CURS 10 — 5.04.2016

GEOMETRIE COMPUTAȚIONALĂ

• $d = 1$ coordonata x

regiune fezabilă = intersecție de intervale

maximizarea se face la unul din capetele intervalului

• $d=2$ coordonate x și y

Exemplu (curs anterior! → pag 3,4)

— ALGORITM LPMARG2D $(H, \vec{c}, m_1, m_2)$ —

• Notații

→ $H = \{h_1, h_2, \dots, h_n\}$ → mulțimea de semiplane

→ Fie $M \gg 0$ fixat

Fie $c = (c_x, c_y)$ → vectorul care dă funcția obiectiv

• $m_1 : \begin{cases} x < M, & \text{dacă } c_x > 0 \\ x \geq -M, & \text{dacă } c_x \leq 0 \end{cases}$

• $m_2 : \begin{cases} y < M, & \text{dacă } c_y > 0 \\ y \geq M, & \text{dacă } c_y \leq 0 \end{cases}$

În continuare lucrăm cu $\vec{c} = (0,-1)$. A maximiza funcția obiectiv revine la a găsi punctul „cel mai de jos"!.

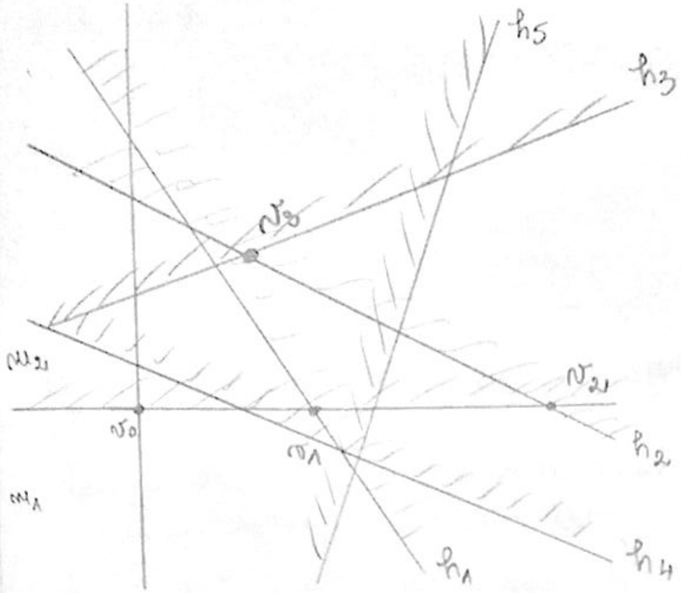
→ $C_0 \overset{NOT}{=} m_1 \cap m_2$

→ $H_i = \{m_1, m_2, h_1, \dots, h_i\}$

$C_i \overset{NOT}{=} m_1 \cap m_2 \cap h_1 \cap \dots \cap h_i \quad (i = 1, \dots, n)$

În particular, are loc, $C_0 \supseteq C_1 \supseteq C_2 \supseteq C_3 \supseteq \dots \supseteq C_n$

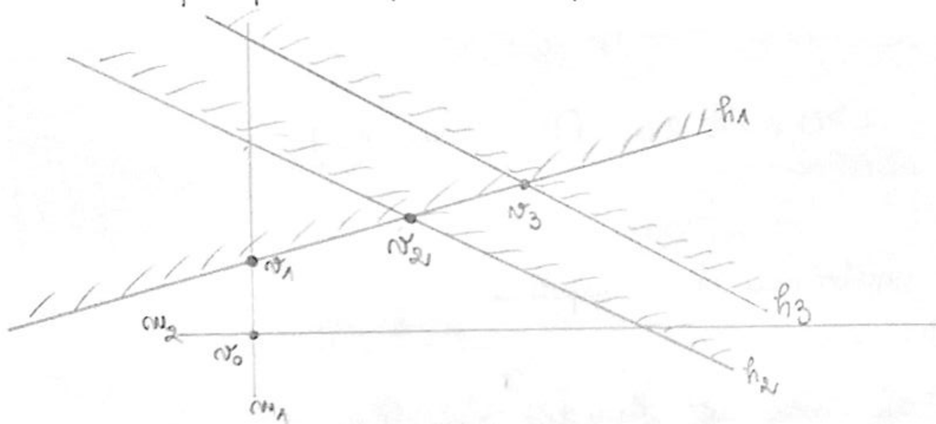
regiune fezabilă



dacă se trece prin PASUL 6

$\Rightarrow O(n^2)$, dacă nu $O(n)$

Pentru fiecare i, regiunea poligonală convexă C_i admite un vârf optim pentru funcția obiectiv, notat cu v_i.



