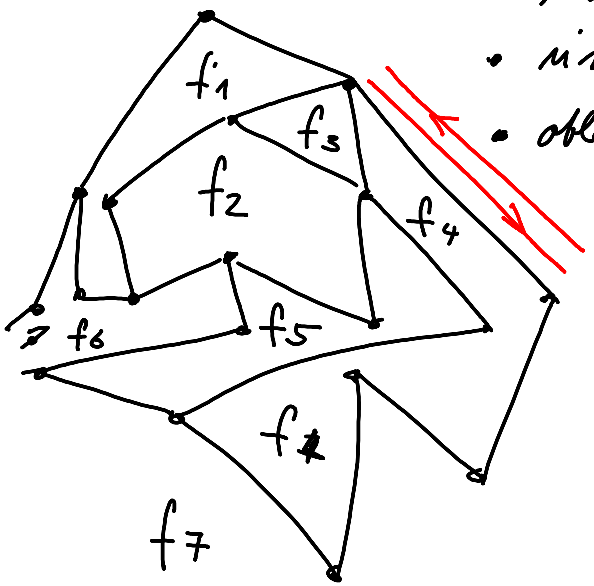
Průniky map

Mapa ~ rovinné podrozdělení z

- bodů
- úseček
- oblastí



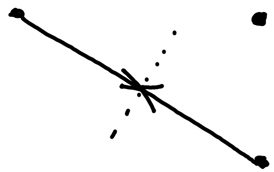
Doubly connected edge list

Dvojitě navázaný seznam

se skládá z

- vrcholů (vertex)
- hran (edge) → orientované
- oblastí (face)

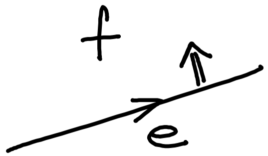
1 úsečka ~ 2 hrany dvojčata (twins)



2

Výhoda orientace: k dané hraně můžeme přiřadit levo přilehlou oblast

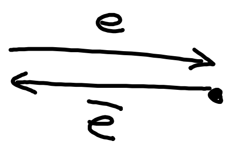
Ta je vlevo od hrany.



DCEL jsou tři tabulky

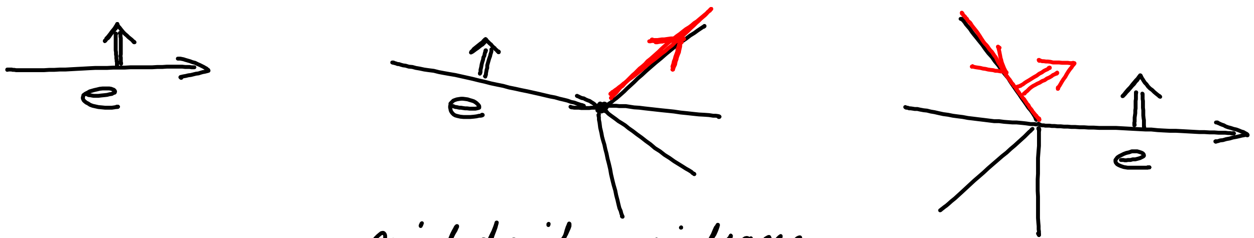
(1) pro vrcholy   jméno, souřadnice, $\underbrace{\text{1 vycházející hrana}}_{\text{ukazatel (pointer) do tabulky hran}}$

(2) pro hrany   jméno, $\underbrace{\text{vrchol, ze kterého vychází}}_{\text{pointer}}$, $\underbrace{\text{dvojče}}_{\text{pointer}}$



(3)

$\underbrace{\text{přilehlá oblast}}_{\text{pointer}}$, $\underbrace{\text{následující hrana}}_{\text{pointer}}$, $\underbrace{\text{předchozí hrana}}_{\text{pointer}}$



