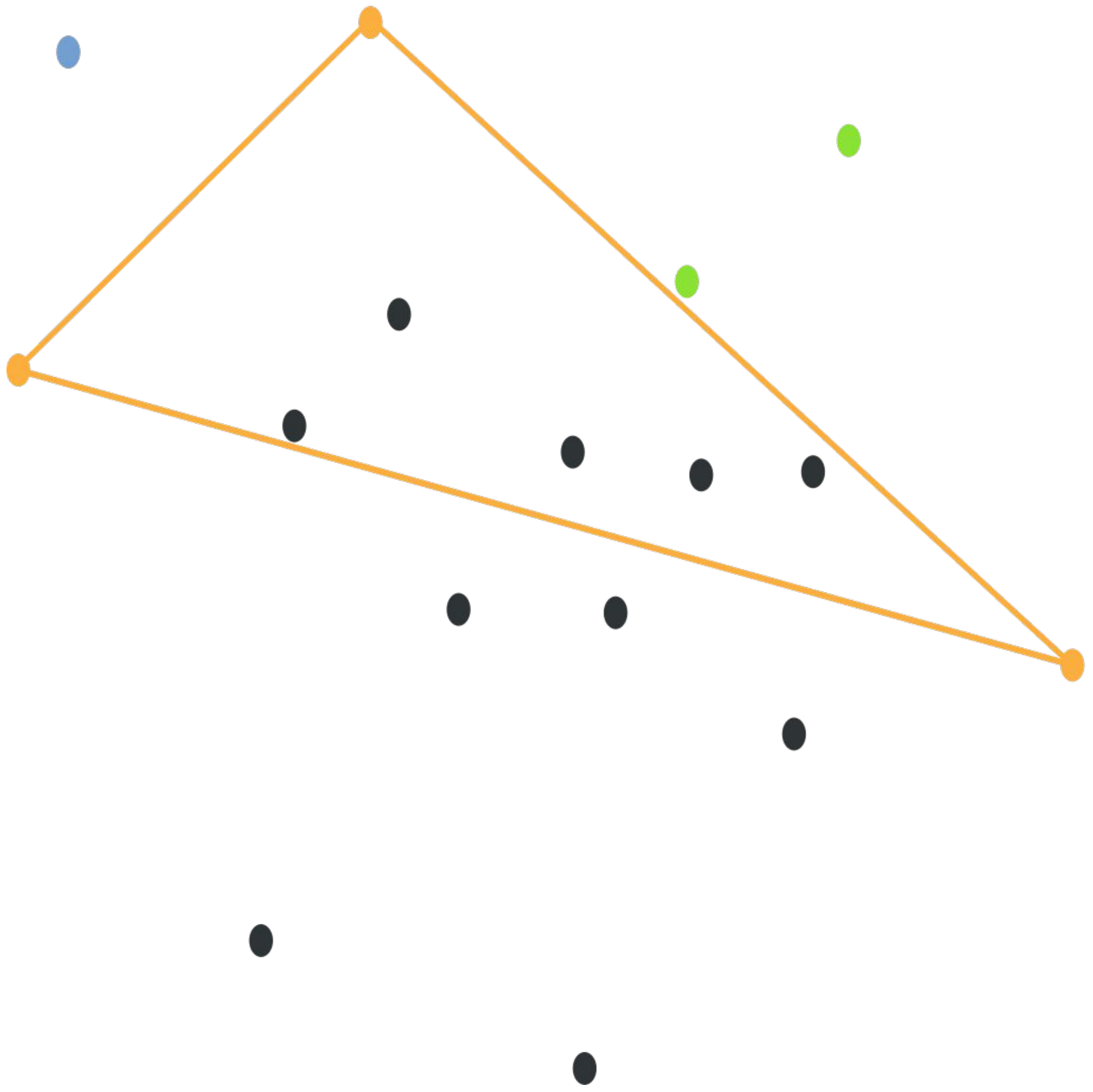
Step 2: Find the point farthest away and form a triangle.

