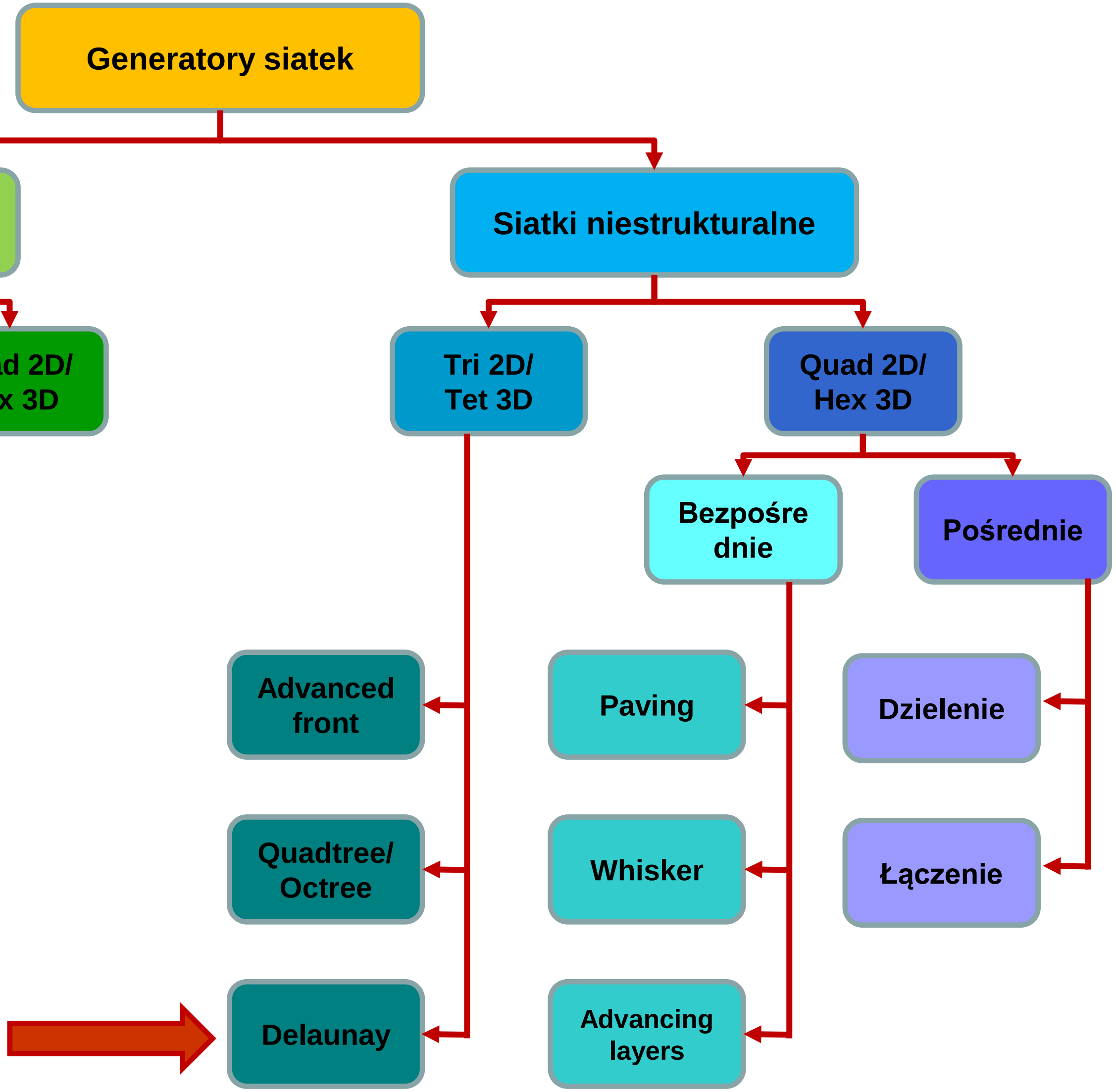
Geometria Obliczeniowa

Dr hab. inż. **Łukasz Madej**, prof. AGH
Mgr inż. **Daniel Bachniak**
Katedra Informatyki Stosowanej i Modelowania
Wydział Inżynierii Metali i Informatyki Przemysłowej