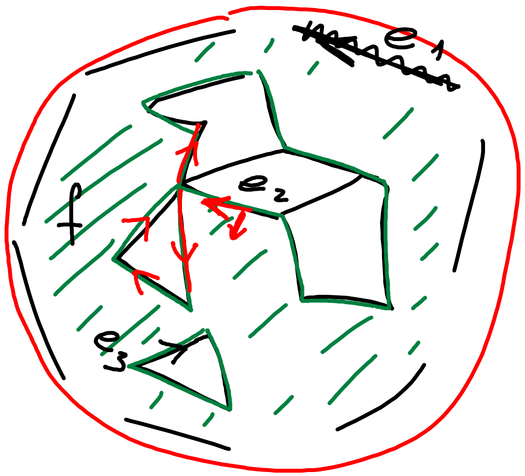
následník je hrana
vycházející z koncového
vrcholu hrany e a mající
stejnou přilehlou oblast

předchůdce hrany
e definován
analogicky

4

(3) Tabulka pro oblasti

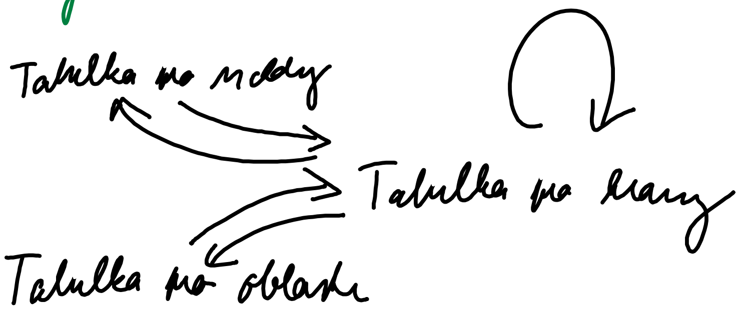
jméno, jedna hrana z vnější hranice, jedna hrana z každé komponenty vnitřní hranice
pointer



vnější hranice

2 komponenty vnitřní hranice

$f \mid e_1 \mid e_2, e_3$



5

Úkoly

- najít všechny hrany e vnější hranice dané oblasti f

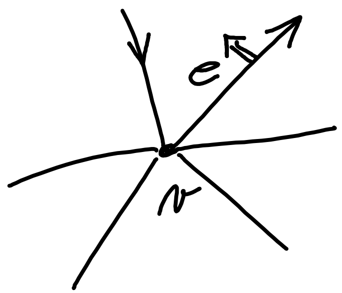
  $f \longrightarrow$ 1 hrana vnější hranice $\longrightarrow$ následník $\longrightarrow$ následníka

  atd až se dostaneme do e.

  Čas potřebný k nalezení je úměrný počtu hran této hranice
- najít všechny hrany všech vnitřních hranic
- najít všechny hrany vycházející z daného vrcholu v

  a napsat je podle směru hod. ručiček

(6)



$v \longrightarrow e \longrightarrow$ dvojče k $\underbrace{\text{předchozí}}_{e}$ e

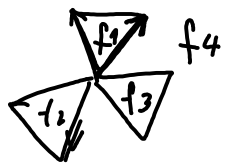
a děláme tak dlouho až se dostaneme do počátečního e

Potřebný čas je úměrný počtu ~~...~~ vycházejících hran

- vypsat všechny oblasti na jejichž hranici leží uzel v podle směru hod ručiček

hranice f_1

vnitřní hranice f_2



$\boxed{f_1\ f_4\ f_2\ f_4\ f_3\ f_4}$

7

Překryv map

Dvě mapy popsané dvojitě souvislými seznamy

$\mathcal{S}_1$ jedno podrozdělení

$\mathcal{S}_2$ druhé podrozdělení

$\mathcal{O}(\mathcal{S}_1, \mathcal{S}_2)$ překryv podrozdělení

$\mathcal{D}(\mathcal{S}_i)$ dvojitě souvislé seznamy pro $\mathcal{S}_1$ a $\mathcal{S}_2$

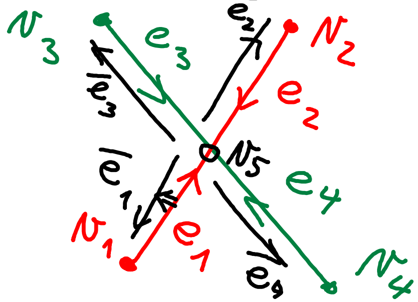
Chceme vytvořit dvojitě souvislý seznam $\mathcal{D}$ pro $\mathcal{O}(\mathcal{S}_1, \mathcal{S}_2)$.