Convex hull: QuickHull



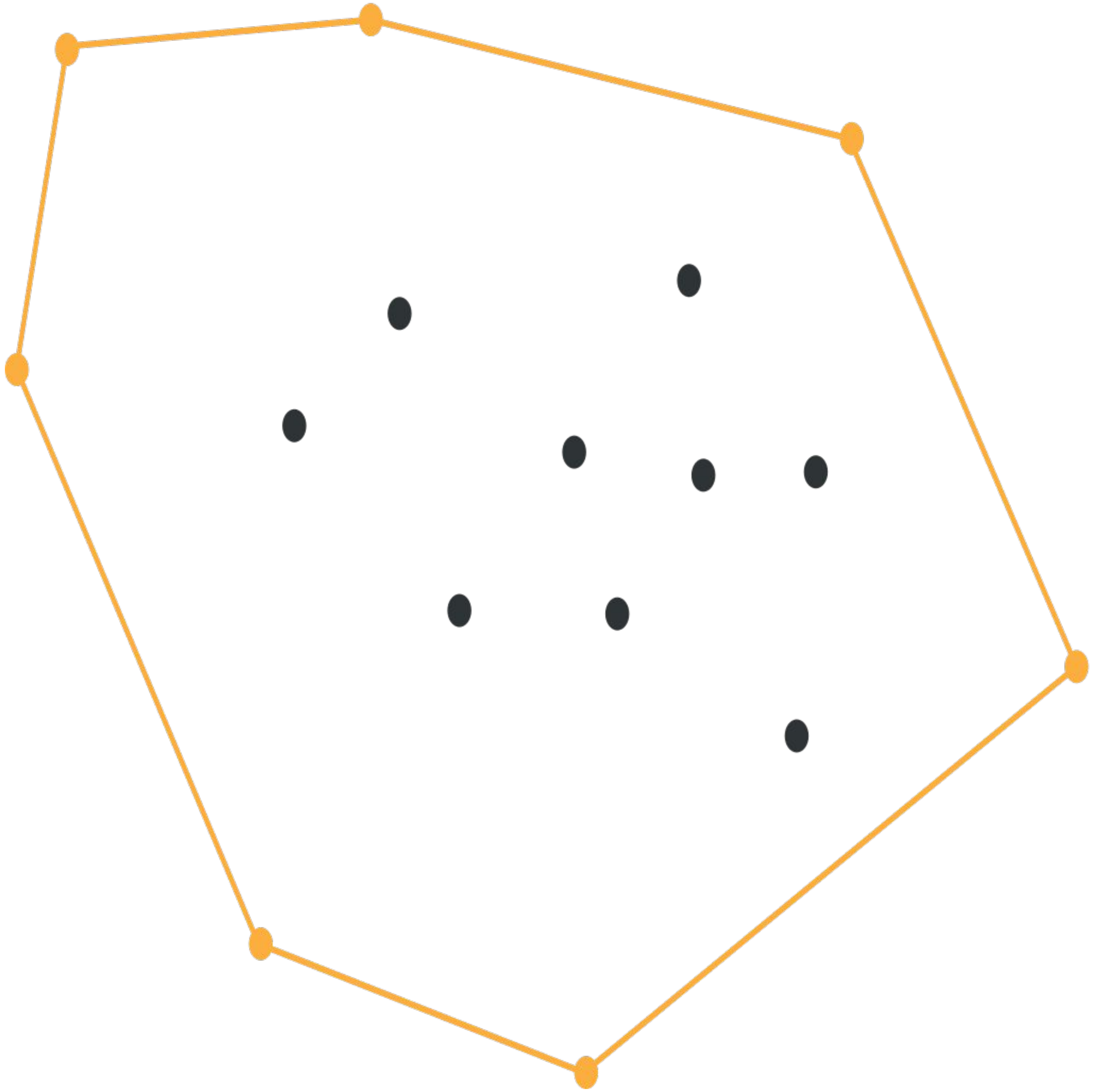
Step 3: Discard the points inside the triangle.

