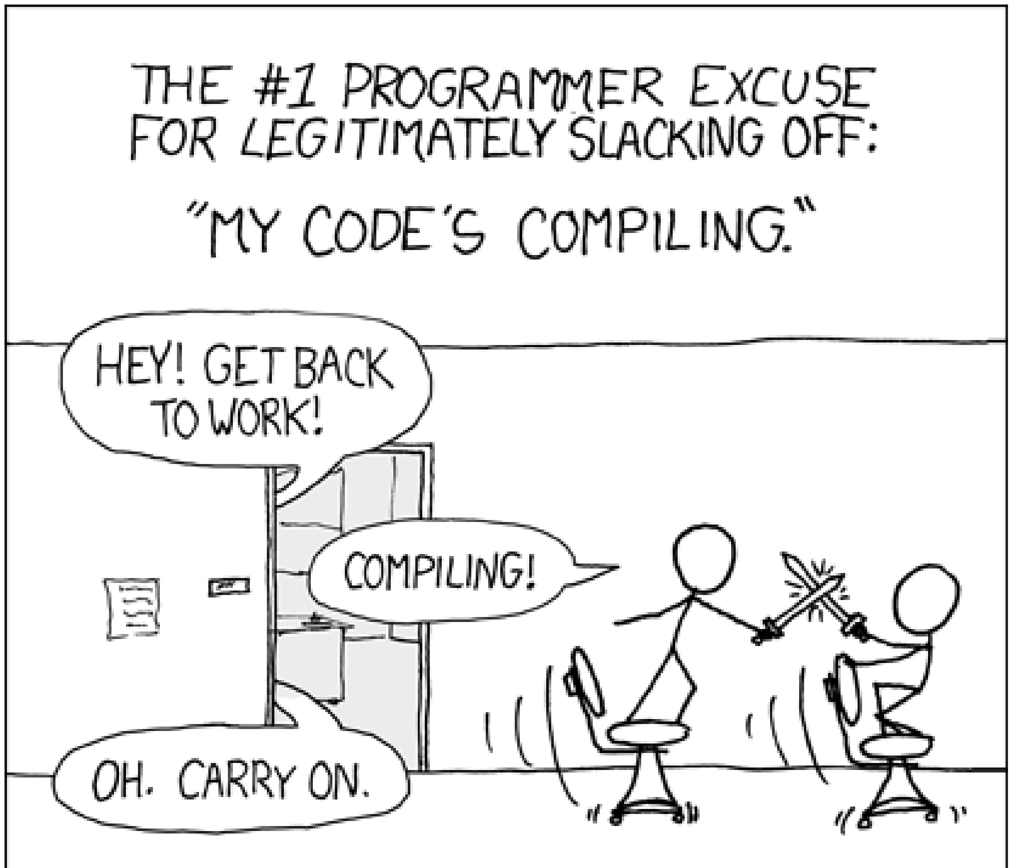
Before We Start

- Can something explain what is it trying to say?