Budynek B5
p. 716
lmadej@agh.edu.pl
home.agh.edu.pl/lmadej



Generatory siatek
Siatki strukturalne
Siatki niestrukturalne
Tri 2D/ Tet 3D
Quad 2D/ Hex 3D
Tri 2D/ Tet 3D
Quad 2D/ Hex 3D
Bezpośre dnie
Pośrednie
Advanced front
Quadtree/ Octree
Delaunay
Paving
Whisker
Advancing layers
Dzielenie
Łączenie

Triangulacja to podział figury geometrycznej na sympleksy (trójkąty lub czworościany) w taki sposób, że część wspólna dowolnych dwu różnych sympleksów jest ich wspólną ścianą, wspólnym wierzchołkiem, wspólnym bokiem lub wspólnym trójkątem.

Sympleks – *n* wymiarowym sympleksem jest *n*-wymiarowy wielościan, który jest wypukłą otoczką swoich *n*+1 wierzchołków.

Triangulacja to technika stosowana w grafice komputerowej polegająca na rozbiciu bardziej złożonych obiektów na trójkąty. Rozkładowi takiemu poddane mogą być nawet figury o łukowych krawędziach, jak np. koło czy elipsa.

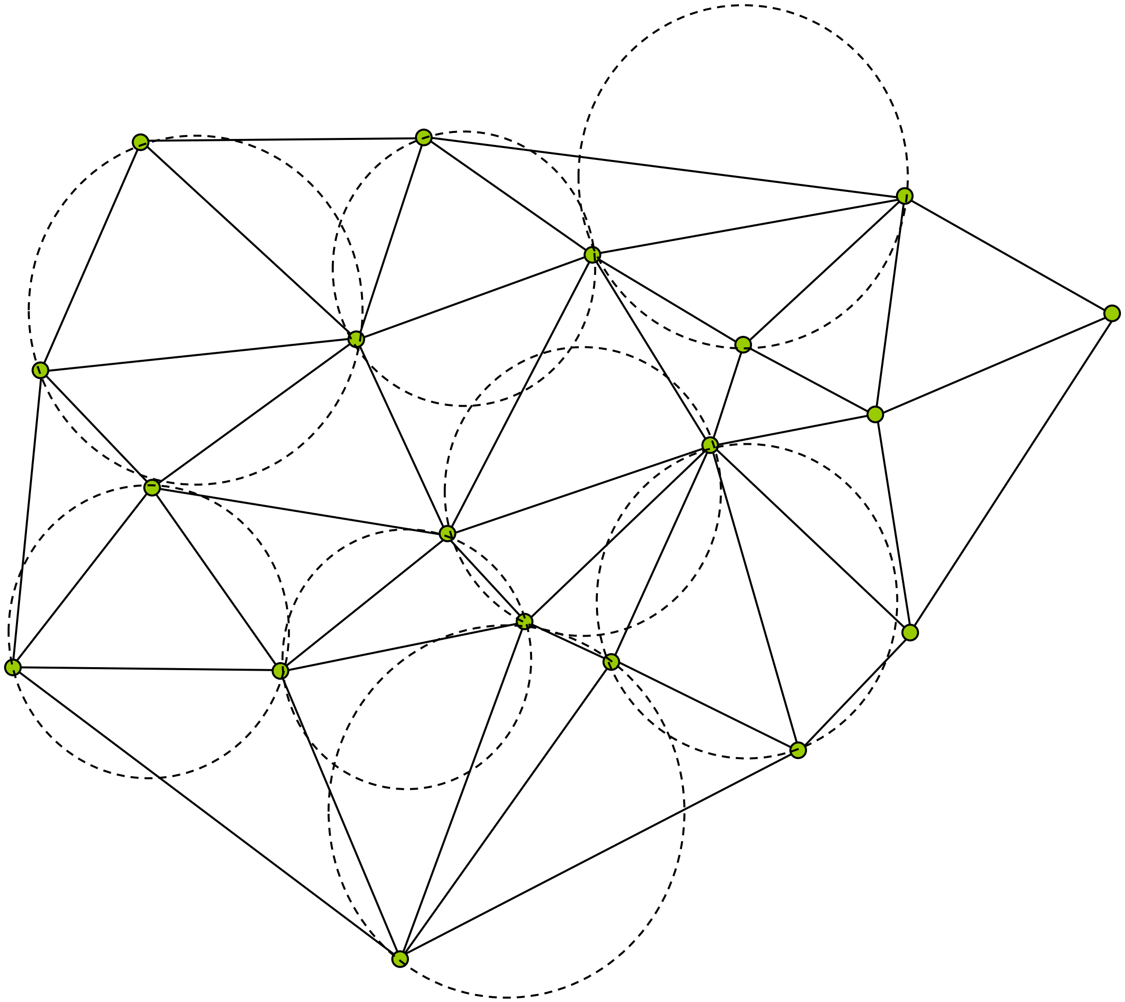
Triangulacja Delaunay (Delone) – triangulacja *T* przestrzeni *R* zdefiniowana następująco:
T to podział *R* na (n+1) - sympleksy, takie że:
- każde dwa sympleksy z T mają wspólną ścianę lub nie mają części wspólnej wcale,
- każdy ograniczony zbiór w *R* ma część wspólną jedynie ze skończenie wieloma sympleksami z *T*,
- wnętrze kuli opisanej na dowolnym sympleksie z *T* nie zawiera wierzchołków żadnego sympleksu z *T.*