Convex hull: QuickHull



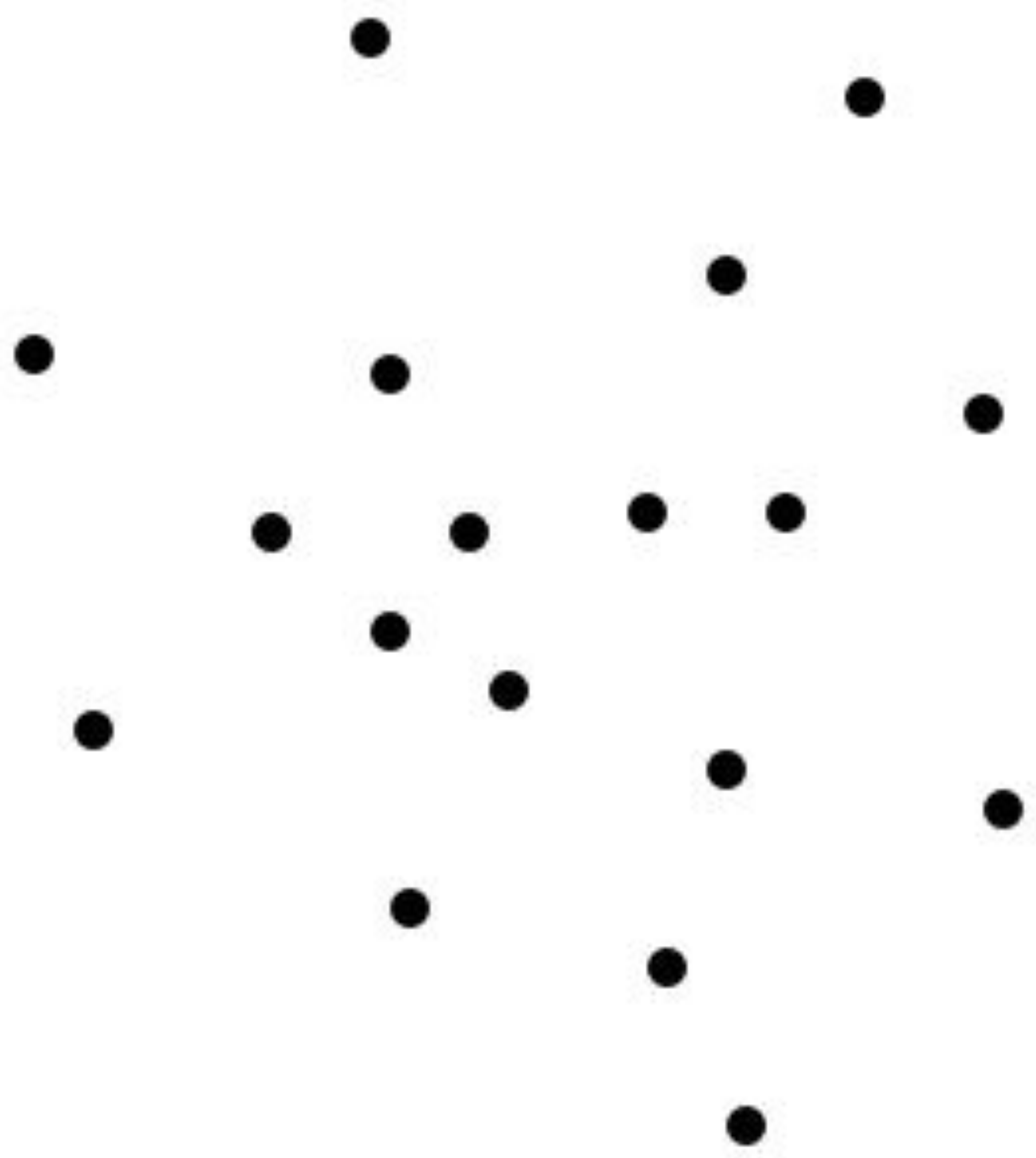
Step 4: Repeat the classification using the two new sides of the triangle.