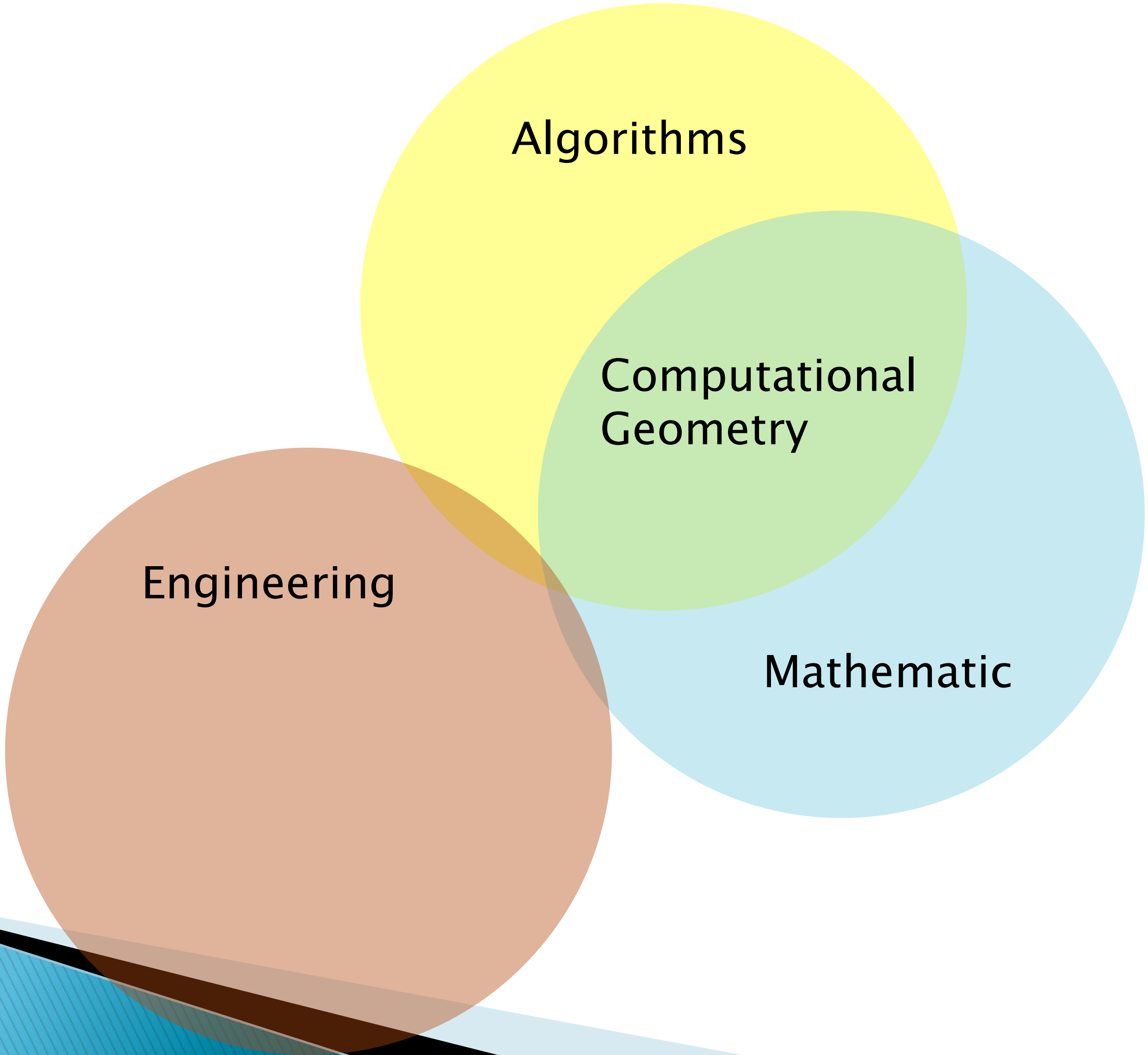


Source: XKCD: https://xkcd.com/303/

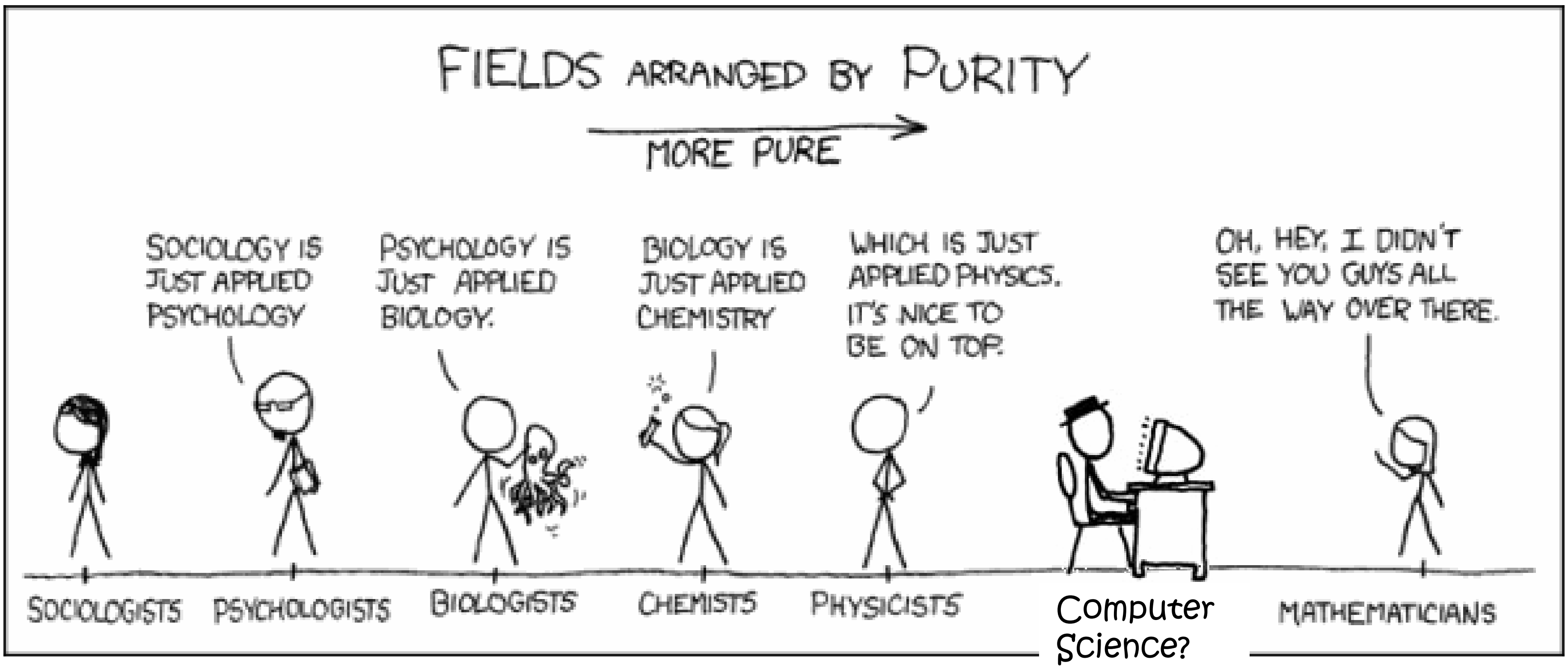
Computational Geometry

A Hybrid of Mathematics and Computer Science

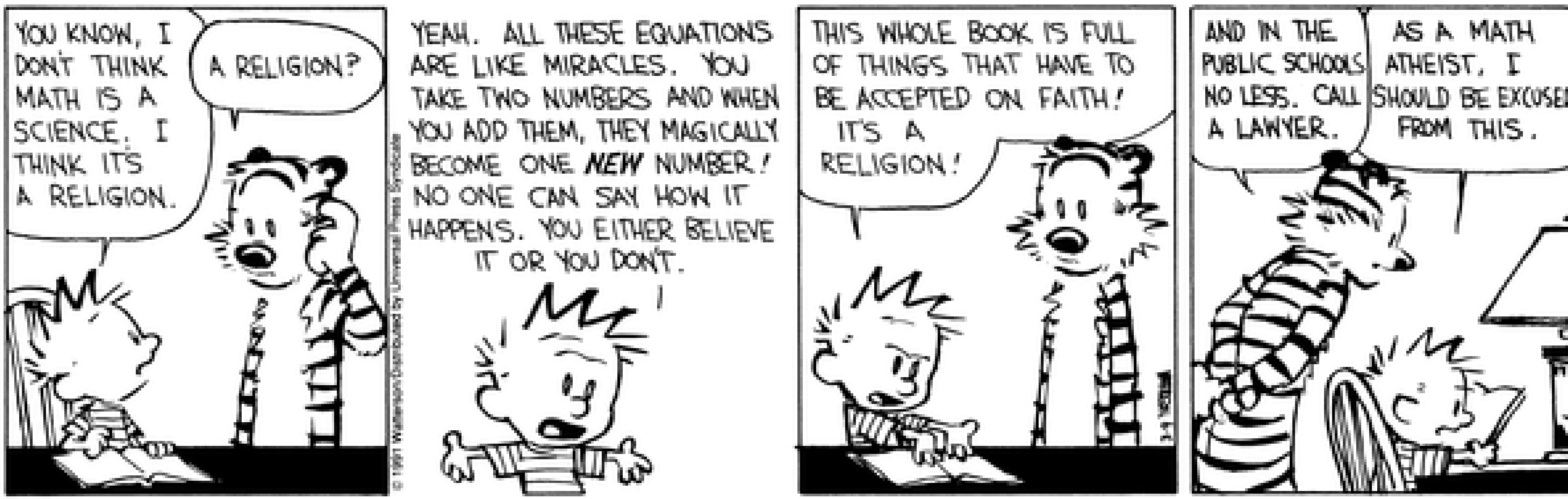
Computational Geometry



Different Fields



Mathematic as a Religion



- Some universities put their math departments under the Art faculties a place of mind

THE UNIVERSITY OF BRITISH COLUMBIA

STUD

Visit the SSC | NEW TO UBC | ACADEMIC SUCCESS | HEALTH & WELLNESS | CAMPUS LIFE | CAREER & EXPE

Vancouver Academic Calendar 2017/18

Search Acade

IN THIS SECTION

- » Introduction
- » Bachelor of Arts
- » Bachelor of Fine Arts
- » Bachelor of International Economics
- » Bachelor of Media Studies
- » Co-operative Education

Homepage > Faculties, Colleges, and Schools > The Faculty of Arts > Bachelor of Arts

Bachelor of Arts

Contents

Computational Geometry

- Solving computational problems by geometric methods
  - E.g. Simplex algorithm in linear programming

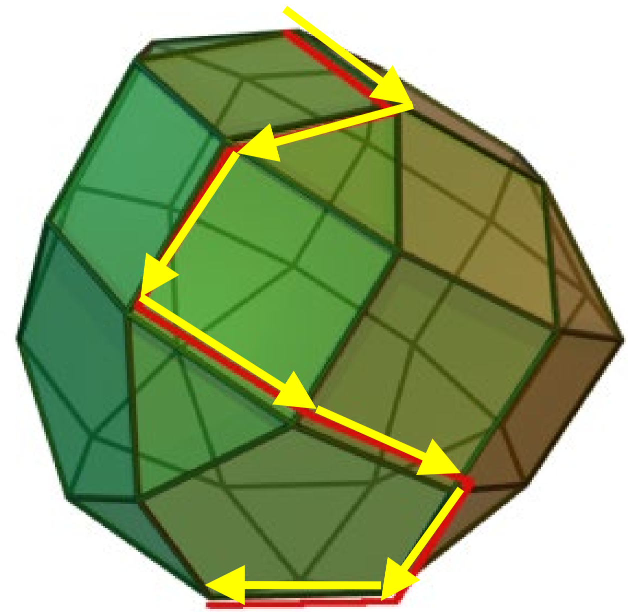
Minimize $z = x_1 + x_2 + x_3$

Subject to constraints: $x_1 - 3x_2 + 4x_3 = 5$

$x_1 - 2x_2 \leq 3$

$2x_2 + x_3 \geq 4$

$x_1 \geq 0$, $x_2 \geq 0$ and x_3 is unrestricted.



Applications: Solving What?

- Geometric Problems
  - Graphics
    - Radiosity, ray tracing
  - Geometric design
    - Shape manipulation, shape reconstruction
  - GIS/GPS
    - Data structure tuning, polygon overlay and update
  - Molecular science
    - Protein Structures, docking, drug design
  - Medical imaging
    - Reconstruction, feature detection/matching
  - Engineering analysis
    - Mesh generation
  - Robotics
    - Configuration space

Applications: Solving What?

- General Problems
  - Data mining, searches, optimization
  - Computer vision
    - Model-based recognition
  - Game, genetics, etc.

How good is your geometry?

Mathematical Thinking

- Step 1: Does a solution exist?
- Step 2: Is there more than one solution?
- Step 3: How many solutions in total?
- However ……, all these can be achieved **WITHOUT** knowing any actual solution

What to know your "Ancestors"?

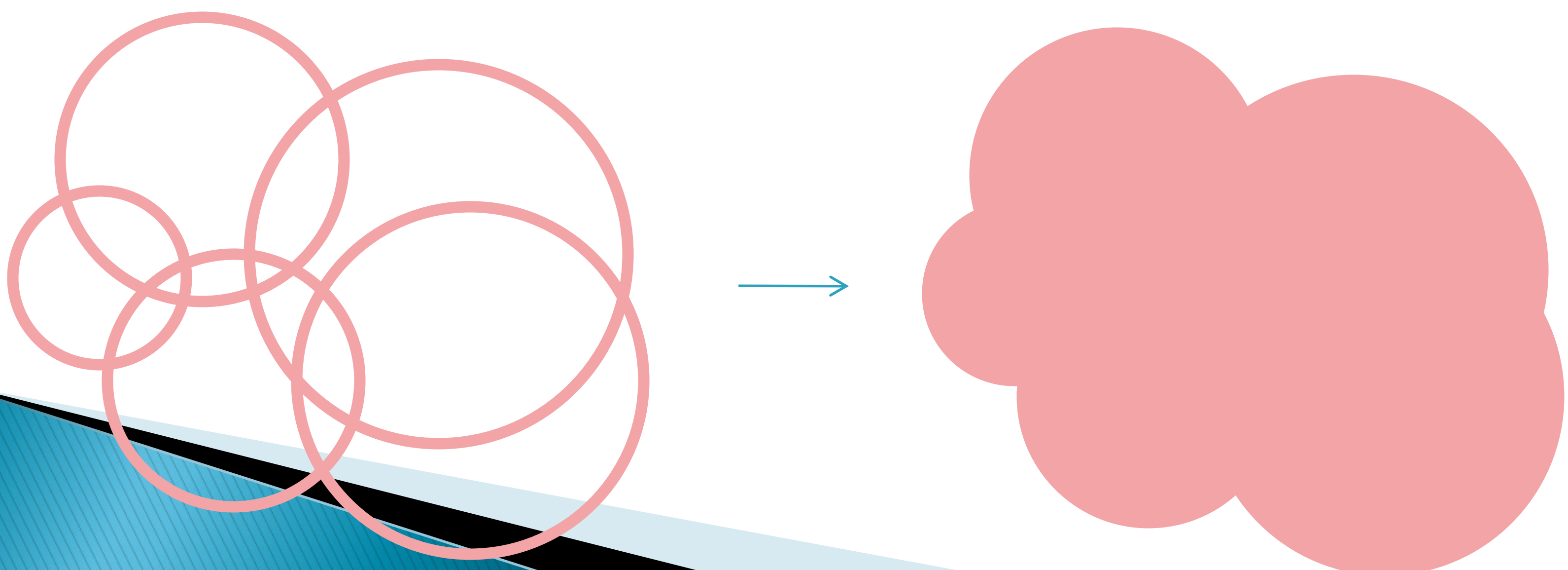
(I mean academic ancestors)