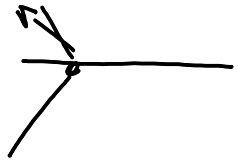
8

Algoritmus pro průnik polygonů začne tím, že tabulky vrcholů a hran pro S_1 a S_2 dáme „dohromady". V takto vzniklých tabulkách posadíme změny vždy, když zametací přímka narazí na průsečík hran z S_1 a S_2

Změny budeme demonstrovat na příkladech

Nový průsečík leží uvnitř modré i červené hrany





9

Nový vrchol v_5, jedna hrana vyhovující $\overline{e_1}$

Nové hrany $\overline{e}_1, \overline{e}_2, \overline{e}_3, \overline{e_4}$ s poč. vrcholem v_5

Měníme řády pro e_1, e_2, e_3, e_4

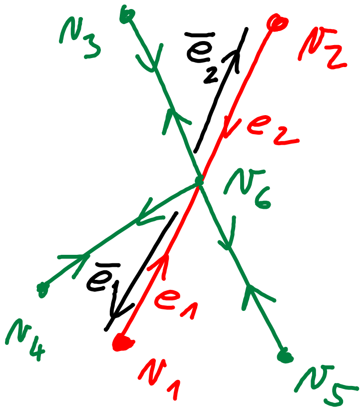
Vytvoříme nové řádky pro $\overline{e}_1, \overline{e}_2, \overline{e}_3, \overline{e_4}$

e_1 poč. vrchol zůstane, dvojče se změní na $\overline{e}_1$, předchůdce zůstane, následník $\overline{e_3}$, ukazatel na oblast nevyplňujeme

$\overline{e_1}$ poč vrchol je v_5, ~~dvojče~~ e_1, předchůdce nové e_4, následník jako u e_2 ukazatel na oblast nevyplňujeme

10

Situace, kdy ~~nový prvek~~ průsečík je měnění pouze pro červenou hranu



Nové začátky pouze pro nové hrany $\bar{e}_1$ a $\bar{e}_2$

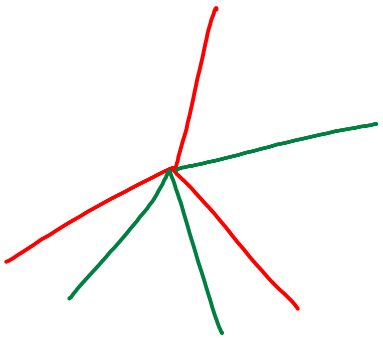
Analogicky

Změny jsou ale i v řádcích pro všechny hrany $e_1, e_2, e_3, e_4, \ldots e_8$

Použijeme analogicky jako v předchozím případě

11

Průsečík neleží uvnitř žádné hrany



Žádné nové uzly ani hrany. Nicméně je nutno modifikovat následníky a předchůdce u hran jdoucích do průsečíku

Tímto postupem dostaneme dvojitě navázaný seznam pro uzly a pro hrany (s výjimkou odkazů na oblasti)

Zbývá najít tabulku pro oblasti.

12

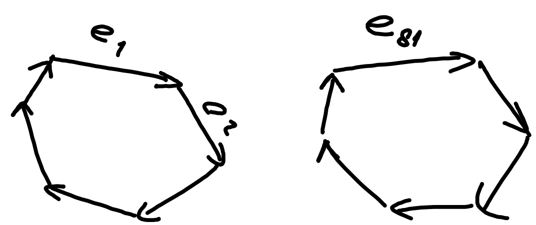
Každá oblast je určena jednou vnější hranicí a několika vnitřními.
Tyto hranice můžeme z našich tabulek již určit

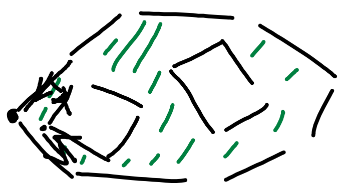
Cyklus .. posloupnost hran $e_1, e_2, \ldots e_k$ takových, že