Convex hull: QuickHull



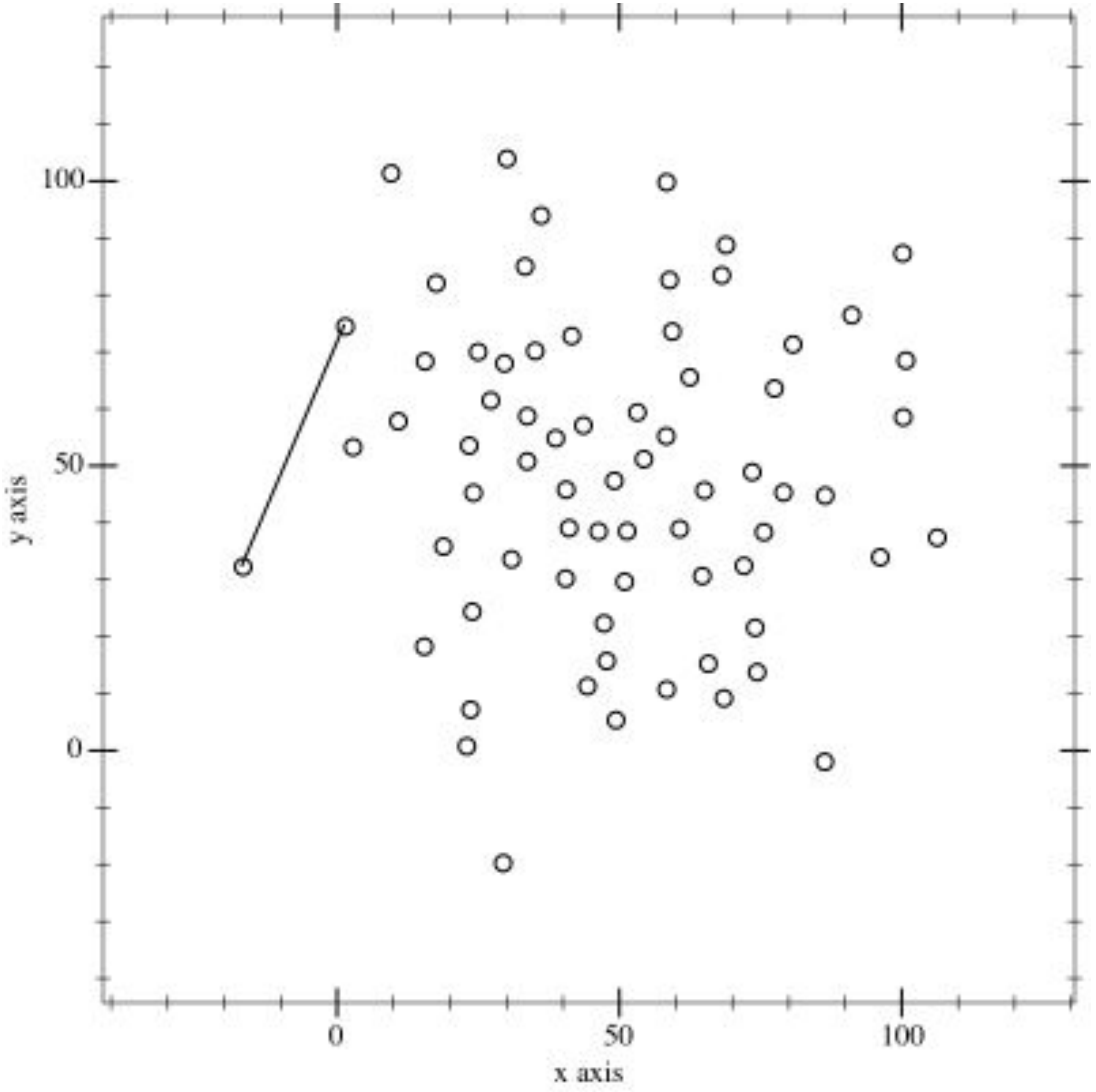
Step 5: Final result.