Boris Delaunay

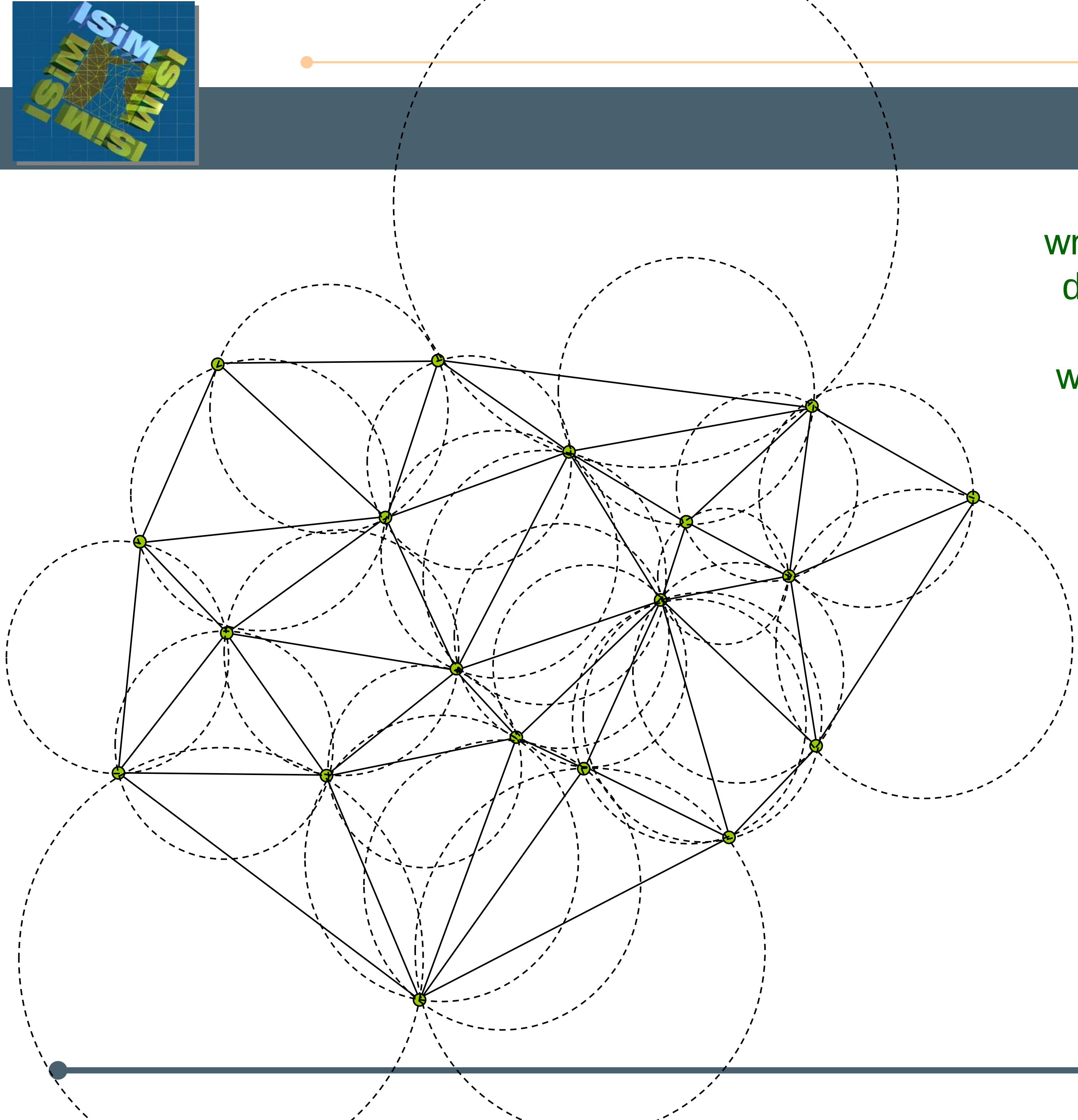
15 marca 1890

– rosyjski matematyk
– alpinista

Mark de Berg, Mirosław Kowaluk: Geometria obliczeniowa. Algorytmy i zastosowania.
Warszawa: Wydawnictwa Naukowo-Techniczne, 2007, s. 221-250. ISBN 978-83-204-3244-2.