Computational Thinking

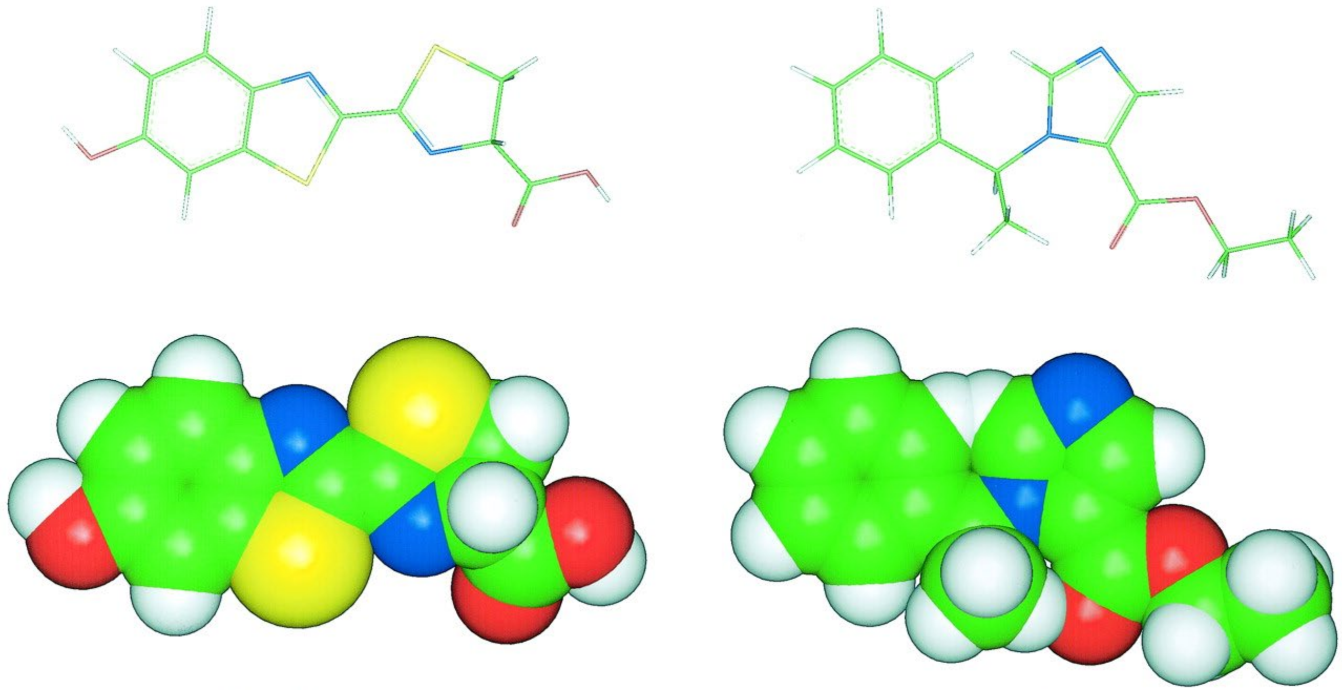
- Is it computable?
- How fast can we compute?
- Example:
  - Given a set of disks

$$B = \{b_i = \{x \mid |x - z_i| \le r_i, \text{for } i = 1..n\}$$

  - Computing the area of union of disks



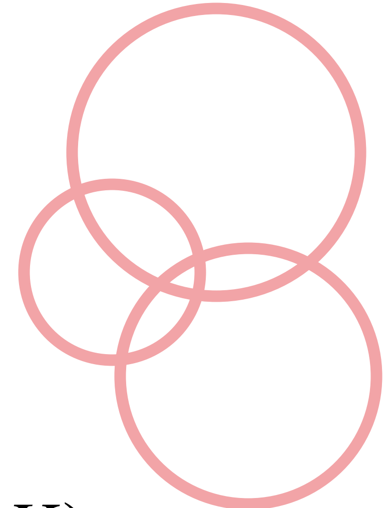
Wait, why do we have to do this



Luciferin Etomidate

- The molecular volume of luciferase is 271 Å^3 and that of etomidate is 286 Å^3.
- Franks and Lieb raises the question of how closely luciferase resembles the actual site of anesthetic action.

Computational Thinking



- Mathematically
  - Inclusion-exclusion formula

$$\text{Area}\left(\bigcup B\right) = \sum_{X \subseteq B} (-1)^{\text{card}(X)-1} \text{Area}\left(\bigcap X\right)$$

  - When $\text{card}(X) =$
    - O(n) 1:
    - O(nC2) 2:
    - O(nC3) 3:
    - 4:
    - ......
    - n:

$$\text{Area}(b_1) + \text{Area}(b_2) + \text{Area}(b_3) + \ldots. \text{Area}(b_n)$$

$$-\left(\text{Area}(b_1 \cap b_2) + \text{Area}(b_1 \cap b_3) + \ldots \text{Area}(b_{n-1} \cap b_n)\right)$$

$$\ldots \text{Area}(b_1 \cap b_2 \cap b_3) + \text{Area}(b_1 \cap b_2 \cap b_4) + \ldots \text{Area}(b_{n-2} \cap b_{n-1} \cap b_n)$$

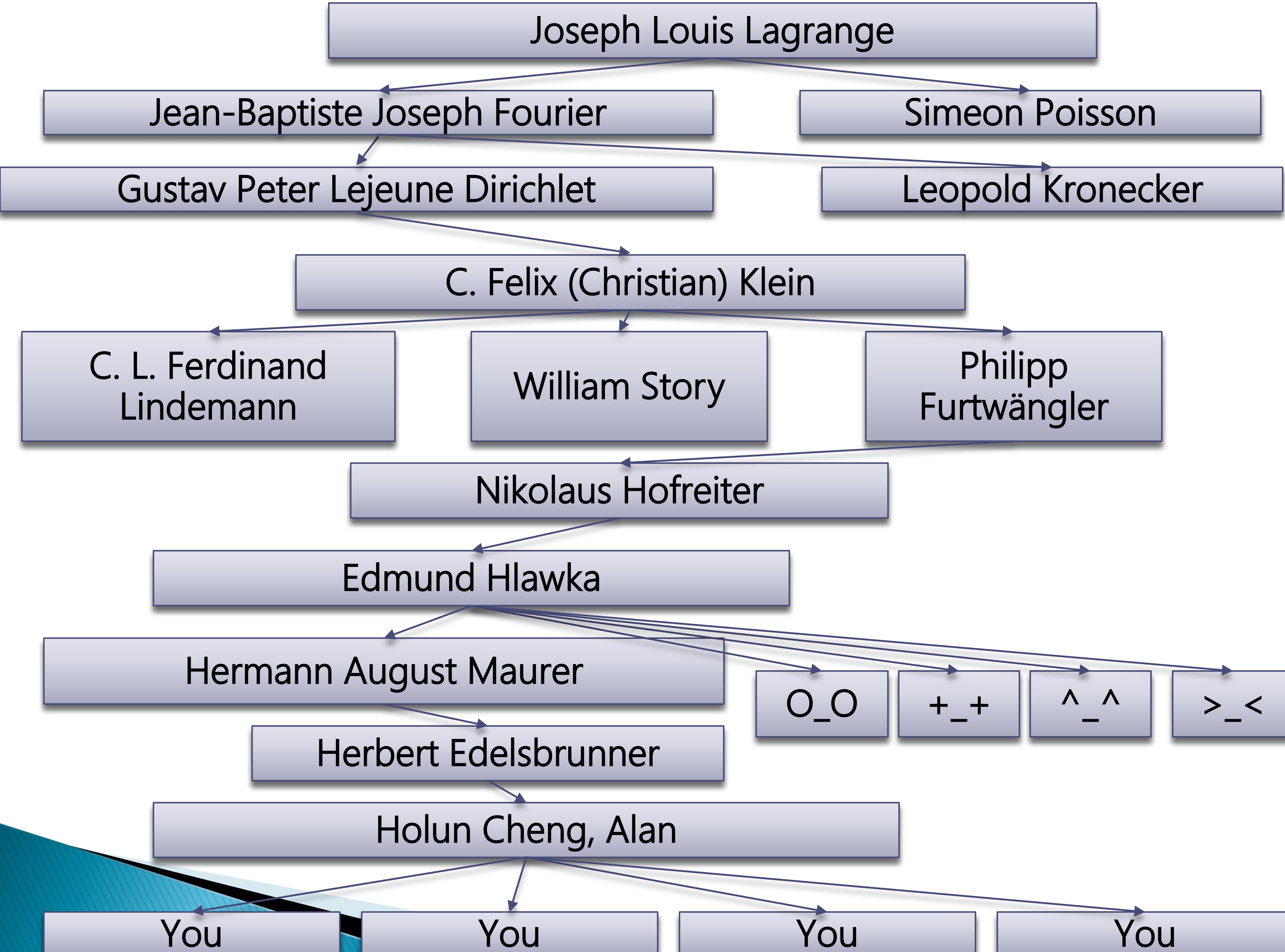
$$\text{Area}(b_1 \cap b_2 \cap b_3 \ldots. \cap b_n)$$

Computational Thinking

- Mathematically
  - Inclusion-exclusion formula

$$\text{Area}\left(\bigcup B\right) = \sum_{X \subseteq B} (-1)^{\text{card}(X)-1}\,\text{Area}\left(\bigcap X\right)$$

  - Complexity $= \mathrm{O}({}_nC_1) + \mathrm{O}({}_nC_2) + \mathrm{O}({}_nC_3) + \ldots + \mathrm{O}({}_nC_n)$
    $= \mathrm{O}(2^n)$
- Computationally, it's a sin!
- Edelsbrunner, $\mathrm{O}(N)$, N is the number of simplices
  - In 2D, N is $\mathrm{O}(n)$
  - In 3D, N is $\mathrm{O}(n^2)$