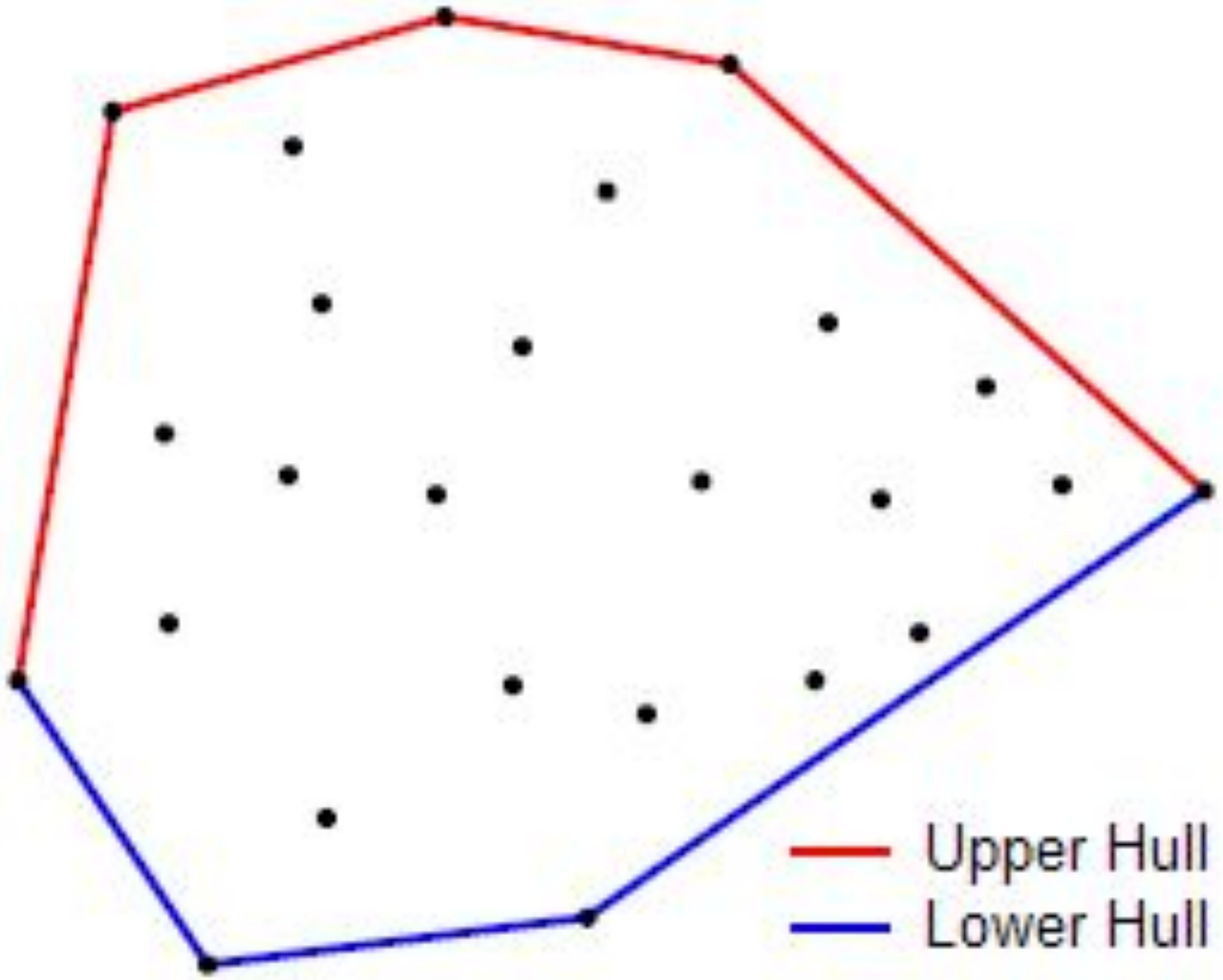
Convex hull: QuickHull



$CH = \{\}$

Sweep Line: Andrew's monotone Chain





Discussion: Detect intersections in a set of N segments

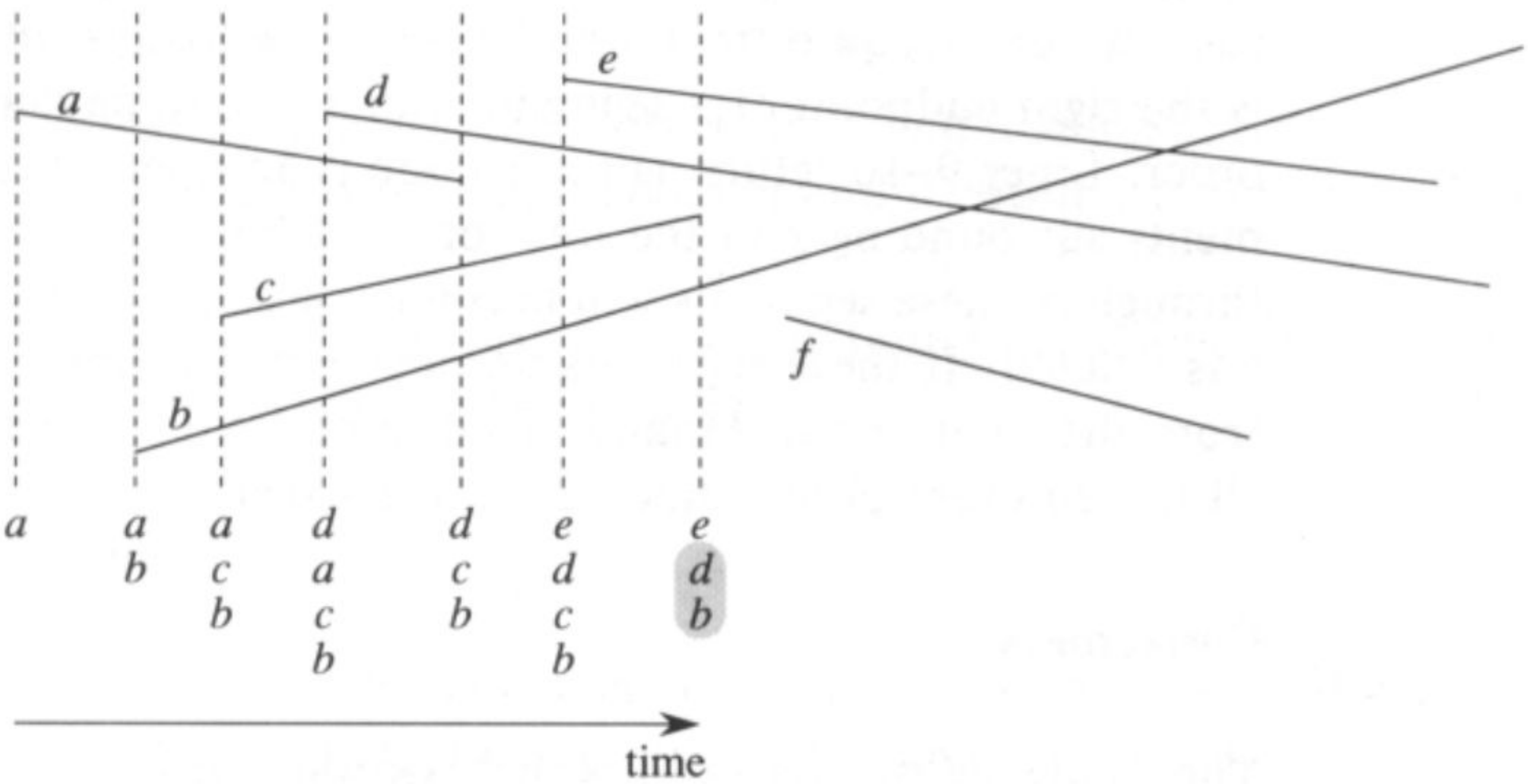


Figure 35.5 The execution of ANY-SEGMENTS-INTERSECT. Each dashed line is the sweep line at an event point, and the ordering of segment names below each sweep line is the total order T at the end of the **for** loop in which the corresponding event point is processed. The intersection of segments d and b is found when segment c is deleted.

