§3 Average-Case Analysis

- Motivation: Incrementing a Binary Counter
- Example: naïve QuickSort
- Simplex Algorithm
- *Smoothed* Analysis

pivot choice

Motivation: Binary Increment

Cost(n) := #bit flips of n-bit binary counter

```
000…000
000…001
000…010
000…011
000…100
000…101
      ……
111…111
```

Depends on *value* N of the counter

Worst-case: $O(n)$ bits flip

$$= \max_{N<2^n} \#\text{bitflips}(N)$$

Conservative estimate, rare case

Average case cost:

$$2^{-n} \cdot \sum_{N<2^n} \#\text{bitflips}(N)$$

$$= \boxed{1} + \boxed{1/2} + 1/4 + \ldots < 2$$

Example: naïve QuickSort

Cost(n) := #comparisons/swap operations

Classical recurrence:

$$T(n) = T(j) + T(n-j) + O(n)$$

$j = 1 \quad \Rightarrow$

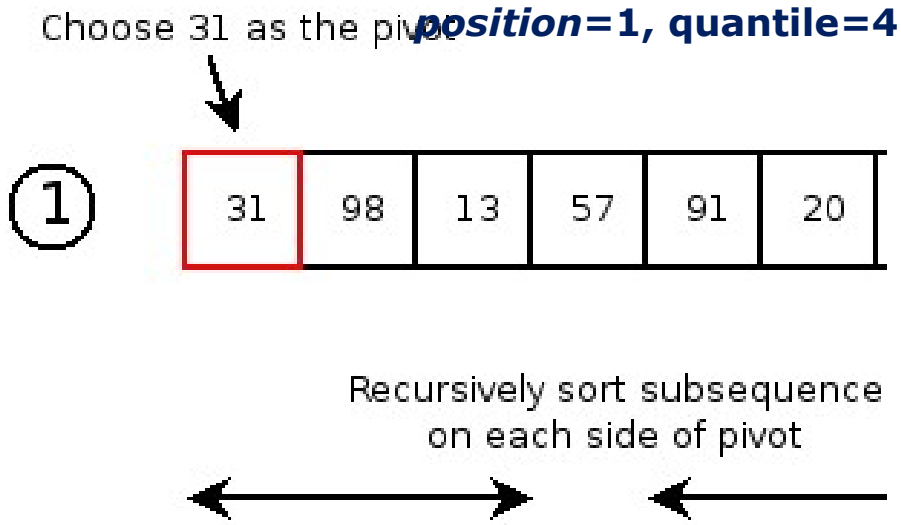
$$T(n) = O(n^2)$$

Average-case recurrence:

$$T(n) = O(n \cdot \log n)$$

$$T(n) = 1/n \cdot \sum_j \left(T(j) + T(n-j) + O(n)\right)$$

j = **quantile** of chosen pivot **position** (e.g. middle/first/last array element)



Quiz

a) Why is *best-case* **not** a common mode of algorithm analysis? Explain!

b) Verify that $T(n) = O(n \cdot \log n)$ indeed solves the below average-case recurrence.

Average-case recurrence:

$$T(n) = 1/n \cdot \sum_j T(j) + T(n-j) + O(n)$$

c) Does $T(n) = O(n)$ solve the recurrence?