Fapt

Determinat v_{i-1}, sunt două posibilități:

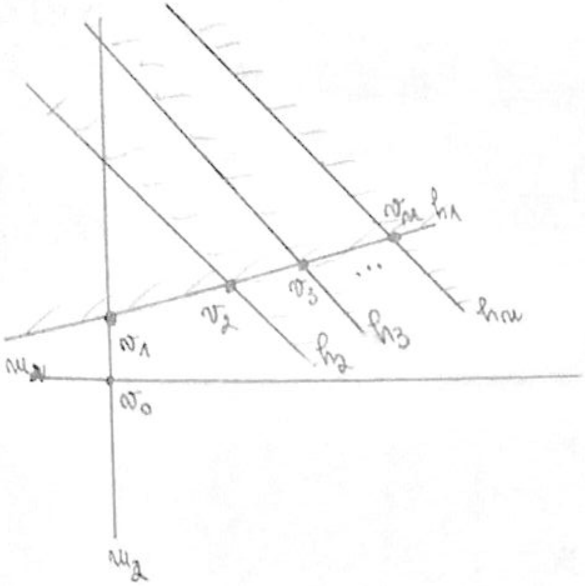
$v_{i-1} \in h_i \rightarrow$ pasul 5

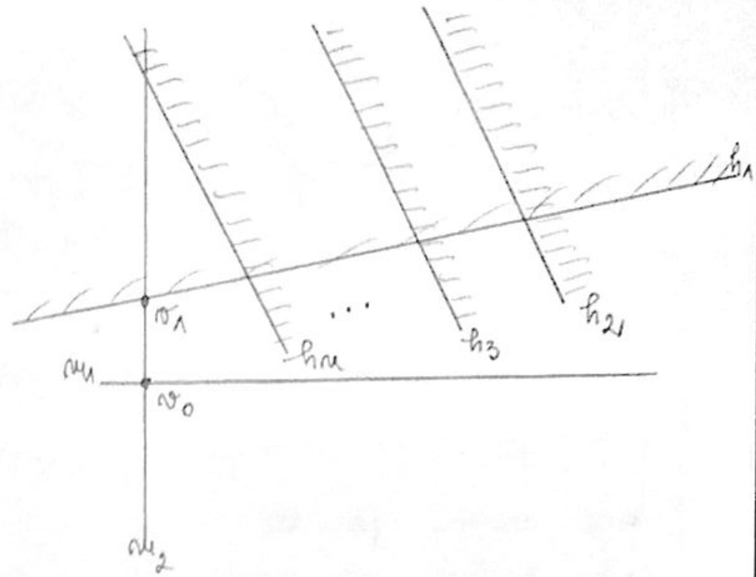
$v_i \leftarrow v_{i-1}$

$v_{i-1} \notin h_i \rightarrow$ pasul 6

trebuie rezolvată o problemă de programare liniară unidimensională

$\Rightarrow$ complexitatea $\underset{\text{(pasul 6)}}{\sum O(i)} = O(n^2)$





↑ trece prin pas 6

— PROGRAMARE LINIARĂ —

ALGORITM → RANDOMIZAT
→ ALEATOR
→ PROBABILISTIC

Semiplanele sunt introduse aleator (v. pasul 2)

Fie X_i variabilă aleatoare:

$$X_i = \begin{cases} 0, & \text{dacă } v_{i-1} \in h_i \text{ (ales pasul 5)} \\ 1, & \text{dacă } v_{i-1} \notin h_i \text{ (ales pasul 6)} \end{cases} \Rightarrow \text{timpul total } \sum_{i=1}^{n} X_i \, O(i)$$

Valoarea așteptată (timpul mediu):

$$E\left(\sum_{i=1}^{n} X_i \, O(i)\right) \xrightarrow{\text{altă not.}} \mu\left(\sum_{i=1}^{n} X_i \cdot O(i)\right) = \sum_{i=1}^{n} O(i)\,\mu(X_i) \le$$