Discussion: Closest pair of points in a set of N points

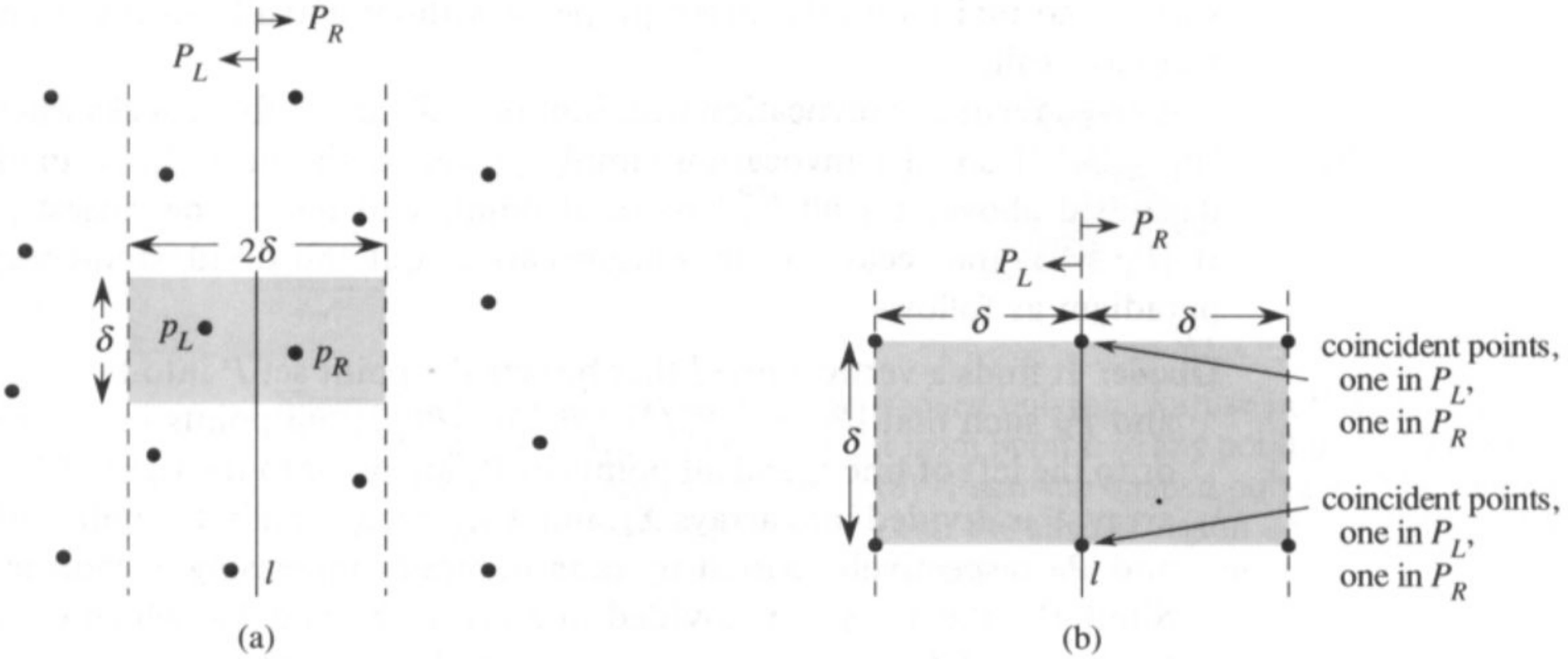


Figure 35.12 Key concepts in the proof that the closest-pair algorithm needs to check only 7 points following each point in the array Y'. **(a)** If $p_L \in P_L$ and $p_R \in P_R$ are less than δ units apart, they must reside within a $\delta \times 2\delta$ rectangle centered at line l. **(b)** How 4 points that are pairwise at least δ units apart can all reside within a $\delta \times \delta$ square. On the left are 4 points in P_L, and on the right are 4 points in P_R. There can be 8 points in the $\delta \times 2\delta$ rectangle if the points shown on line l are actually pairs of coincident points with one point in P_L and one in P_R.

Exercises (100 points is the perfect score. You can exceed the perfect score by doing the additional exercises)

- (50 points) Let A = (x_a, y_a), B = (x_b, y_b), C = (x_c, y_c) be three points given by their coordinates. Propose an algorithm to decide if the angle ABC is smaller than 90 degrees, 90 degrees or larger than 90 degrees.
- (50 points) In which situations the Jarvis March is faster then the Graham Scan?
- (50 points) Design an algorithm that, given N points, returns a path that passes through each point exactly once, and it is guaranteed to not cross.
- (50 points) Design an algorithm that, given N rectangles with sides parallel to the axis x and y, computes the area of the union of them.