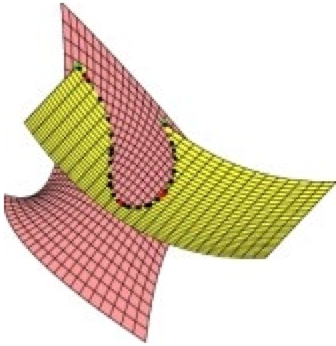

Joseph Louis Lagrange
Jean-Baptiste Joseph Fourier
Simeon Poisson
Gustav Peter Lejeune Dirichlet
Leopold Kronecker
C. Felix (Christian) Klein
C. L. Ferdinand Lindemann
William Story
Philipp Furtwängler
Nikolaus Hofreiter
Edmund Hlawka
Hermann August Maurer
O_O
+_+
^_^
>_<
Herbert Edelsbrunner
Holun Cheng, Alan
You
You
You
You

Convex Hull

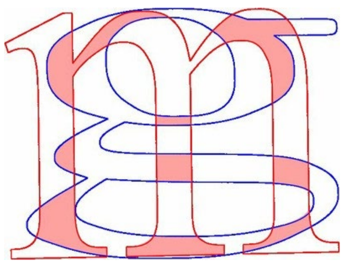
Outline

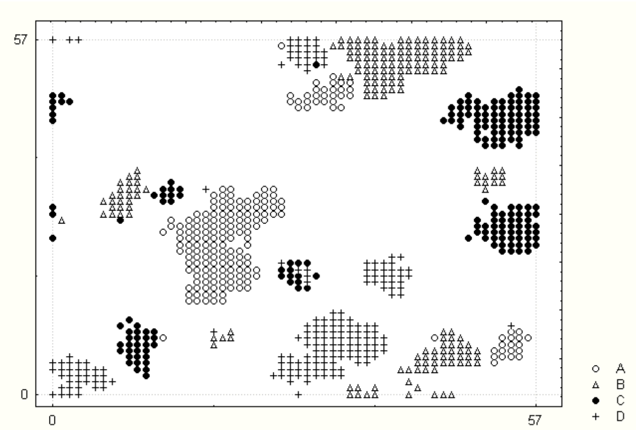
- Definition
- Motivations
- Algorithms in R^2
  - Graham's Scan
  - Jarvis' March (Wrapping)
  - Divide-and-conquer
- Algorithms in R^3
  - Does not look so easy…
  - Quickhull
- Complexity
- Degeneracy

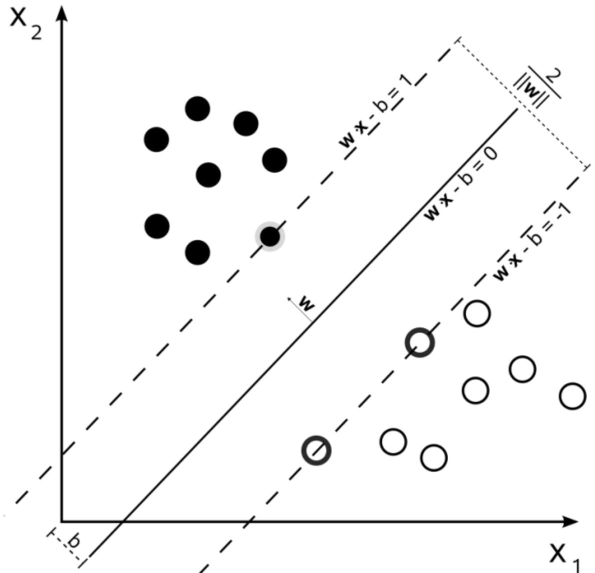
Application of Convex Hull

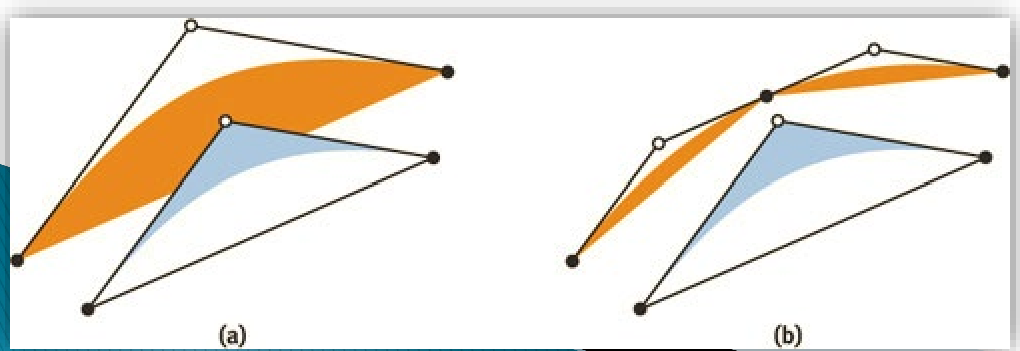


- In general, a "simpler" representation of a complicated object
  - Graphics
    - Bezier curve drawing/intersection, ray tracing
  - Data mining/computer vision
    - Classification of data
  - GIS, path finding, etc…





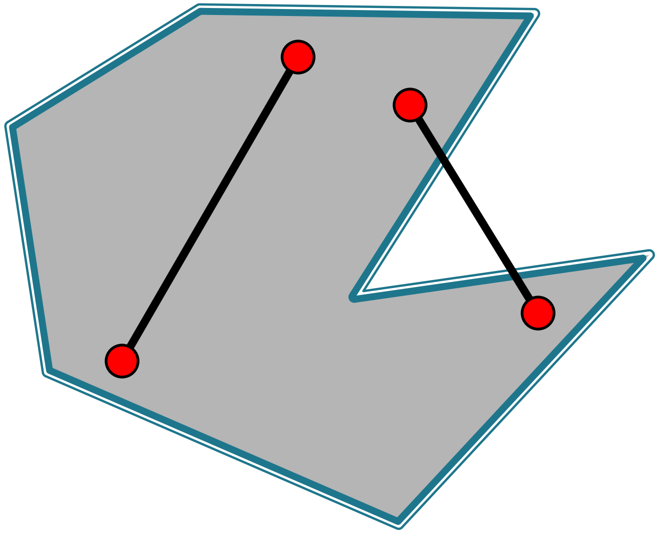




Convexity

A point set P is <u>convex</u> if $\forall x, y \in P, \overline{xy} \subseteq P$

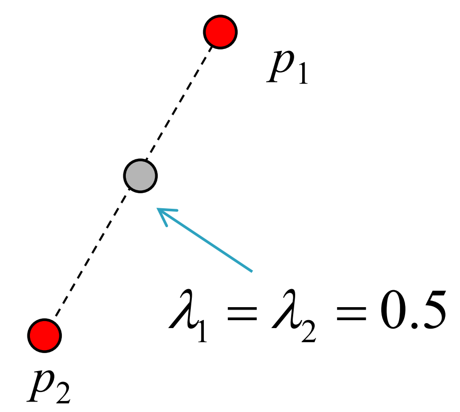
means for all 2 points in P
will have a line that is connecting them that is also still in P



Convex Combinations

- Let $S = \{p_i \in R^d \mid i = 1..n\}$
- An <u>convex combination</u> of S is

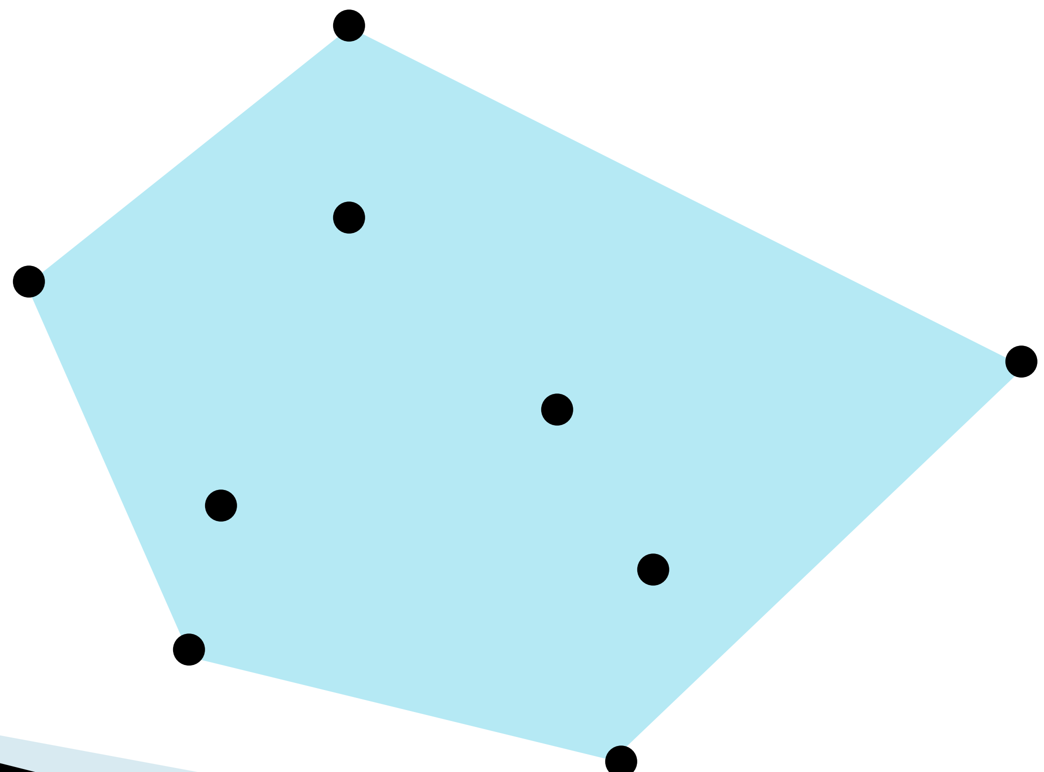
$$\sum_{p_i \in S} \lambda_i p_i \quad \text{such that} \quad \sum_{i=1}^{n} \lambda_i = 1, \forall \lambda_i \geq 0$$



- E.g. $d = 2$, card(S)=2
- The set of ALL the convex combinations?
  - Convex hull!