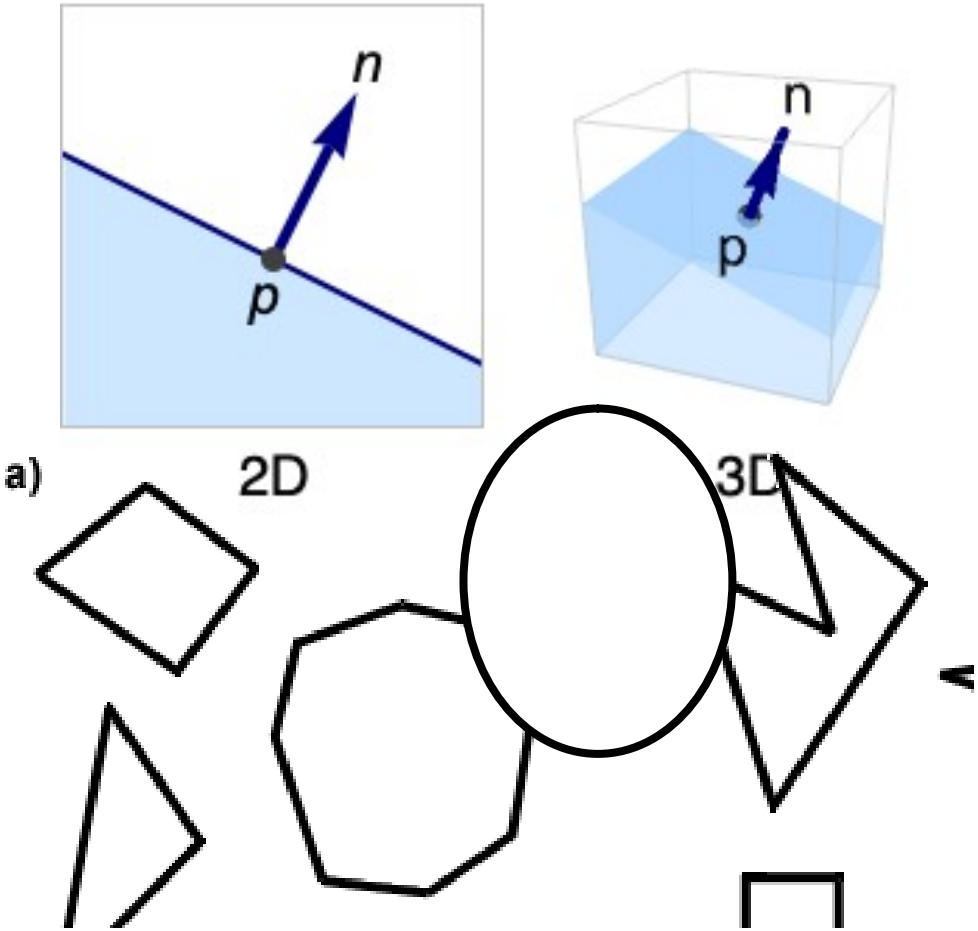
Convex Geometry

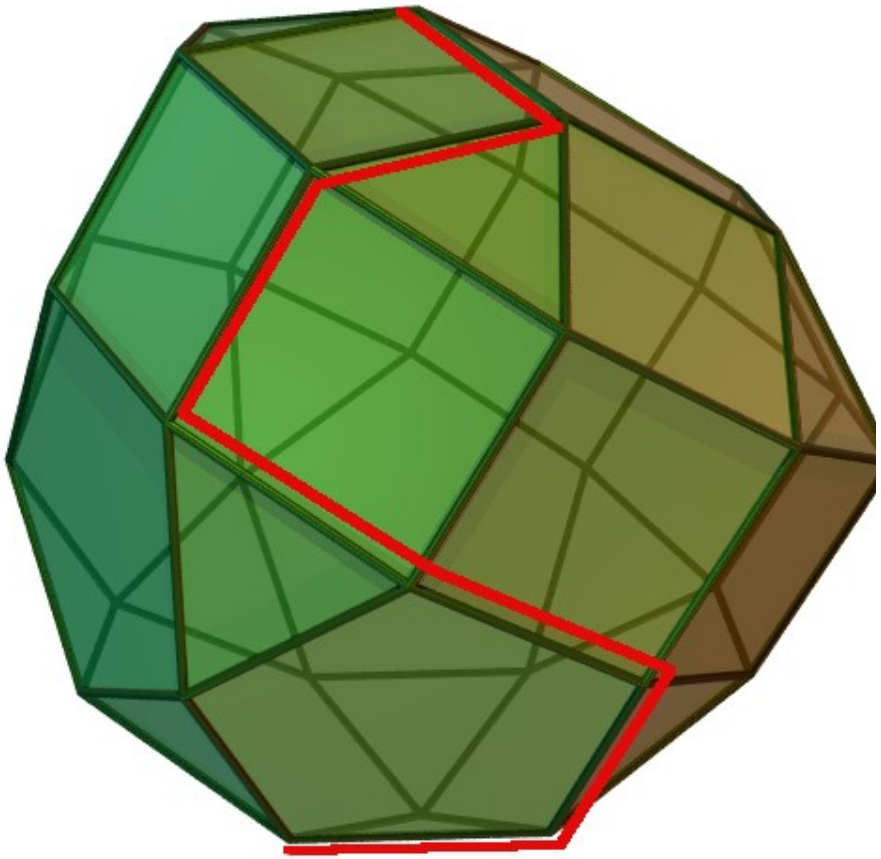
n affine halfspaces in $\mathbb{R}^d$.

