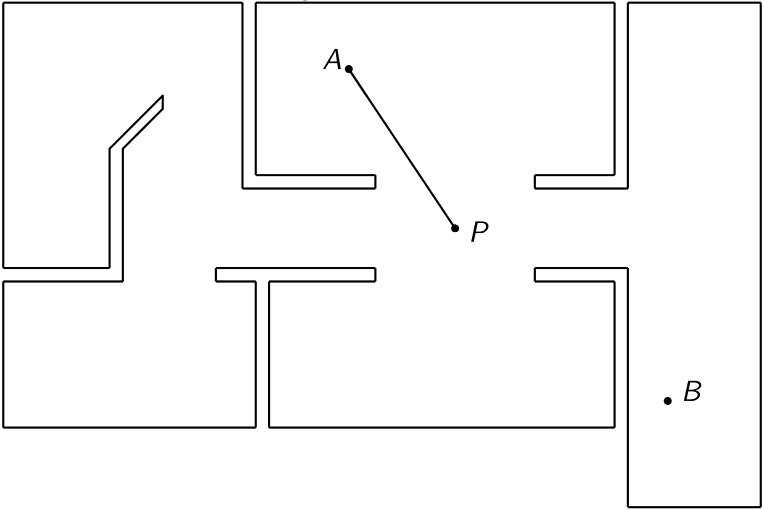
Triangulări

Mihai-Sorin Stupariu

Sem. I, 2019 - 2020

Supravegherea unei galerii de artă

Camera din P poate supraveghea A, dar nu B.



Formalizare

- O galerie de artă poate fi interpretată (în contextul acestei probleme) ca un poligon simplu $\mathcal{P}$ (adică un poligon fără autointersecții) având n vârfuri.
- O cameră video (vizibilitate 360^0) poate fi identificată cu un punct din interiorul lui $\mathcal{P}$; ea poate supraveghea acele puncte cu care poate fi unită printr-un segment inclus în interiorul poligonului.
- **Problema galeriei de artă:** *câte camere video sunt necesare pentru a supraveghea o galerie de artă și unde trebuie amplasate acestea?*

Numărul de camere vs. forma poligonului

- Se dorește exprimarea numărului de camere necesare pentru supraveghere în funcție de n (sau controlarea acestuia de către n).
- Pentru a supraveghea un spațiu având forma unui poligon convex, este suficientă o singură cameră.
- Numărul de camere depinde și de forma poligonului: cu cât forma este mai "complexă", cu atât numărul de camere va fi mai mare.
- **Principiu:** Poligonul considerat: descompus în triunghiuri (triangulare).

Definiții

- Fie $\mathcal{P}$ un poligon plan.
- (i) O **diagonală** a lui $\mathcal{P}$ este un segment ce unește două vârfuri ale acestuia și care este situat în interiorul lui $\mathcal{P}$.
- (ii) O **triangulare** $\mathcal{T}_P$ a lui $\mathcal{P}$ este o descompunere a lui $\mathcal{P}$ în triunghiuri, dată de o mulțime maximală de diagonale ce nu se intersectează.
- **Teoremă.** *Orice poligon simplu admite o triangulare. Orice triangulare a unui poligon cu n vârfuri conține exact $n-2$ triunghiuri.*

Rezovlarea problemei galeriei de artă

- Amplasarea camerelor se poate face în vârfurile poligonului.
- Dată o pereche $(\mathcal{P}, \mathcal{T}_P)$ se consideră o 3-colorare a acesteia: fiecărui vârf îi corespunde o culoare dintr-un set de 3 culori și pentru fiecare triunghi, cele 3 vârfuri au culori distincte.
- **Observație.** Dacă $\mathcal{P}$ este simplu, o astfel de colorare există, deoarece graful asociat perechii $(\mathcal{P}, \mathcal{T}_P)$ este arbore.

Teorema galeriei de artă

- **Teoremă.** [Chvátal, 1975; Fisk, 1978] *Pentru un poligon cu n vârfuri,* $\left[\frac{n}{3}\right]$ *camere sunt* **uneori necesare** *și întotdeauna* **suficiente** *pentru ca fiecare punct al poligonului să fie vizibil din cel puțin una din camere.*
- Despre Teorema Galeriei de Artă: J. O'Rourke, *Art Gallery Theorems and Algorithms*

Metode de triangulare: ear cutting / clipping / trimming

- Concepte:
  - **vârf principal**,
  - **ear (vârf / componentă de tip E)** [Meisters, 1975];
  - **mouth (vârf / componentă de tip M)** [Toussaint, 1991].
- Orice vârf de tip E este convex; orice vârf de tip M este concav (reflex). Reciproc nu neapărat!
- **Teoremă.** (Two Ears Theorem [Meisters, 1975]) *Orice poligon cu cel puțin* 4 *vârfuri admite cel puțin două componente de tip E care nu se suprapun.*
- **Corolar.** *Orice poligon simplu admite (cel puțin) două diagonale.*
- Găsirea unei componente de tip E: complexitate $O(n)$ [ElGindy, Everett, Toussaint, 1993]. Se bazează pe Two Ears Theorem!
- Algoritmul de triangulare bazat de metoda *ear cutting*: complexitate $O(n^2)$.
- Link despre triangulări
  Link pentru algoritmul *Ear cutting*

Metode de triangulare: descompunerea în poligoane monotone

- Concept: **poligon y-monoton**
- Algoritmi de triangulare eficienți: complexitate $O(n)$ pentru poligoane y-monotone [Garey et al., 1978].
- Descompunerea unui poligon oarecare in componente y-monotone poate fi realizată cu un algoritm de complexitate $O(n \log n)$ [Lee, Preparata, 1977].
- Există și alte clase de algoritmi mai rapizi; [Chazelle, 1990]: algoritm liniar.
- Link pentru alte abordări
- Găsirea unui algoritm liniar "simplu" Problemă în *The Open Problems Project*

Triangularea poligoanelor monotone

Input: Un poligon y-monoton $\mathcal{P}$.
Output: O triangulare a lui $\mathcal{P}$.

1. Lanțul vârfurilor din partea stângă și al celor din partea dreaptă sunt unite într-un singur șir, ordonat descrescător, dupa y (dacă ordonata este egală, se folosește abscisa). Fie $v_1, v_2, \ldots, v_n$ șirul ordonat.
2. Inițializează o stivă vidă $\mathcal{S}$ și inserează v_1, v_2.
3. **for** $j = 3$ **to** $n - 1$
4.   **do** **if** v_j și vârful din top al lui $\mathcal{S}$ sunt în lanțuri diferite
5.   **then** extrage toate vârfurile din $\mathcal{S}$
6.   inserează diagonale de la v_j la vf. extrase, exceptând ultimul
7.   inserează v_{j-1} și v_j în $\mathcal{S}$
8.   **else** extrage un vârf din $\mathcal{S}$
9.   extrage celelalte vârfuri din $\mathcal{S}$ dacă diagonalele formate cu v_j sunt în interiorul lui $\mathcal{P}$; inserează aceste diagonale; inserează înapoi ultimul vârf extras
10.   inserează v_j în $\mathcal{S}$
11. adaugă diagonale de la v_n la vf. stivei (exceptând primul și ultimul)