(1) e_{i+1} následník e_i $\quad i = 1, 2, \ldots k-1$

(2) e_1 je následník e_k

Tím dostaneme množinu hran na uspořádané podmnožiny → cykly

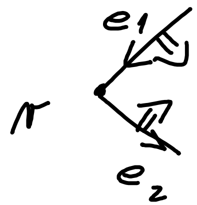




Najdeme vnější a vnitřní cykly

13

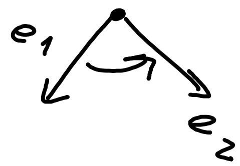
Vrchol v nejvíce vlevo v cyklu



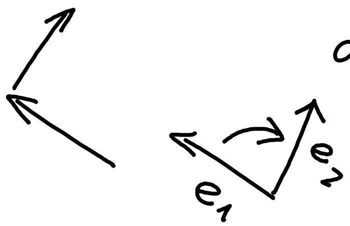
cyklus je vnější právě když zatáčka vlevo $< 180°$

cyklus je vnitřní právě když zatáčka vlevo $> 180°$

(zatáčka vpravo $< 180°$)



$$\det\begin{pmatrix} e_{1x} & e_{1y} \\ e_{2x} & e_{2y} \end{pmatrix} > 0$$ vnější cyklus

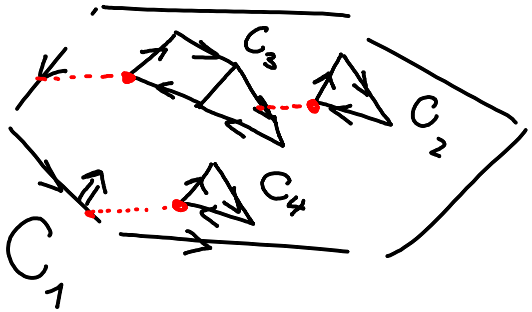


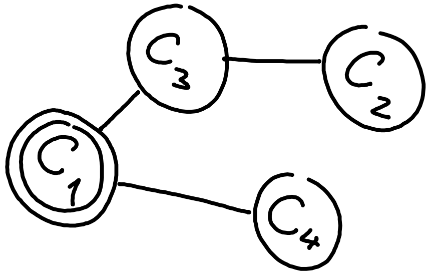
$$\det\begin{pmatrix} e_{1x} & e_{1y} \\ e_{2x} & e_{2y} \end{pmatrix} < 0$$ vnitřní cyklus