parameterized by $\bar{A} \in \mathbb{R}^{d \times n}$, $\underline{b} \in \mathbb{R}^n$

intersection

i.e. $\{ \bar{u} \ : \bar{A} \cdot \bar{u} \leq \underline{b} \}$ componentwise



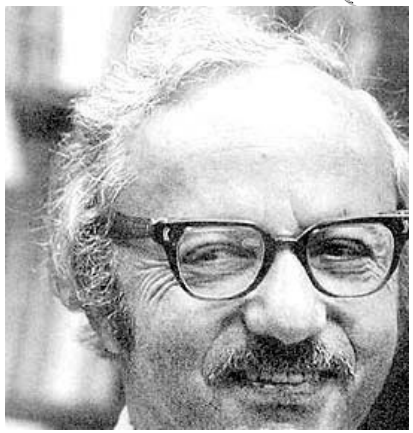


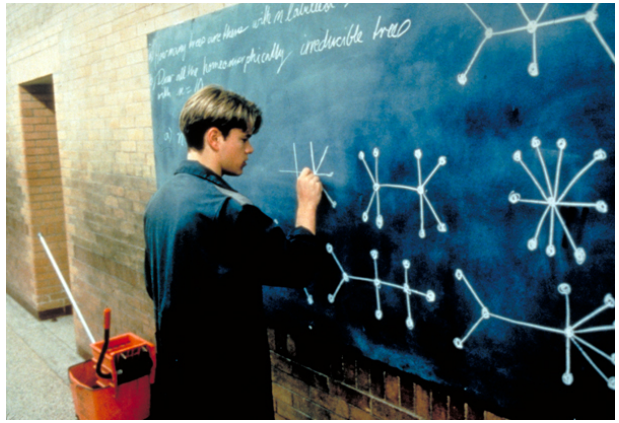
Simplex Algorithm

Input: n affine halfspaces in $\mathbb{R}^d$. parameterized by $\bar{A} \in \mathbb{R}^{d \times n}$, $\underline{b} \in \mathbb{R}^n$