Convex Hull

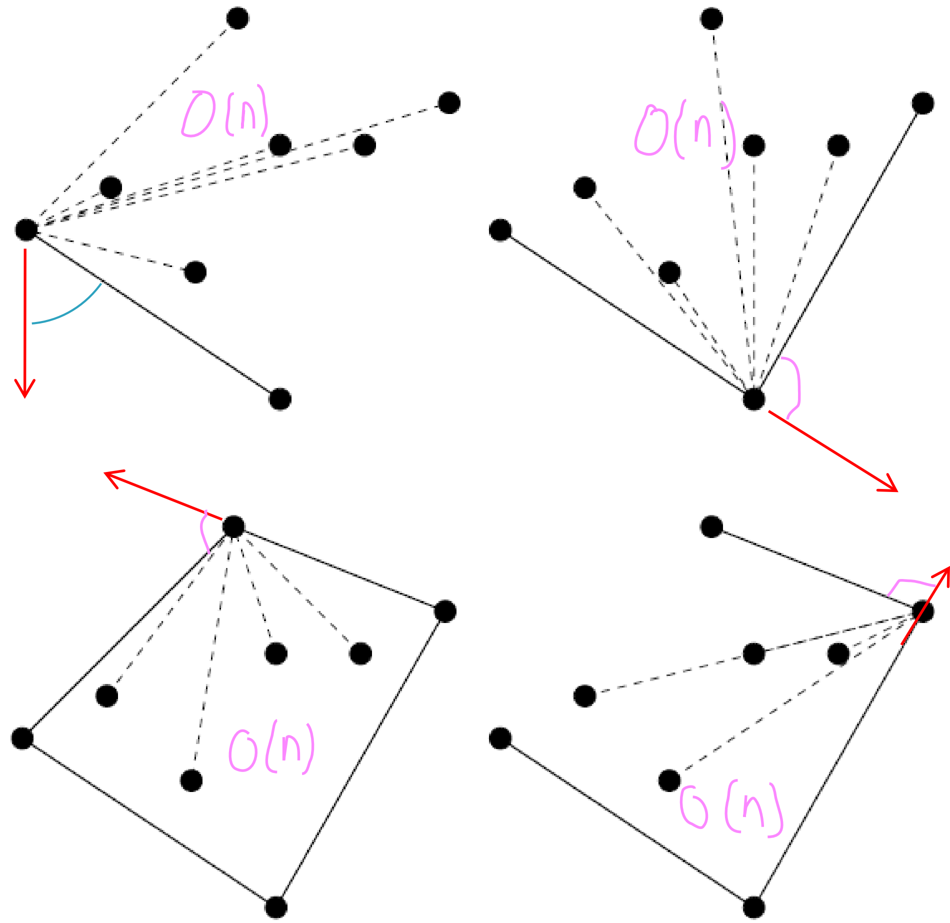
- The <u>convex hull</u> of S, $\mathrm{conv}(S)$, is the collection of all convex combinations



Convex Hull Algorithm

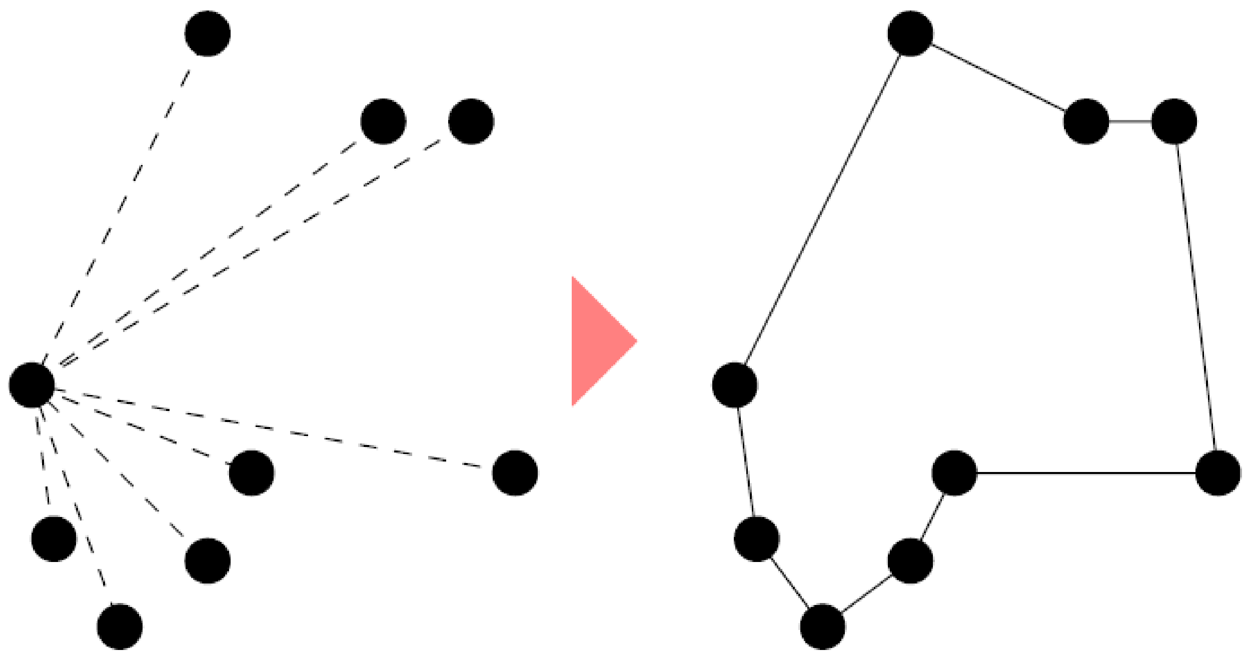
Jarvis' March (Gift Wrapping)

- Take the leftmost vertex
- Repeat
  - Search for the next vertex on the convex hull by choosing the one with the minimal turning angle
- Complexity
  - O(hn)



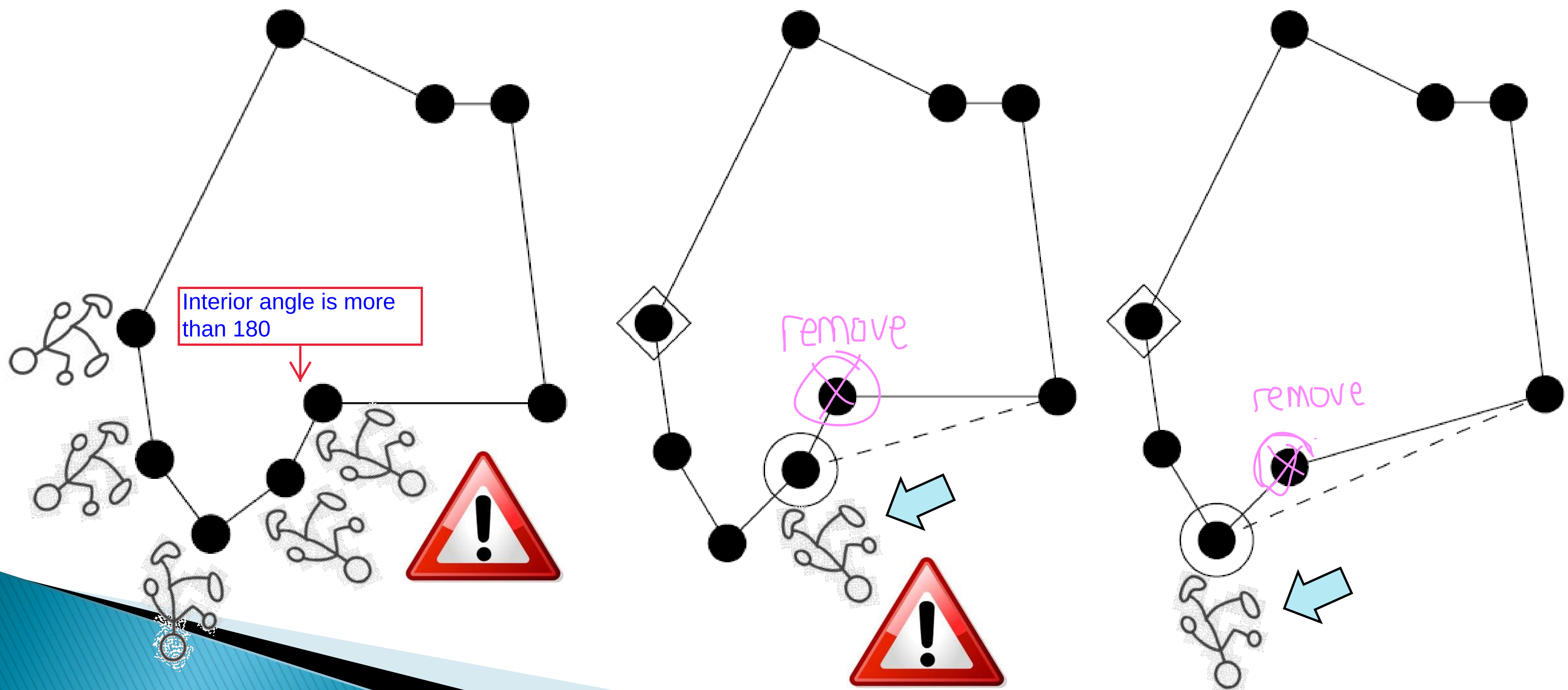
Graham Scan

- Take the left most vertex
- Sort the rest according to their angles
- Connect them in that order and form a polygon