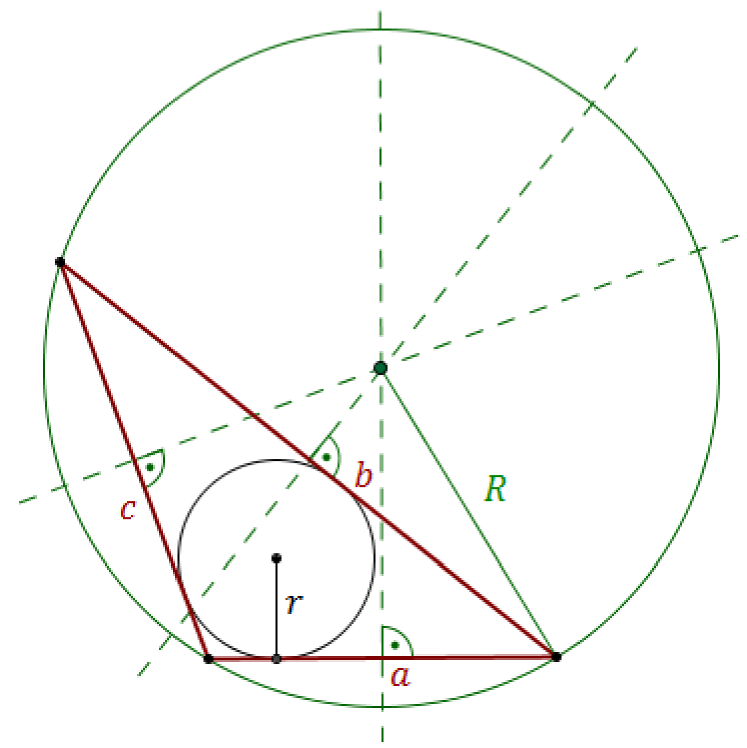


wnętrze kuli opisanej na dowolnym sympleksie z *T* nie zawiera wierzchołków żadnego sympleksu z *T.*



wnętrze kuli opisanej na dowolnym sympleksie z *T* nie zawiera wierzchołków żadnego sympleksu z *T.*