Output: largest x-value in intersection

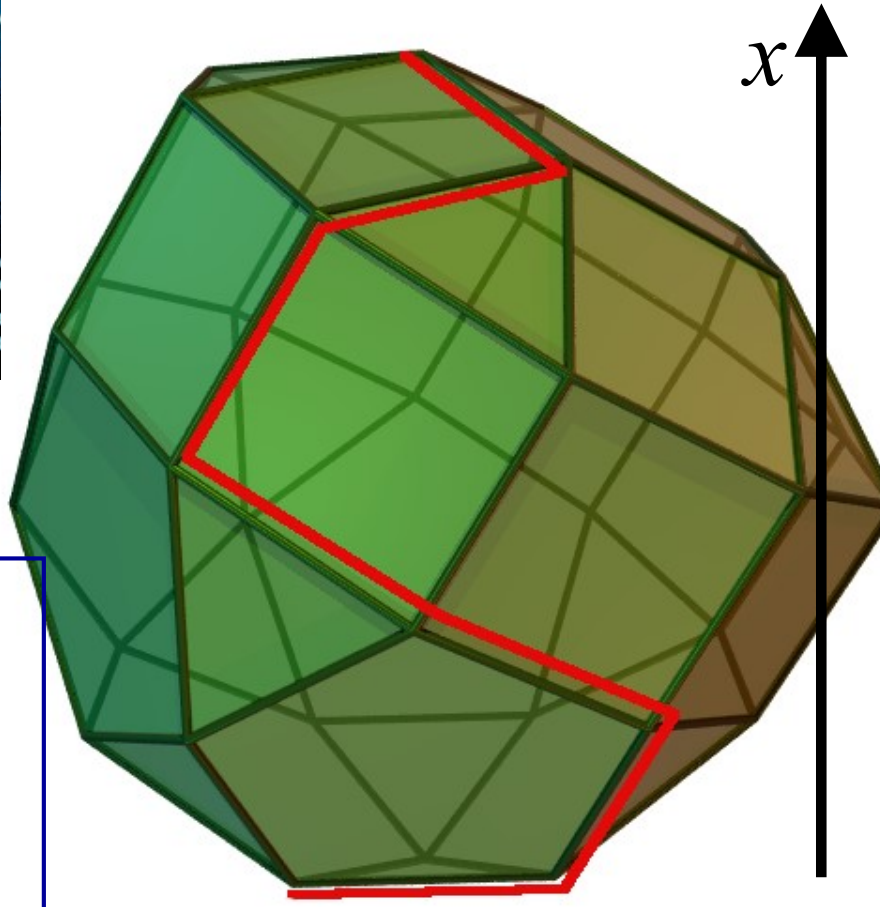
i.e. $\max \{ \bar{u}_1 : \bar{A} \cdot \bar{u} \leq \underline{b} \}$





George Dantzig
(1914~2005)

1946/47



0) Start at some vertex.

1) Move to *highest* neighbor

2) Repeat (1), while possible.

Simplex Algorithm

Input: n affine halfspaces in $\mathbb{R}^d$.

Output: largest x-value in intersection

Questions: a) Does there always <u>exist</u> a path of length $\leq n^{2+\log d}$

Gil Kalai & D.Kleitman 1992

b) Does the *local* rule (1) <u>find</u> it?

Answer a) famously unknown!

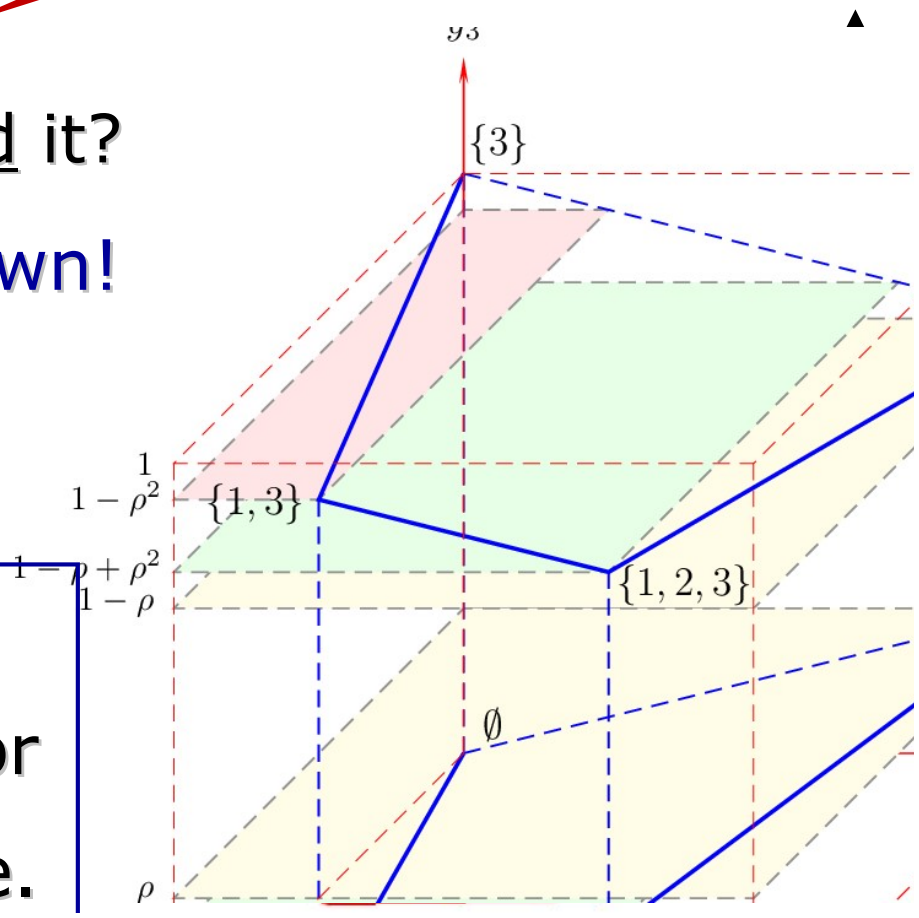
b) No! [Klee&Minty'1973]