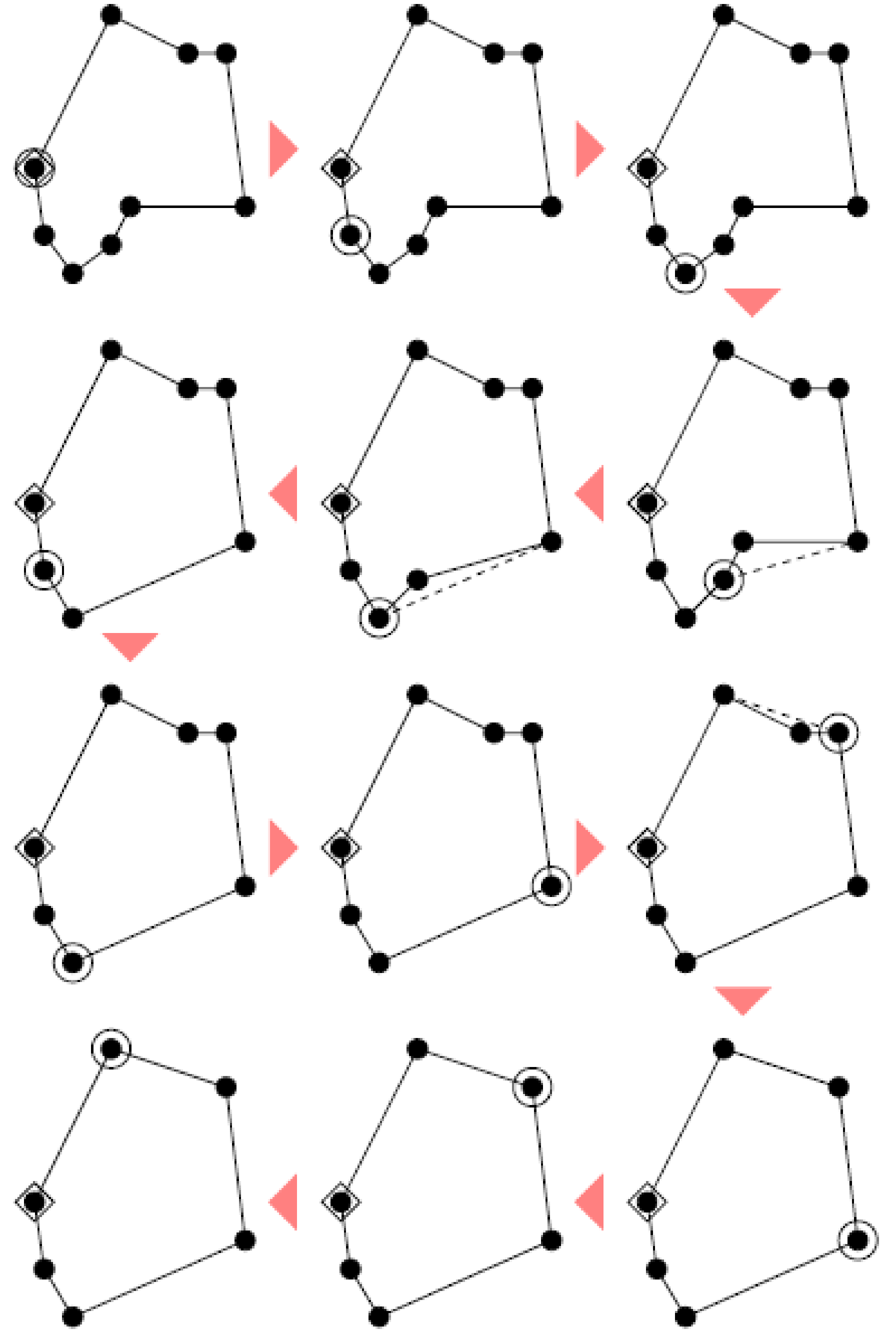
Graham Scan

- Walk around the polygon
- If it's a "convex" corner: continue
- Pit: fill it and FALL BACK! (and check again)



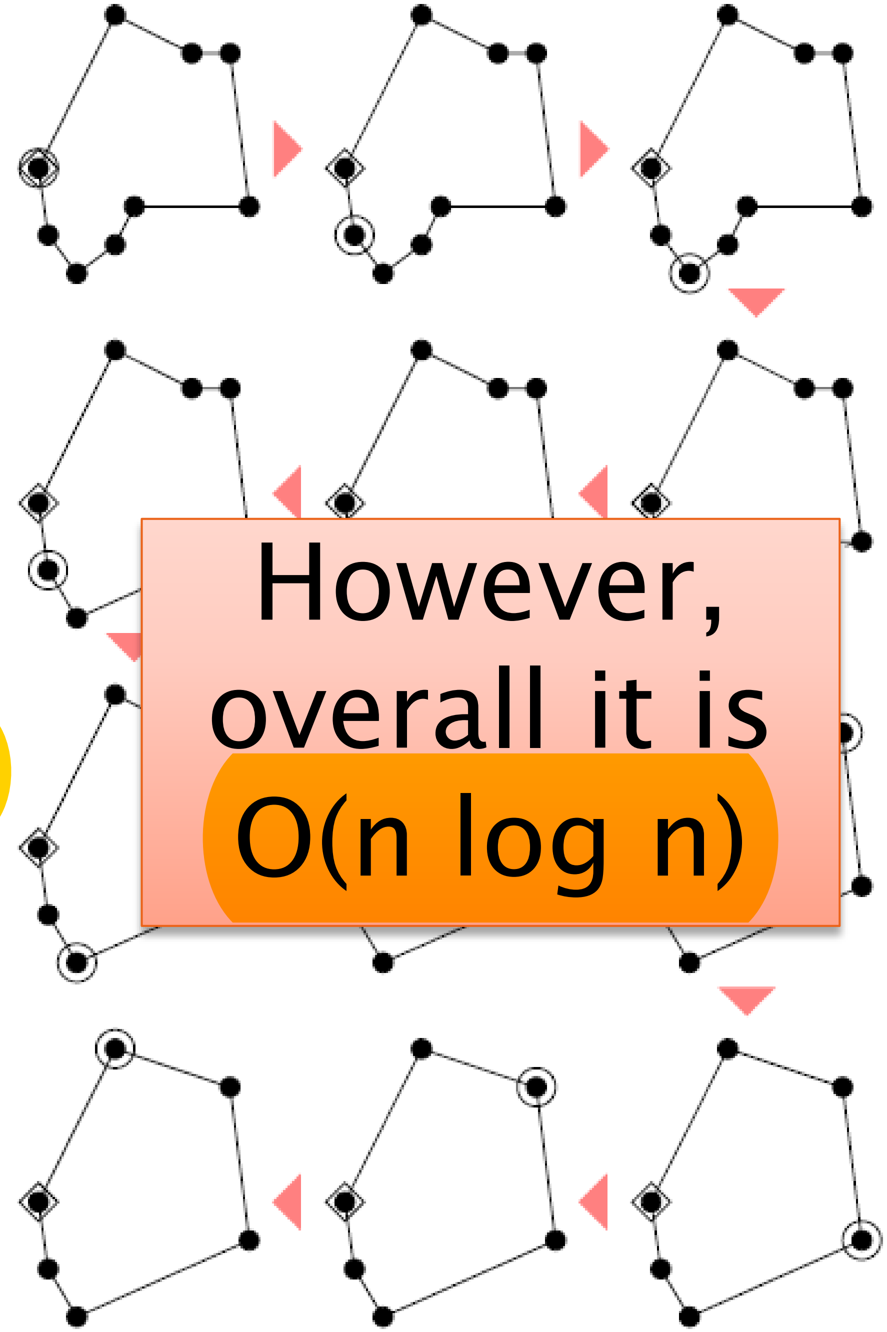
Graham Scan

- Starting from the left most vertex, go around the polygon
- If it is a concave vertex, "make it convex" by "*filling*" it
  - By connecting its two neighbors



Graham Scan

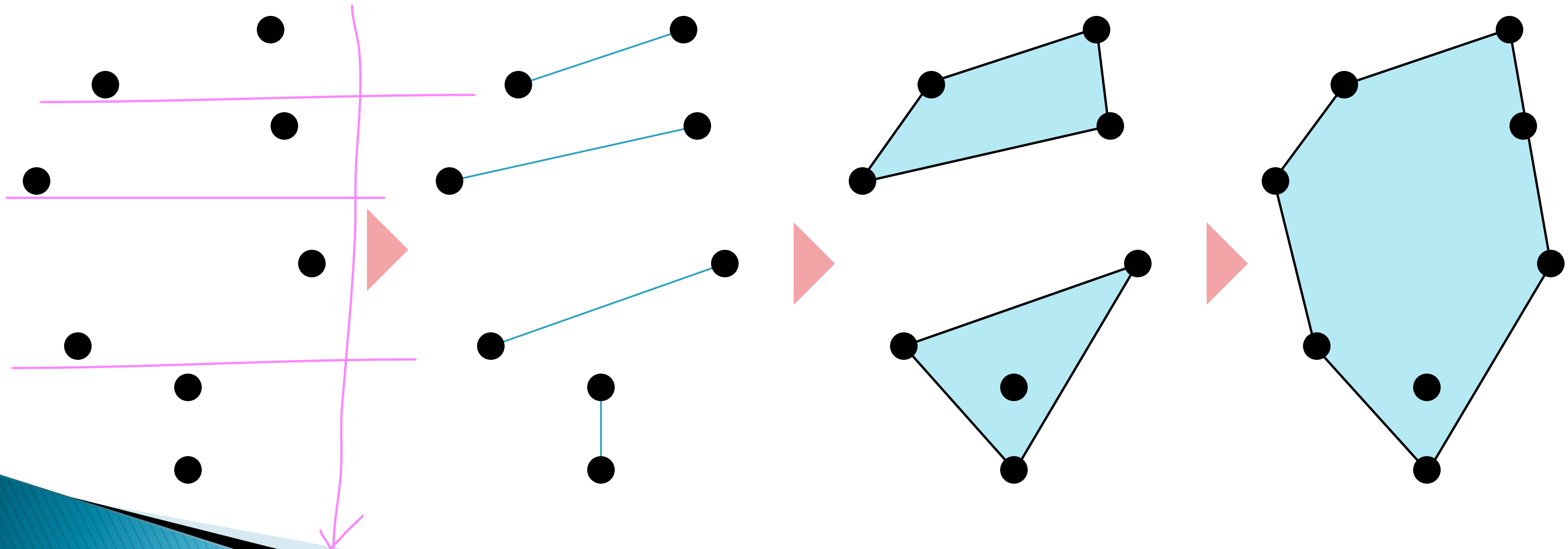
- Complexity?
- How many steps can you "fall back" for each "advance"?
    - O(n)
- And there are at most n advances, does it mean it is O(n²) here?
- No! I argue that it is
    - O(n)
    - Why!?