Środek okręgu opisanego leży na przecięciu **symetralnych** boków trójkąta



Promień okręgu opisanego można obliczyć ze wzoru:

$$R = \frac{abc}{4rp}$$

a, b, c - to długości boków trójkąta
r - to długość promienia okręgu wpisanego w trójkąt
p - to połowa obwodu trójkąta

$$R = \frac{abc}{4P}$$

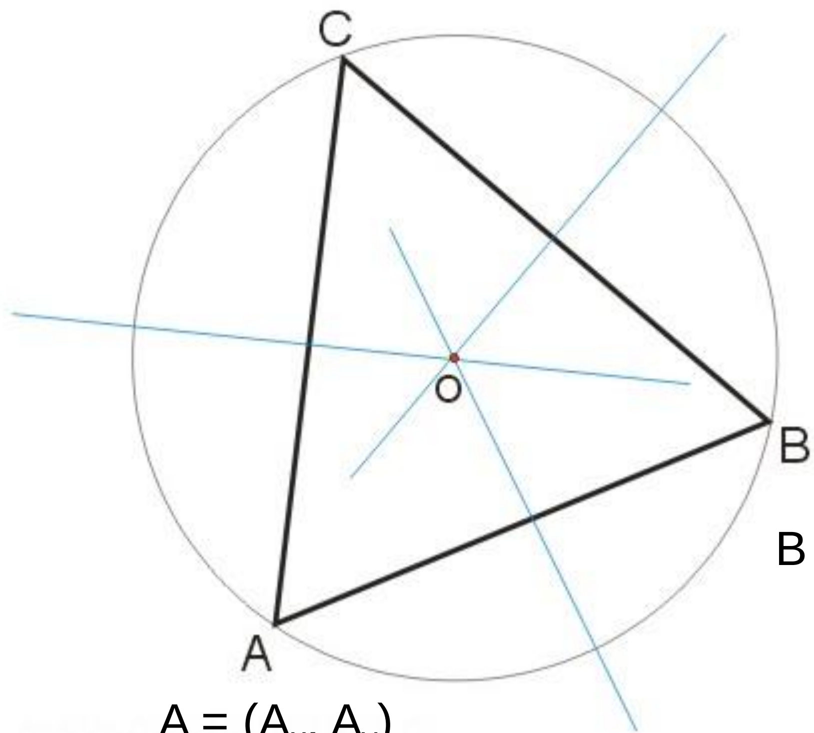
P - pole trójkąta

$$P = \sqrt{p(p-a)(p-b)(p-c)}$$

Symetralna – prosta prostopadła do boku trójkąta, przecinająca jego środek.

$A = (A_x, A_y)$

$B = (B_x, B_y)$



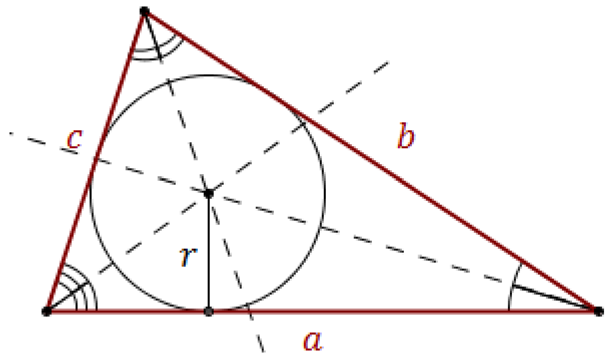
$B = (B_x, B_y)$

$A = (A_x, A_y)$

Równanie symetralnej

$$(2x - A_x - B_x)(A_x - B_x) + \left(2y - A_y - B_y\right)\left(A_y - B_y\right) = 0$$