b') still no!

0) Start at some vertex.

1) Move to *shadow* neighbor

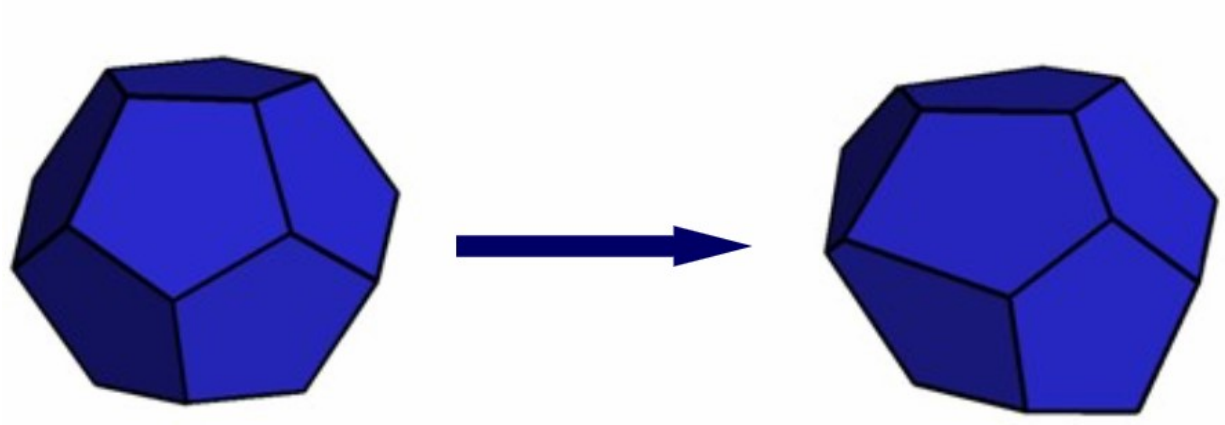
2) Repeat (1), while possible.



Smoothed Analysis

Input: n affine halfspaces in $\mathbb{R}^d$.

Output: largest x-value in intersection, *approximately*



Spielman&Teng JACM **51** (2004):

After a random Gaussian perturbation of variance σ, the *shadow-boundary* pivot rule makes a number of steps polynomial in d, n, and $1/\sigma$.