However, overall it is O(n log n)

Divide and Conquer

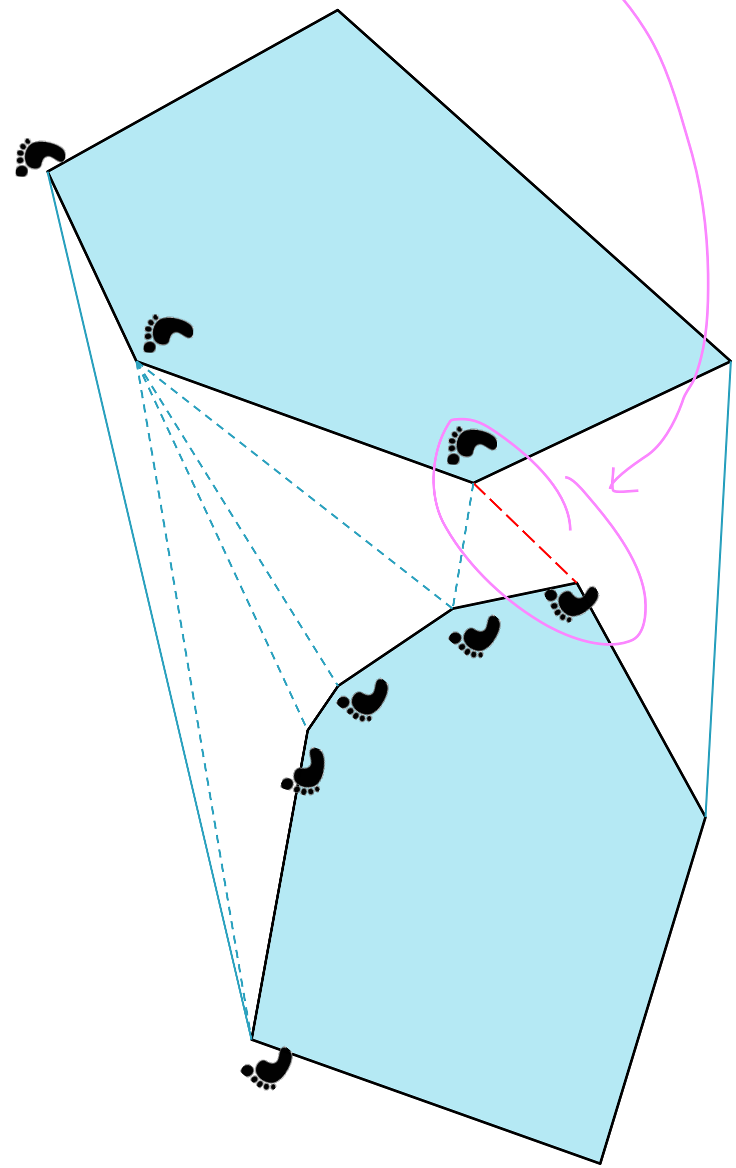
- Sort vertices in a direction, e.g. y direction
- Repeat:
  - Merge every neighboring convex hull



Need to find the 1st line which will not intersect - just find the lowest and highest point

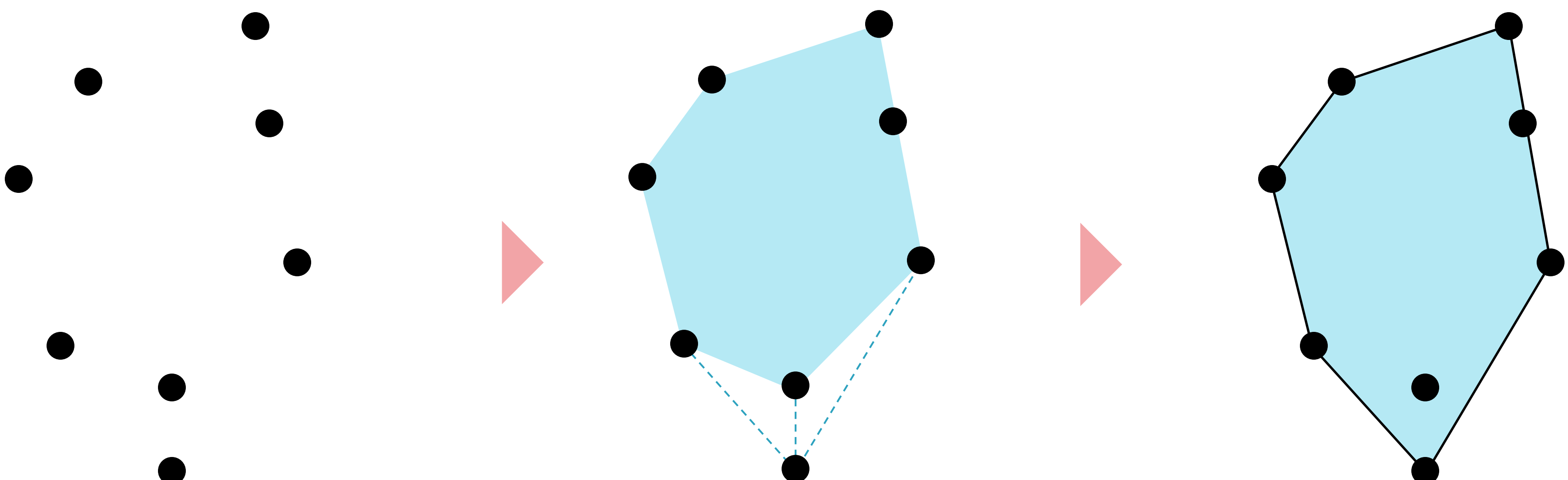
Divide and Conquer

- For every merge, find the two "supporting lines"
  - By taking a walk from an initial line
    - By joining the two neighbors
- Complexities of Graham Scan and D&C can be *inspired* by triangulations



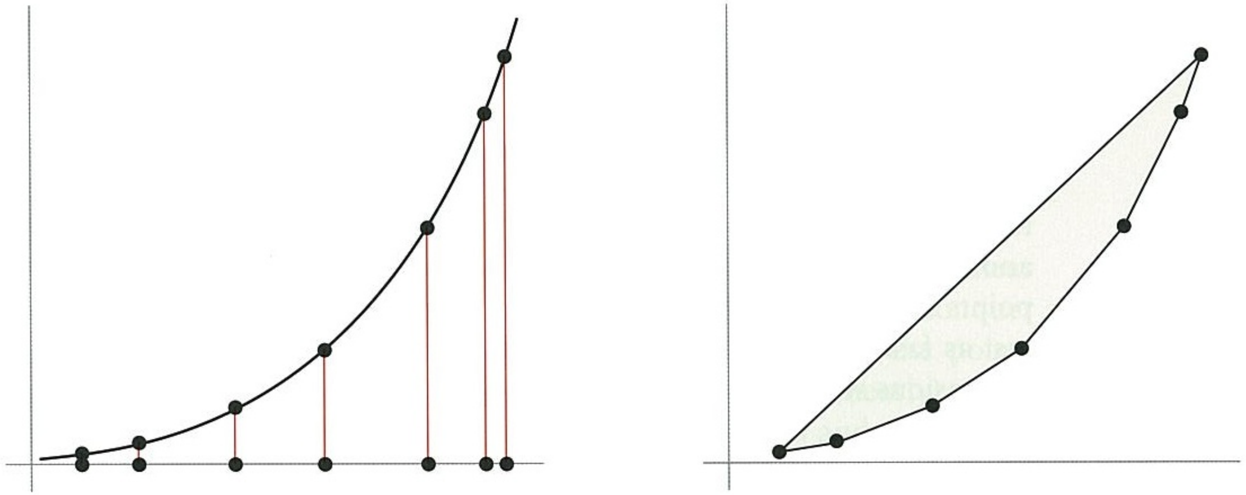
Incremental Method

- Adding a point according to a sorted direction incrementally



Optimal Complexity in R^2

- Why is it O(n log n)?
  - Can we do better than this? Say O(n)?
- We cannot do better than this
  - Otherwise, we can solve sorting problem in O(n) time
    - For n numbers, project them onto a parabola



