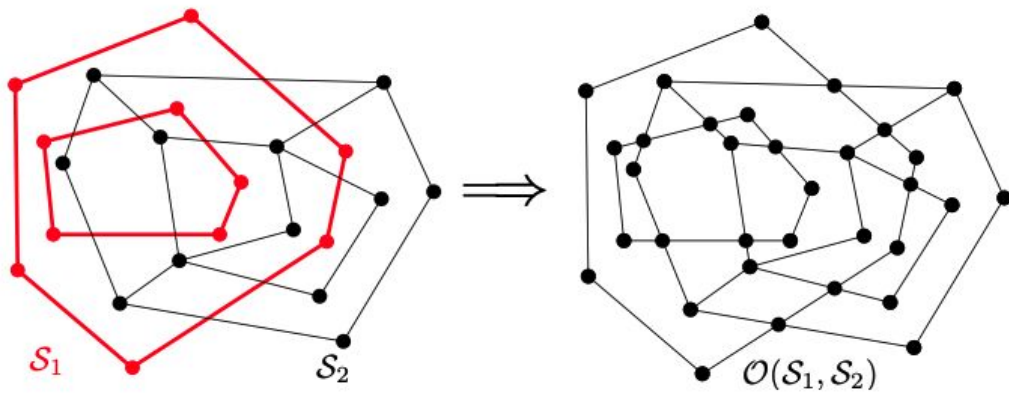


Przecinanie się podziałów na płaszczyźnie

Tomasz Zachwieja & Filip Twardy

Problem

Tematem tego projektu jest wyznaczanie przecięcia dwóch podziałów płaszczyzny i wyznaczenie podziału z tego wynikającego. Jest to rozwinięcie problemu przecinania się odcinków, ponieważ podziały mogą być dużo bardziej złożonymi figurami.



Reprezentacja podziału

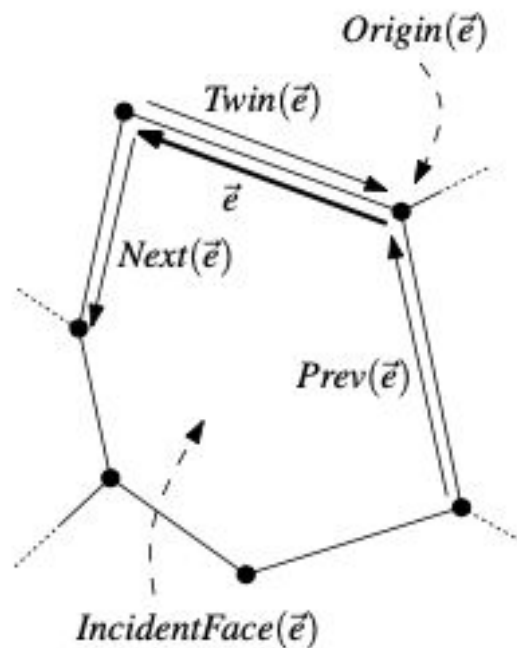
Do reprezentacji podziałów wykorzystana jest dwukierunkowa lista krawędzi (*Doubly Connected Edge List*, DCEL) umożliwiająca efektywne przechodzenie pomiędzy kolejnymi elementami podziału, tj:

- Wierzchołkami
- Krawędziami
- Ścianami

Każdy z tych elementów jest zawarty w DCEL i zawiera odpowiednie atrybuty i wskaźniki.