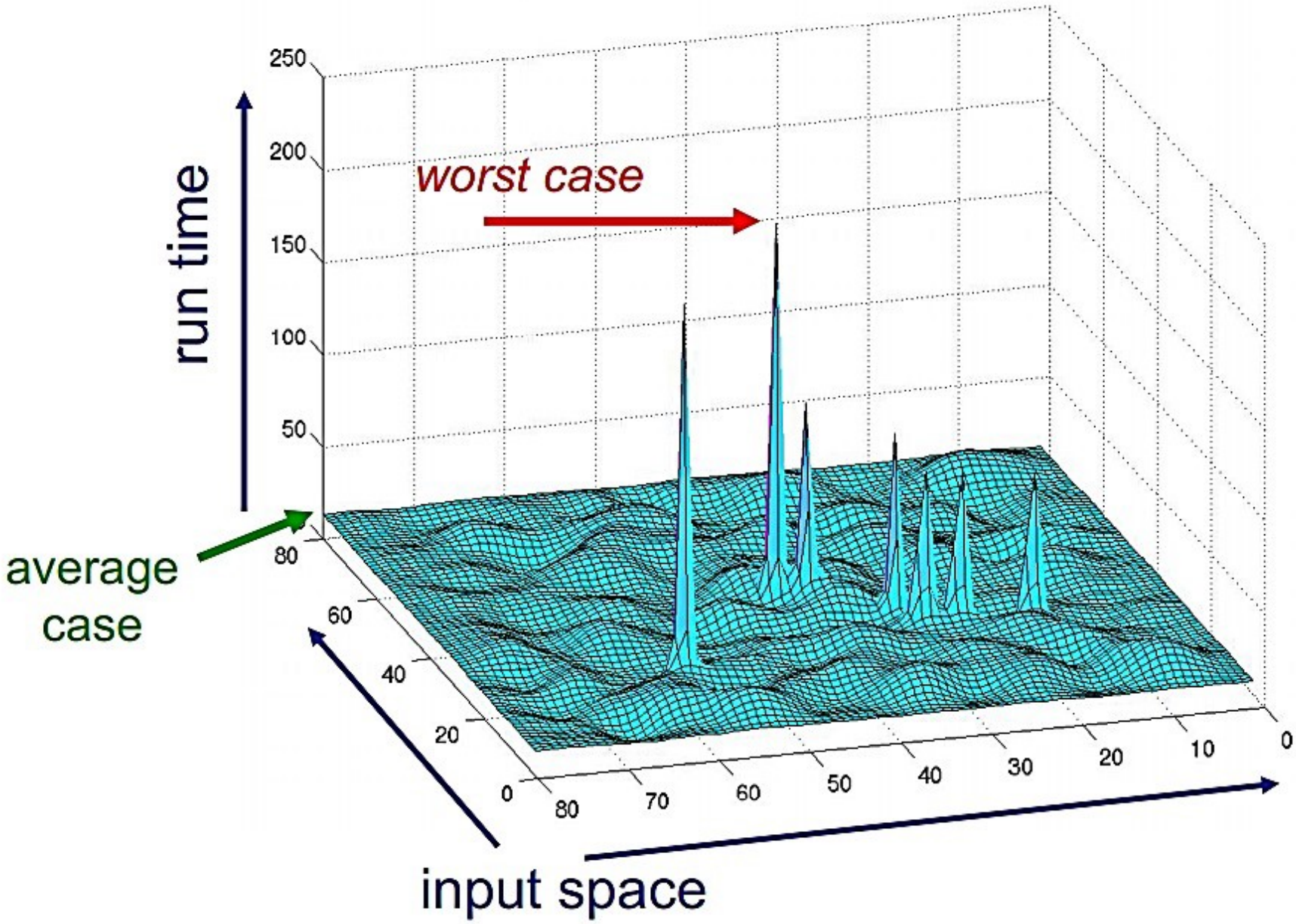
0) Perturb the halfspaces *slightly*

0) Start at some vertex.