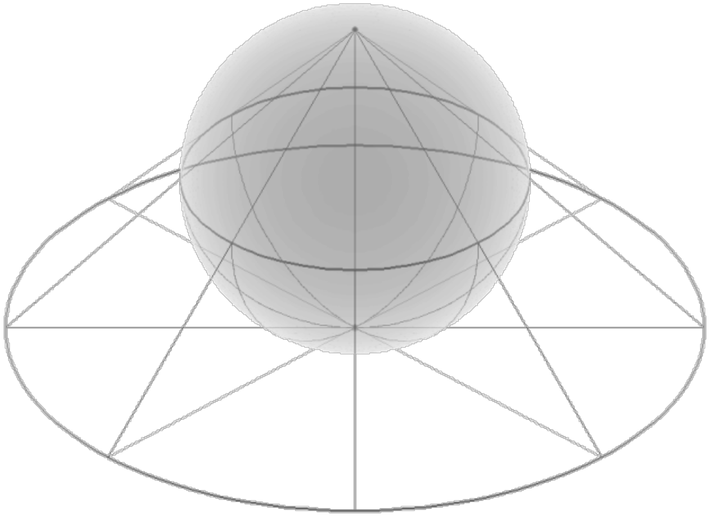
Moving up to 3D

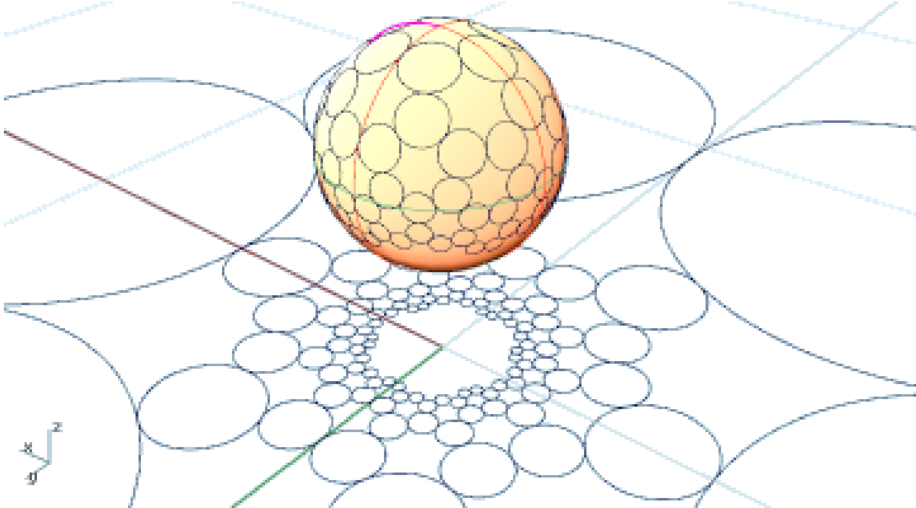
- Gift Wrapping
  - $O(fn)$, f is the number of faces on the convex hull which is $O(n)$
  - Why? Or how many faces are there on the convex hull?

Topology Time!

- How many faces are there on a 3D convex hull?
- One-to-one mapping onto the plane by
  - Project a convex hull to a sphere
  - Stereographic projection

therefore by doing this projection, then we can prove that there is only O(n) maximum





Moving up to 3D

- Graham Scan
  - There is no known equivalence in 3D
  - (According to Joseph O'Rourke 2011)
- Divide and Conquer
  - If we think in a tetrahedralization way, it would be $O(n^2)$
  - Because there could be $O(n^2)$ tetrahedra for n points
    - (Construct a case?)

Divide and Conquer in 3D

- Expected time
  - O(n log n)

Incremental Method in 3D

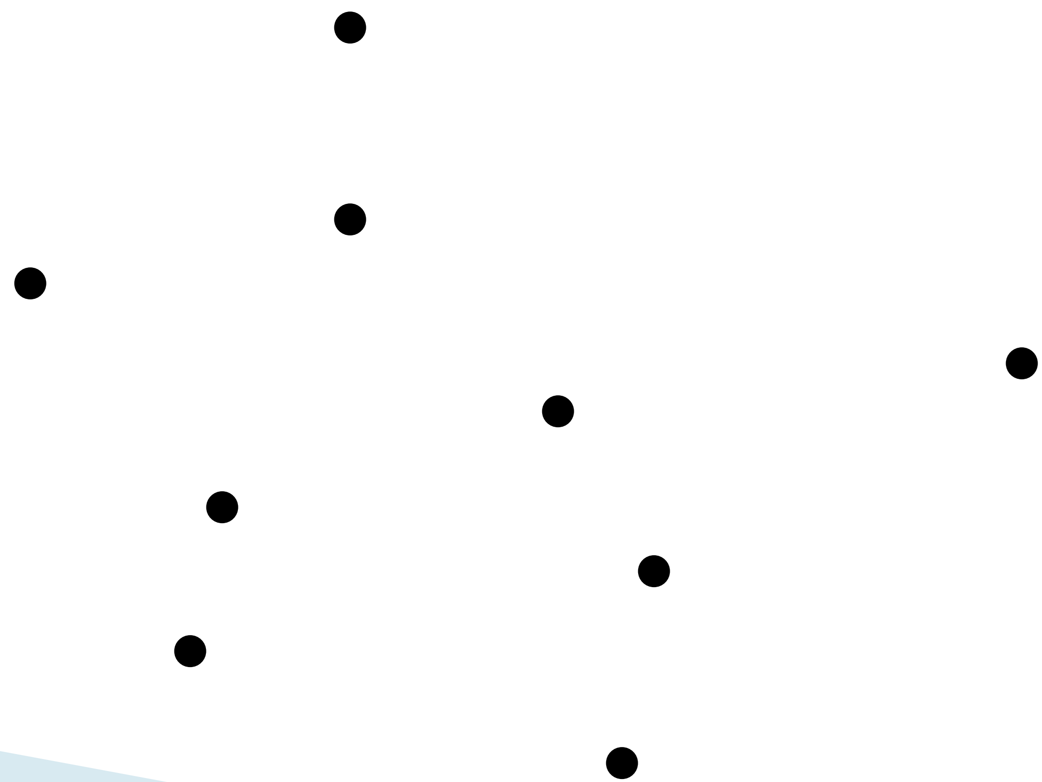
- Expected time O(n log n)
  - With some complicated data structure

Some Comments for D&C CH

- "*The first full description for 3-d convex hulls appeared in Edelsbrunner's book and was 15 pages long.*"
- "*Despite the asymptotic advantage of this algorithm over the incremental algorithm, the delicacy of implementing the wrapping and updating the surface data structure has left this algorithm theoretically important but not the pragmatic choice*"
  - O'Rourke
- "Indeed, its implementation appears to require not only a planar subdivision structure capable of the necessary stitching operations, but also the handling of some very tricky details for the bridge computation."
  - Timothy Chan

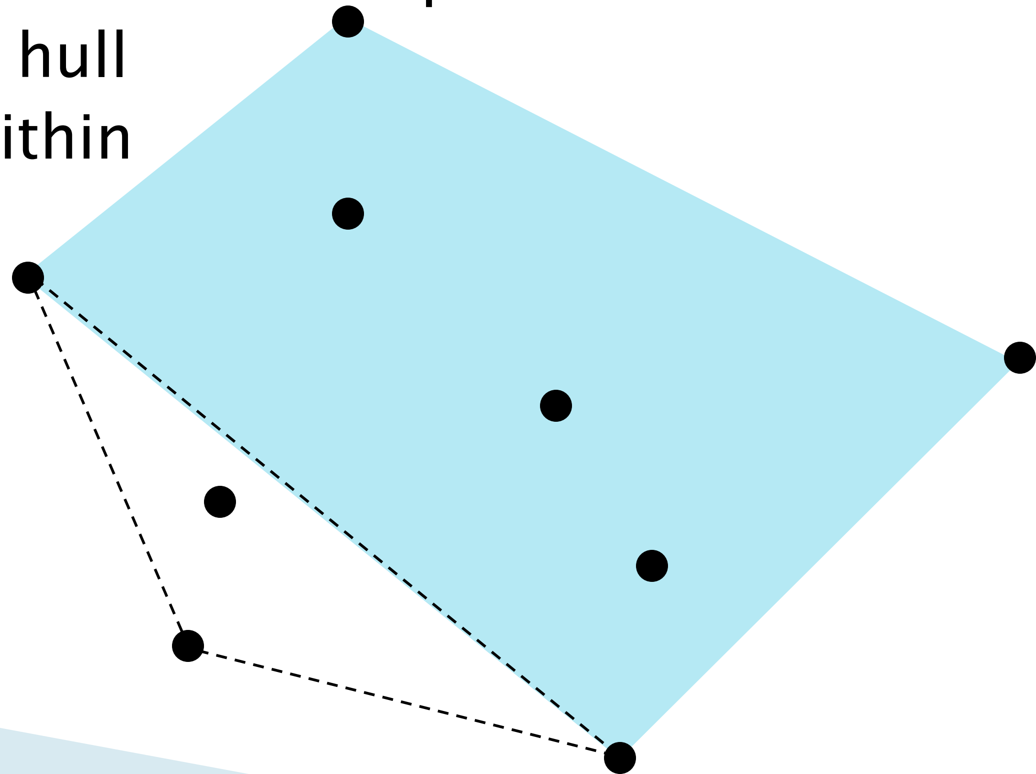
Quickhull

- General idea: discard points that are not on the hull as quickly as possible
- First find the maximum and minimum in x and y directions



Quickhull

- Construct a quadrilateral by these four points
- Discard any point inside
- Repeat
  - For each side, find the furthermost point
  - Include in the convex hull
  - Eliminate any point within
- Worst case
  - $O(n^2)$ (2D, 3D)
  - (Give an example?)



Higher Dimension

- In general:

$$O(n^{\lceil d/2 \rceil - 1} + n \log n)$$

- Because of the complexity of the number of "faces" on a convex hull

Degeneracy

- More than two points that are collinear
  - Especially on the boundary
  - Can be corrected by interchanging the “less than” and “less than or equal to” signs of the left turn test
- Same coordinates in sorting
  - “Slightly” rotate all the points by an infinitely small angle
  - ⇒“If they have the same x values, compare their y values”
  - Beginning of **perturbation**