Środek okręgu wpisanego leży na przecięciu dwusiecznych trójkąta

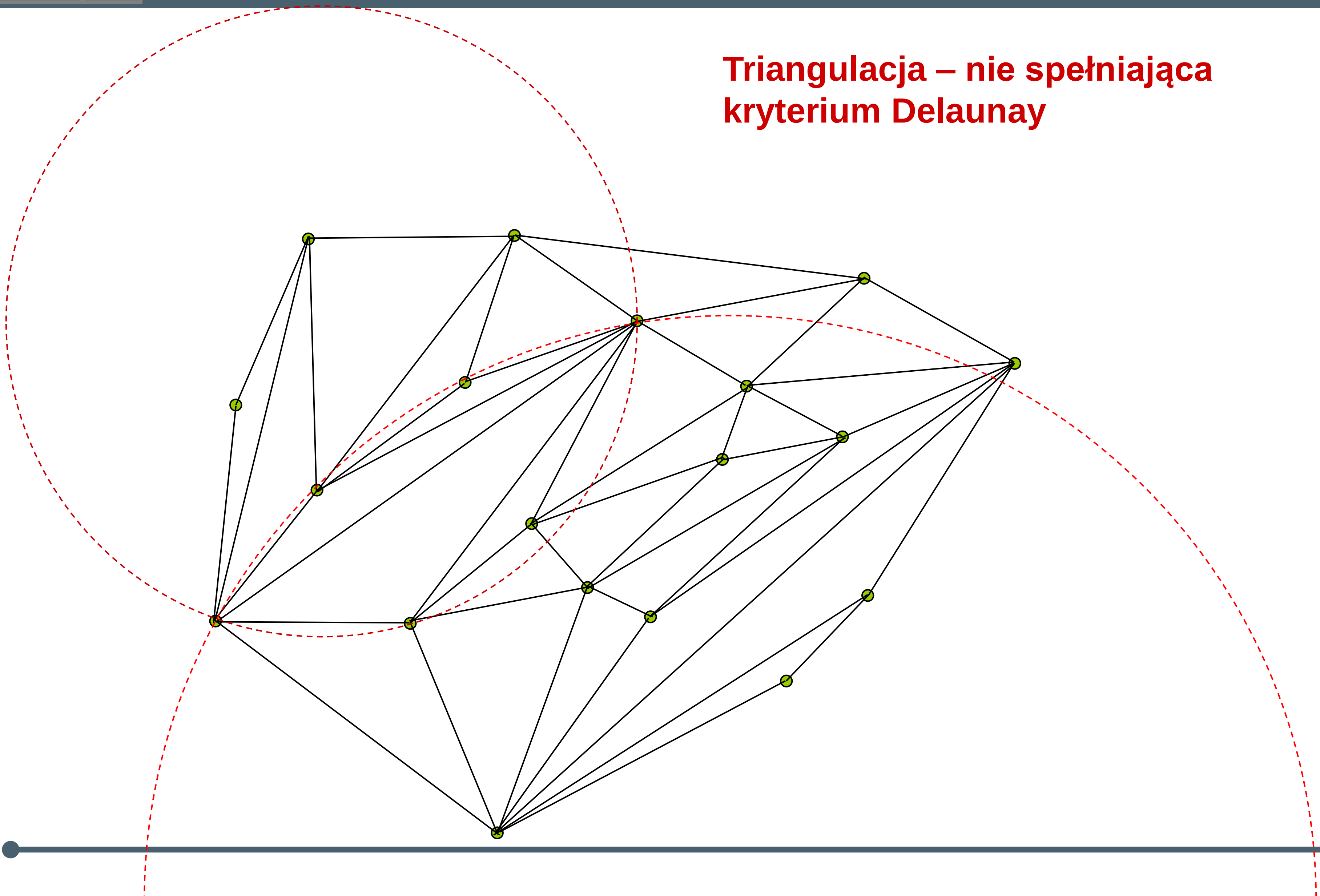


Promień okręgu wpisanego można obliczyć ze wzoru:

$$r = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}} \qquad r = \frac{2P}{a+b+c}$$

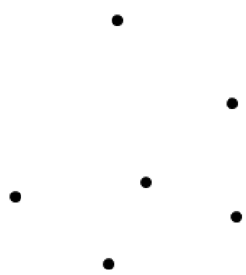
a, b, c - to długości boków trójkąta
p - to połowa obwodu trójkąta