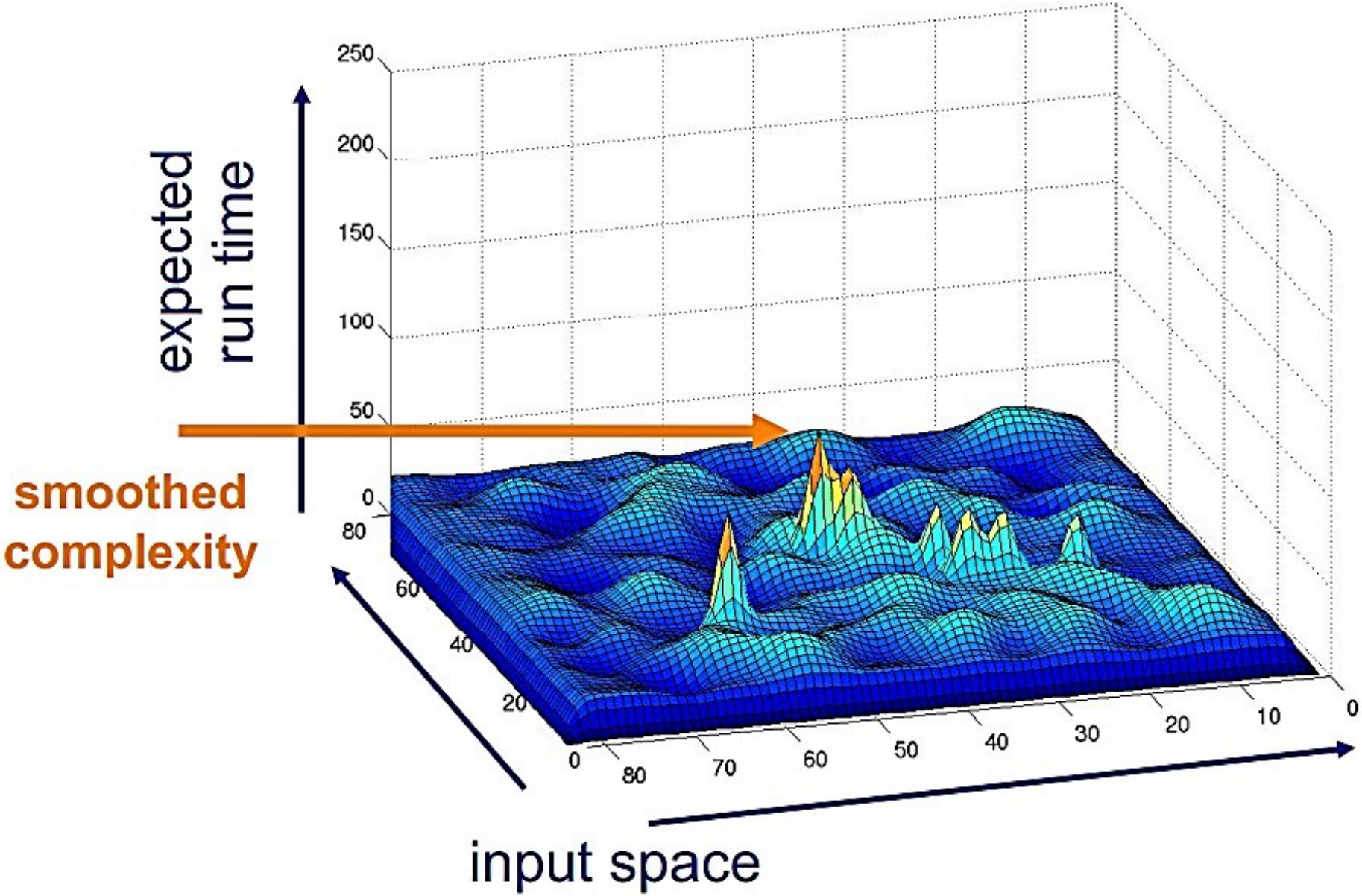
1) Move to *highest* neighbor

2) Repeat (1), while possible.

Worst-Case vs. Average-Case Analysis



Smoothed Analysis



§3 Recap

- Motivation: Incrementing a Binary Counter
- Example: naïve QuickSort
- Simplex Algorithm
- *Smoothed* Analysis

pivot choice

"Smoothed analysis is a hybrid of worst-case and average-case analyses that inherits advantages from both."