14

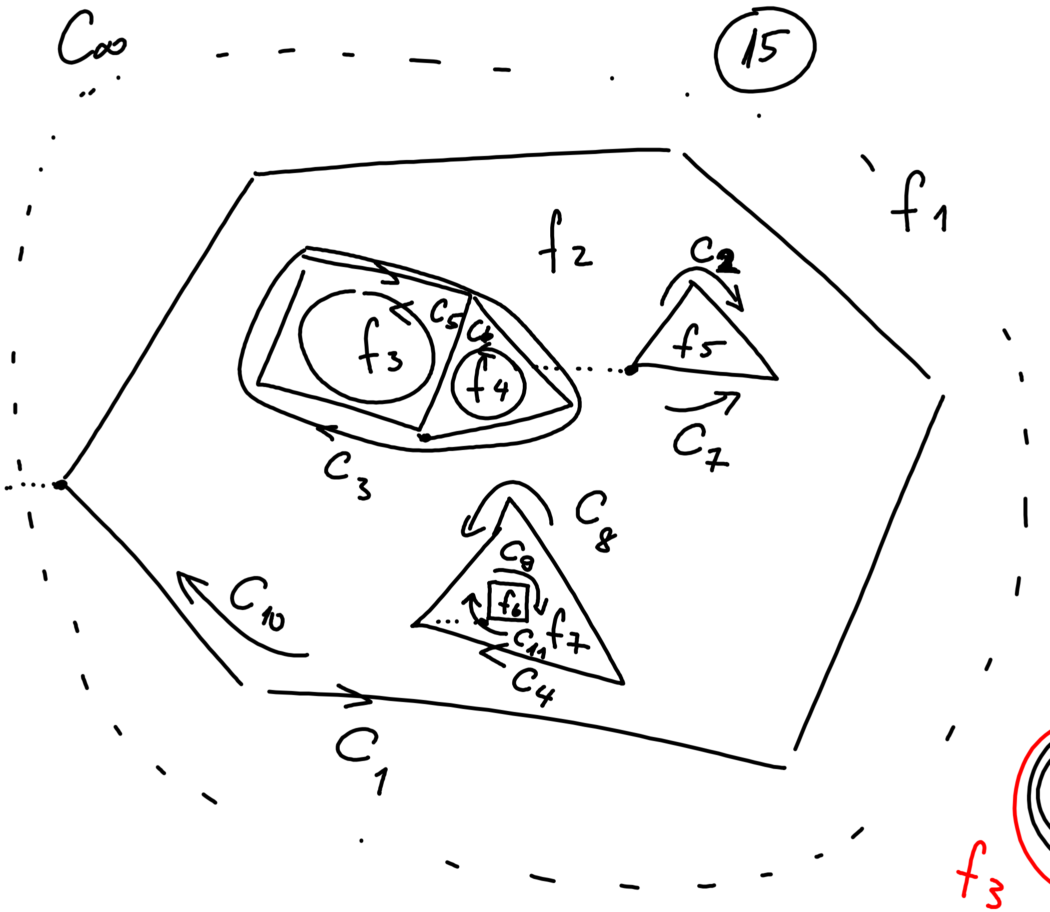
Vytvoříme abstraktní graf G, jehož uzly jsou cykly námi nalezené a hrany vytvoříme takto.

Hranou je spojen minimální a minimální nebo menší cyklus právě tehdy, když úsečka vycházející směrem vlevo z nejlevějšího bodu minimálního cyklu narazí přímo na onen druhý cyklus

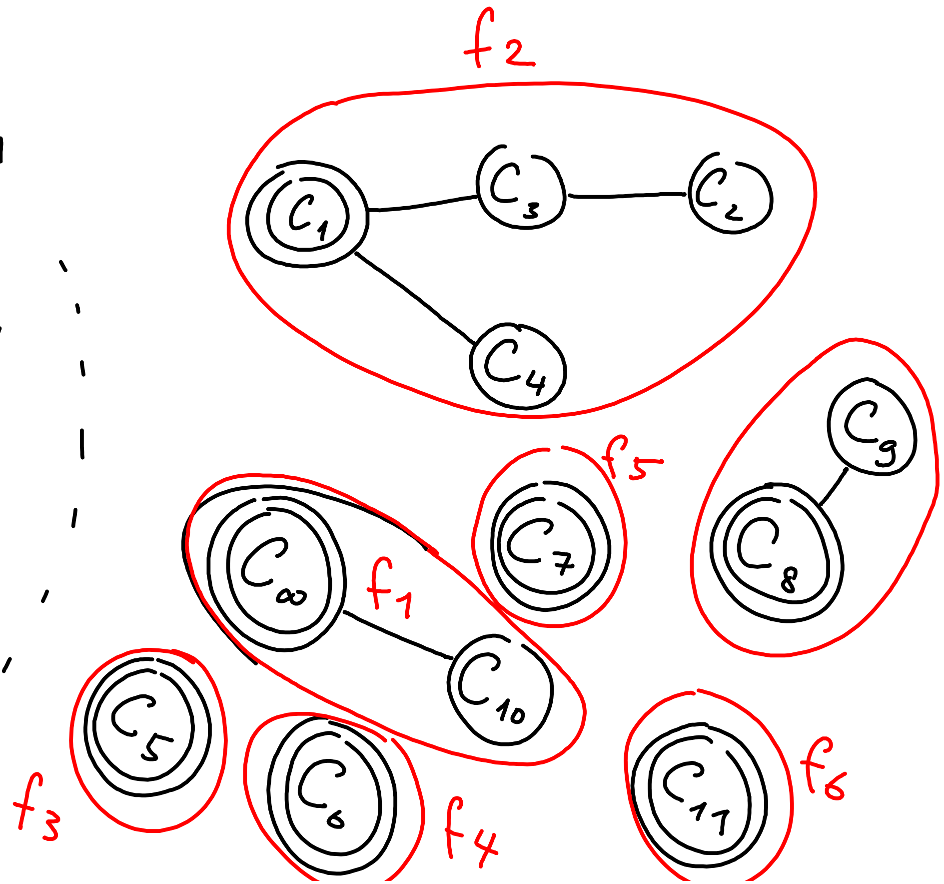




Tato komponenta souvislosti určuje jednu oblast.