Triangulacja – nie spełniająca kryterium Delaunay



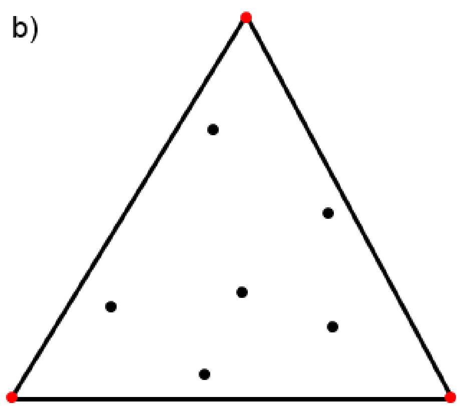
Danymi wejściowymi dla algorytmu jest zbiór punktów na których rozpięta będzie siatka MES.

a)

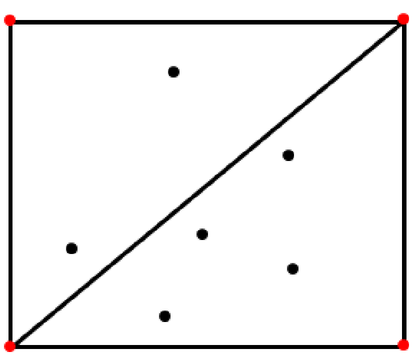


Chmura punktów jest zamykana w początkowym elemencie trójkątnym lub prostokątnym podzielonym na dwa trójkąty (super trójkąt/prostokąt).

b)

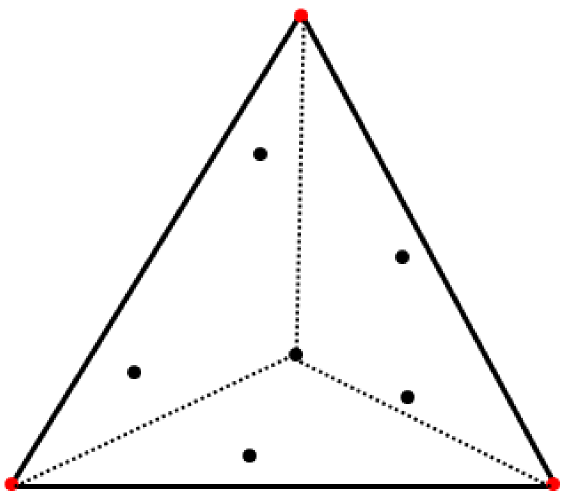


c)



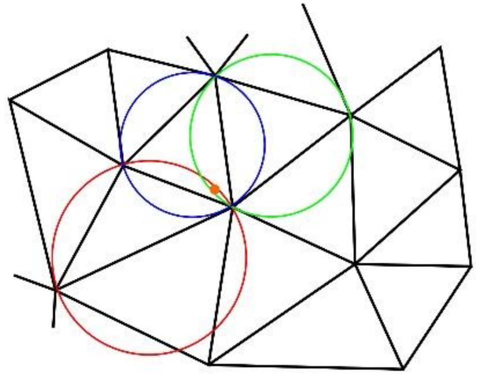
Punkty ze zbioru dostępnych punktów są systematycznie dodawane do siatki.

Każde dodanie nowego punktu skutkuje miejscową przebudową siatki.

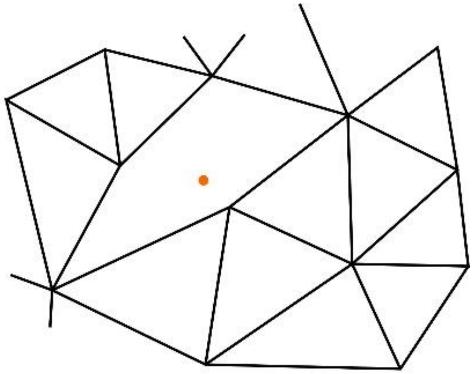


Dodawanie nowego punktu odbywa się zgodnie z jednym z dwóch kryteriów!