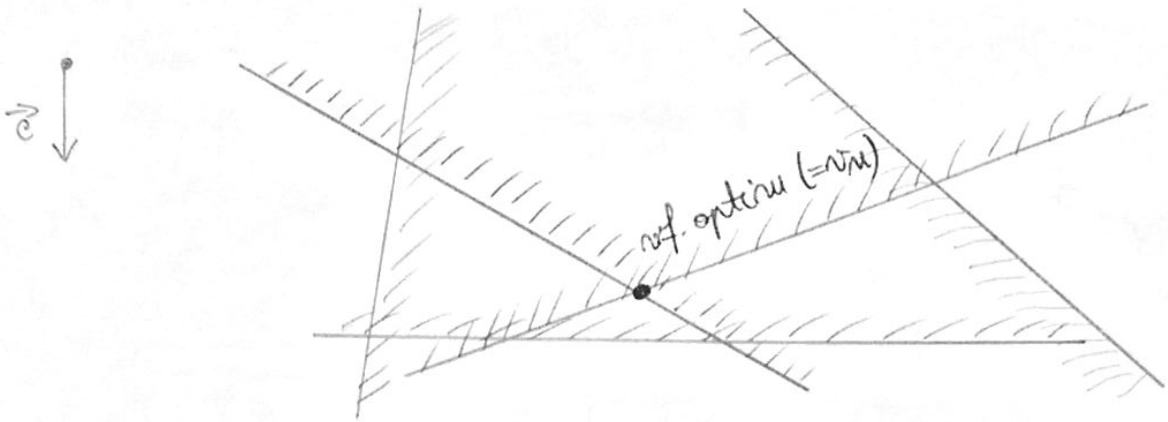
$$\le \sum_{i=1}^{n} O(i) \cdot \frac{2}{i} = O(n)$$

Afirmație: $\mu(X_i) \le \frac{2}{i}$

Justificare: $\mu(X_i)$ este, de fapt, probabilitatea ca $v_{i-1} \notin h_i$.

„Backword analysis": pp. algoritmul terminat; $v_n \to$ vârful optim

• care este probabilitatea ca $v_{n-1} \notin h_n$, adică la adăugarea lui h_n, v_{n-1} să fie modificat în v_n? $\Leftrightarrow$ • care este probabilitatea ca eliminând un semiplan, să fie modificat vârful optim?



- nr. cazuri posibile : n
- nr. cazuri care modifică : 2 (≤ 2)

$\Rightarrow$ probabilitatea ca vârful optim să fie găsit abia la ultimul pas este $\leq \frac{2}{n}$

Raționament analog pentru pasul i $\rightarrow$ afirmația

— PROBLEME de LOCALIZARE —
(POINT LOCATION)

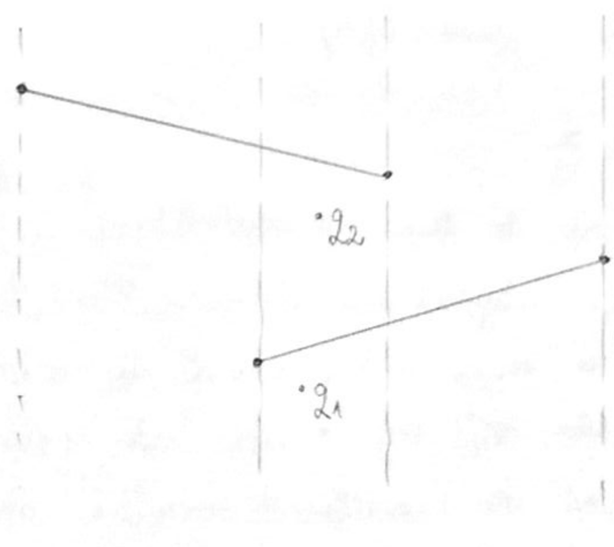
$\Rightarrow$ căutare Google Maps : harta e reținută sub formă de subdiviziuni planare

Subdivizionare a planului în fâșii (benzi) verticale (slab $\Rightarrow$ en)

$\rightarrow$ căutare după abscisă pentru identificarea fâșiei verticale

$\rightarrow$ căutare în cadrul unei fâșii, fiind identificate fața care conține punctul q.

⚠ Pe verticală, căutăm „în raport cu segmentele" (în cadrul unei fâșii)