C_∞
15
f_1
f_2
C_2
C_5
C_6
f_3
f_4
f_5
C_7
C_3
C_8
C_9
f_6
C_{11}
f_7
C_4
C_{10}
C_1

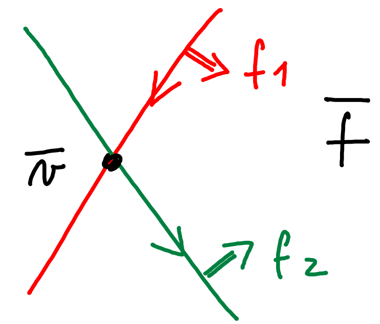


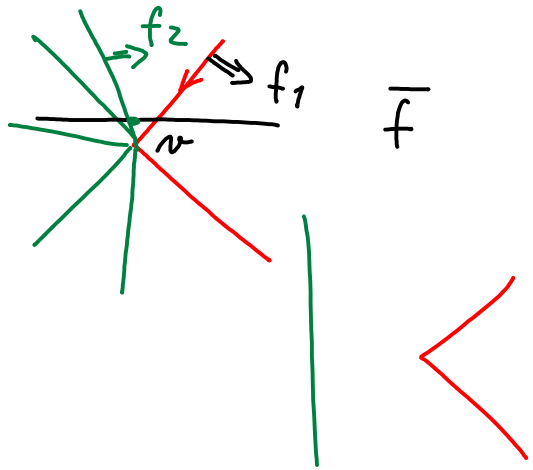
f_2
C_1
C_3
C_2
C_4
f_5
C_9
C_∞
f_1
C_7
C_8
C_{10}
C_5
C_6
C_{11}
f_6
f_3
f_4

16

Graf G se rozpadne na komponenty souvislosti, v každé je právě jeden vnější cyklus (jeden je C_∞). Každá komponenta souvislosti odpovídá jedné oblasti překryvu map. To nám umožní vyplnit tabulku pro oblasti a doplnit tabulku pro hrany. (ukazatel na přilehlou oblast)

Tím máme dvojitě souvislý seznam pro překryv, ale chceme víc, aby u každé nové oblasti byla jména původních oblastí





17

$\overline{f}$ . . . . f_1 f_2

$\overline{f}$ - - - - - f_1 f_2

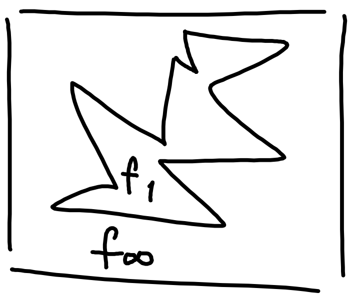
Tema algoritmul mai în caz nerăcinat

$O((n+k)\log n)$

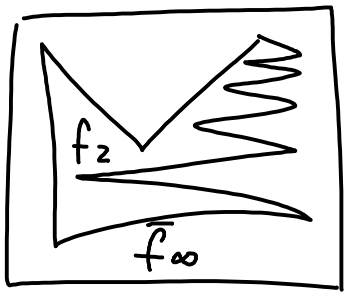
unde n = nr. vârfuri înainte în φ_1 și φ_2
și k = nr. vârfuri intersecție

18

Tento algoritmus umožňuje počítat průniky a sjednocení nekonečných mnohoúhelníků



φ_1



φ_2

Průnik jsou ty oblasti s příznakem, které leží v f_1 a f_2

Sjednocení, ty oblasti, které leží v f_1 nebo f_2