Krawędź (półkrawędź) e

- $\text{Origin}(e)$ - wierzchołek z którego wychodzi
- $\text{Twin}(e)$ - bliźniacza półkrawędź
- $\text{IncidentFace}(e)$ - ściana którą ogranicza
- $\text{Next}(e)$ i $\text{Prev}(e)$ - kolejna i poprzednia półkrawędź na obwodce ściany tej półkrawędzi

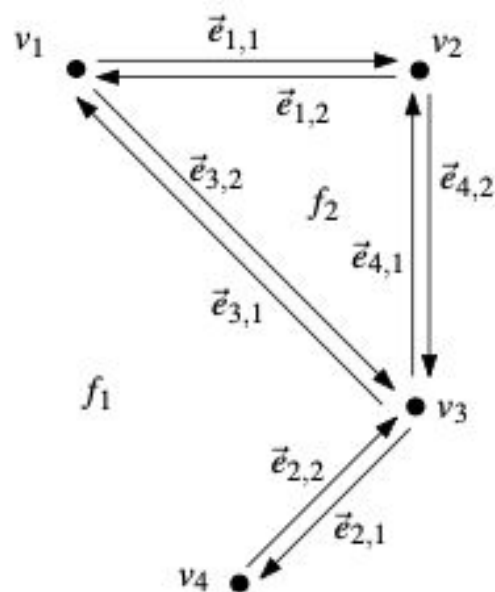


Wierzchołek v

- $\text{Coordinates}(v)$ - współrzędne
- $\text{IncidentEdge}(v)$ - półkrawędzie incydentne do v

Ściana f

- $\text{OuterComponent}(f)$ - półkrawędź ograniczająca f - dowolna
- $\text{InnerComponent}(f)$ - półkrawędzie elementów wewnątrz f



Vertex	Coordinates	IncidentEdge
v_1	(0,4)	$\vec{e}_{1,1}$
v_2	(2,4)	$\vec{e}_{4,2}$
v_3	(2,2)	$\vec{e}_{2,1}$
v_4	(1,1)	$\vec{e}_{2,2}$

Face	OuterComponent	InnerComponents
f_1	nil	$\vec{e}_{1,1}$
f_2	$\vec{e}_{4,1}$	nil

Half-edge	Origin	Twin	IncidentFace	Next	Prev
$\vec{e}_{1,1}$	v_1	$\vec{e}_{1,2}$	f_1	$\vec{e}_{4,2}$	$\vec{e}_{3,1}$
$\vec{e}_{1,2}$	v_2	$\vec{e}_{1,1}$	f_2	$\vec{e}_{3,2}$	$\vec{e}_{4,1}$
$\vec{e}_{2,1}$	v_3	$\vec{e}_{2,2}$	f_1	$\vec{e}_{2,2}$	$\vec{e}_{4,2}$
$\vec{e}_{2,2}$	v_4	$\vec{e}_{2,1}$	f_1	$\vec{e}_{3,1}$	$\vec{e}_{2,1}$
$\vec{e}_{3,1}$	v_3	$\vec{e}_{3,2}$	f_1	$\vec{e}_{1,1}$	$\vec{e}_{2,2}$
$\vec{e}_{3,2}$	v_1	$\vec{e}_{3,1}$	f_2	$\vec{e}_{4,1}$	$\vec{e}_{1,2}$
$\vec{e}_{4,1}$	v_3	$\vec{e}_{4,2}$	f_2	$\vec{e}_{1,2}$	$\vec{e}_{3,2}$
$\vec{e}_{4,2}$	v_2	$\vec{e}_{4,1}$	f_1	$\vec{e}_{2,1}$	$\vec{e}_{1,1}$