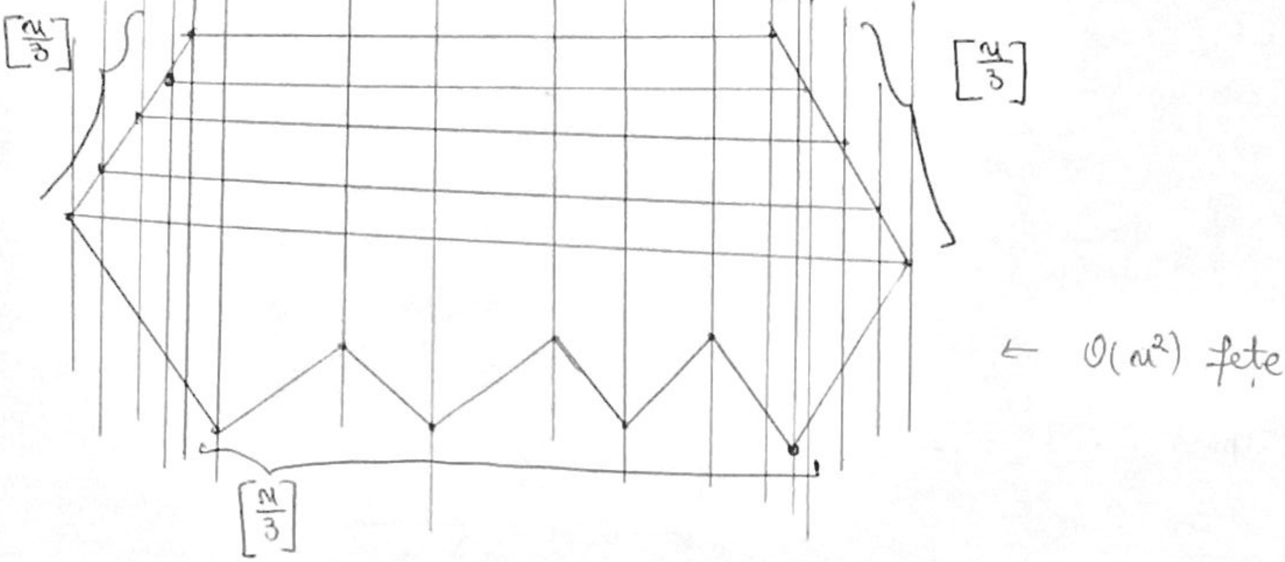


localizare cu testul de orientare

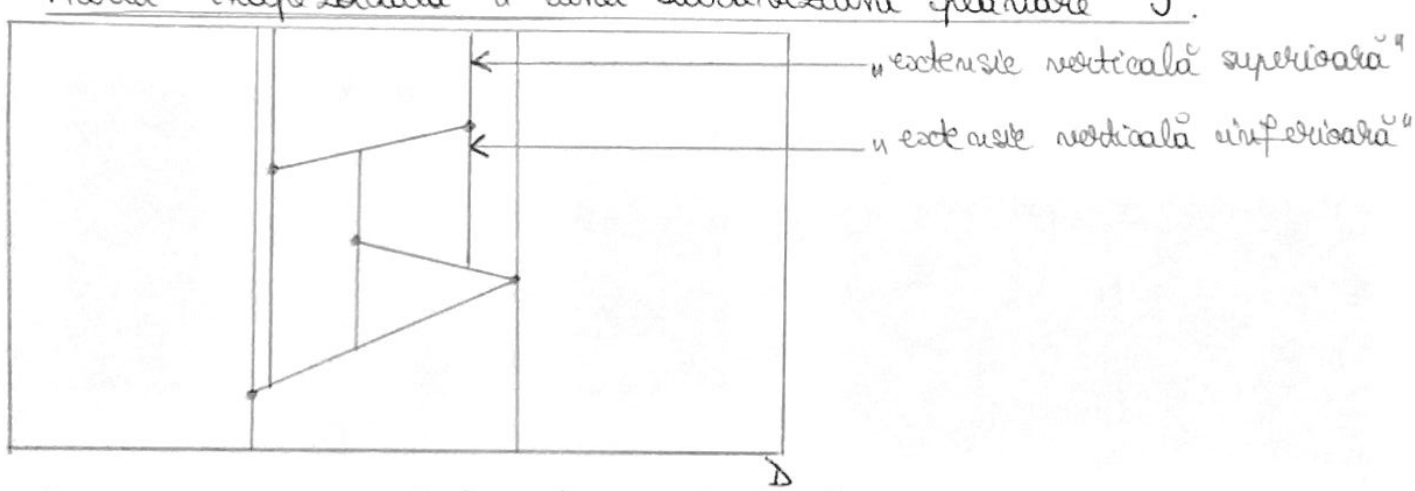
Observații:

i) Timp de căutare $O(\log n)$ (de ce?)

ii) Memorie necesară: uneori $O(n^2)$!



Harta trapezoidală a unei subdiviziuni planare S.



Pentru fiecare vârf sunt considerate două extensii verticale, se opresc când întâlnesc un segment sau D.

$\Rightarrow$ hartă trapezoidală $\left| \begin{array}{l} S \\ D \\ \text{extensii} \end{array} \right.$

• Cum este memorat un trapez?