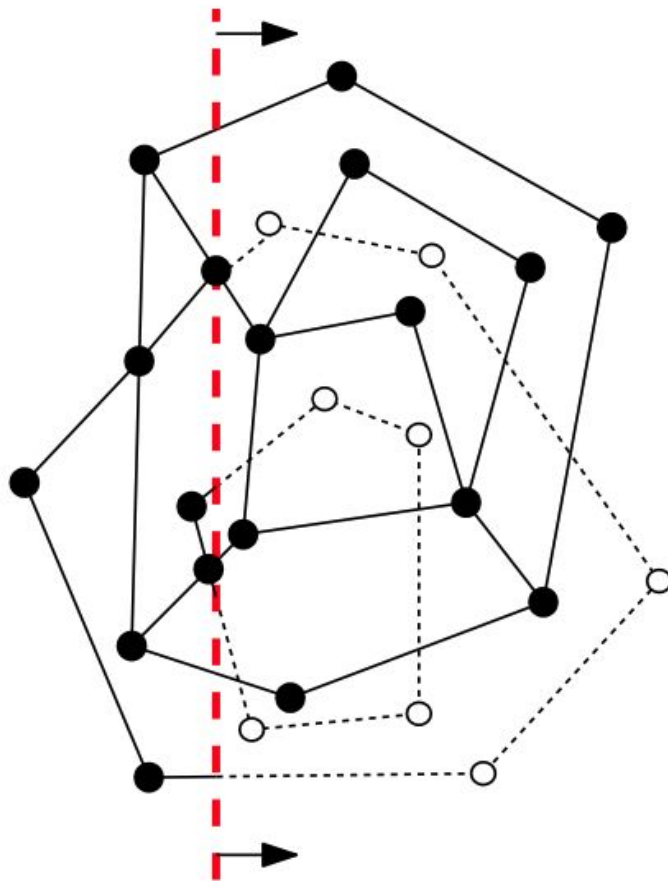
Algorytm

Program przyjmuje dwa podziały w postaci DCEL: $S1, S2$ i wykonuje następujące kroki:

1. Skopiuj $S1, S2$ do nowej DCEL
2. Wyznacz wszystkie punkty przecięć krawędzi i stwórz nowe wierzchołki i krawędzie
3. Zaktualizuj DCEL o powstałe elementy

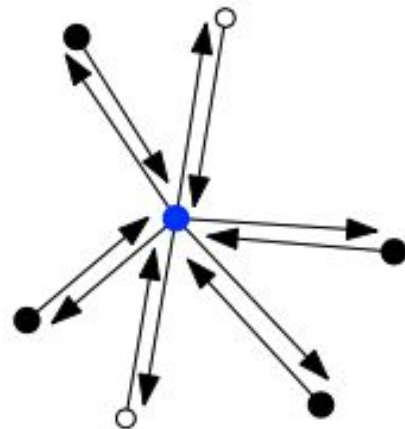
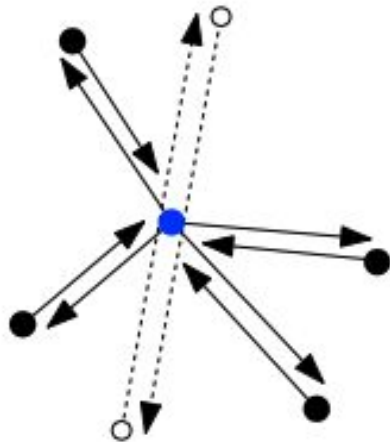
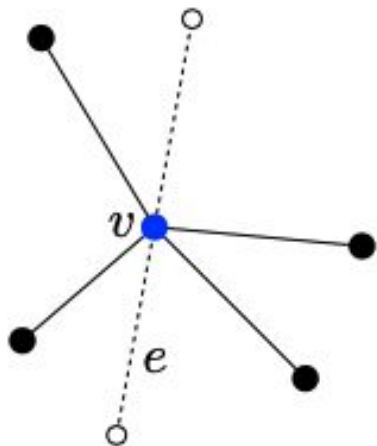
Przecinanie się krawędzi

Znajdowanie i wyznaczanie przecięć krawędzi jest wykonywane przy pomocy algorytmu zmiatania



Aktualizowanie krawędzi i wierzchołków

W zależności od przypadku (przecięcie dwóch krawędzi lub krawędź przechodząca przez wierzchołek) tworzone są odpowiednie półkrawędzie i wierzchołki



Bibliografia

- Computational Geometry Algorithms and Applications, 3rd Ed - Mark de Berg
- homepages.math.uic.edu/~jan/mcs481/suboverlays.pdf