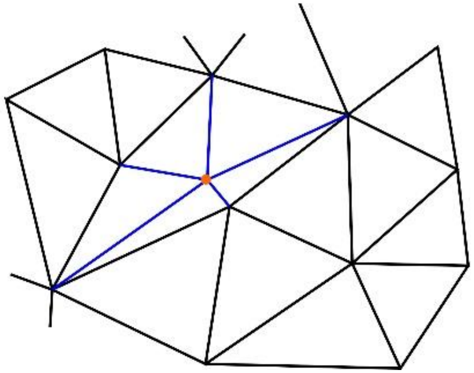
- Kryterium okręgu. Wykorzystywane jest do wyszukiwania z listy dostępnych trójkątów, które zawierają punkt, w którym wykonywana jest retriangulacja. Podczas wykonywania tego kryterium tworzony jest okrąg opisany na badanym trójkącie i sprawdzana jest ilość punktów zawierających się w okręgu. Warunek istnienia badanego punktu w liście punktów leżących w okręgu jest warunkiem określającym zawarcie punktu w trójkącie.



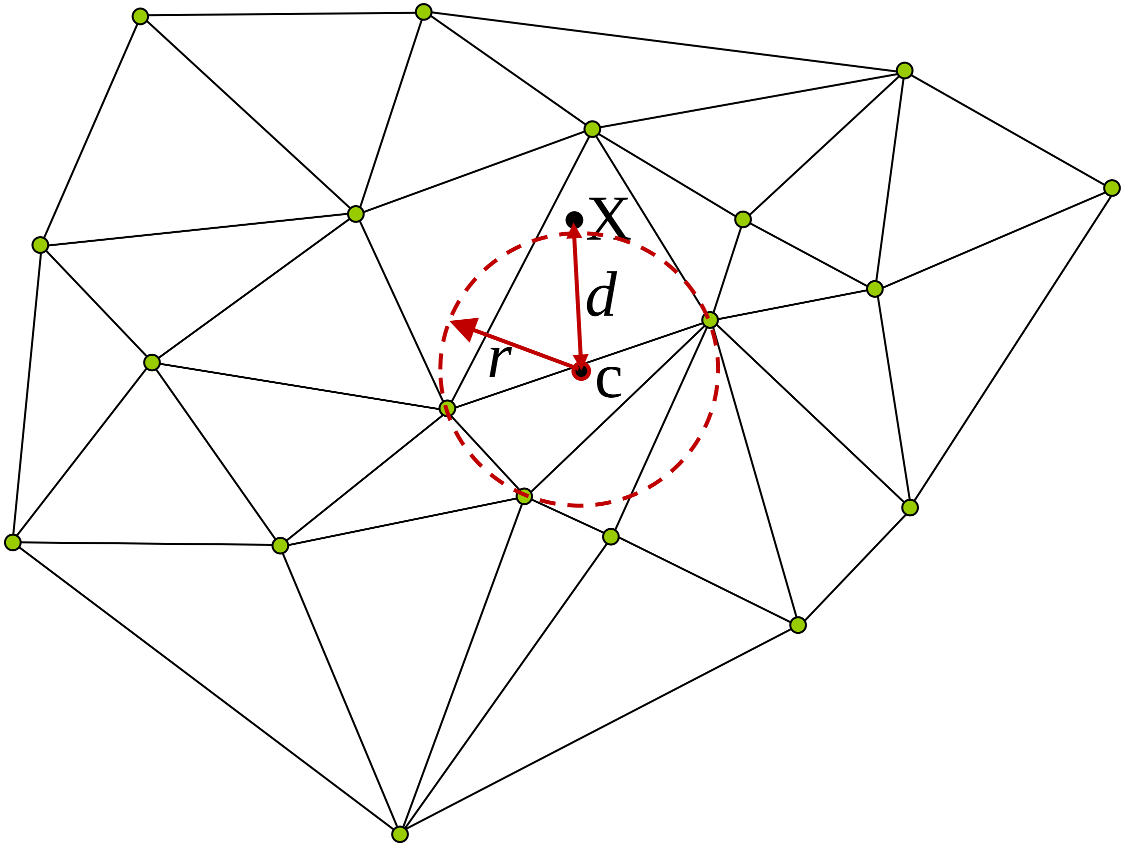
a)



b)



c)



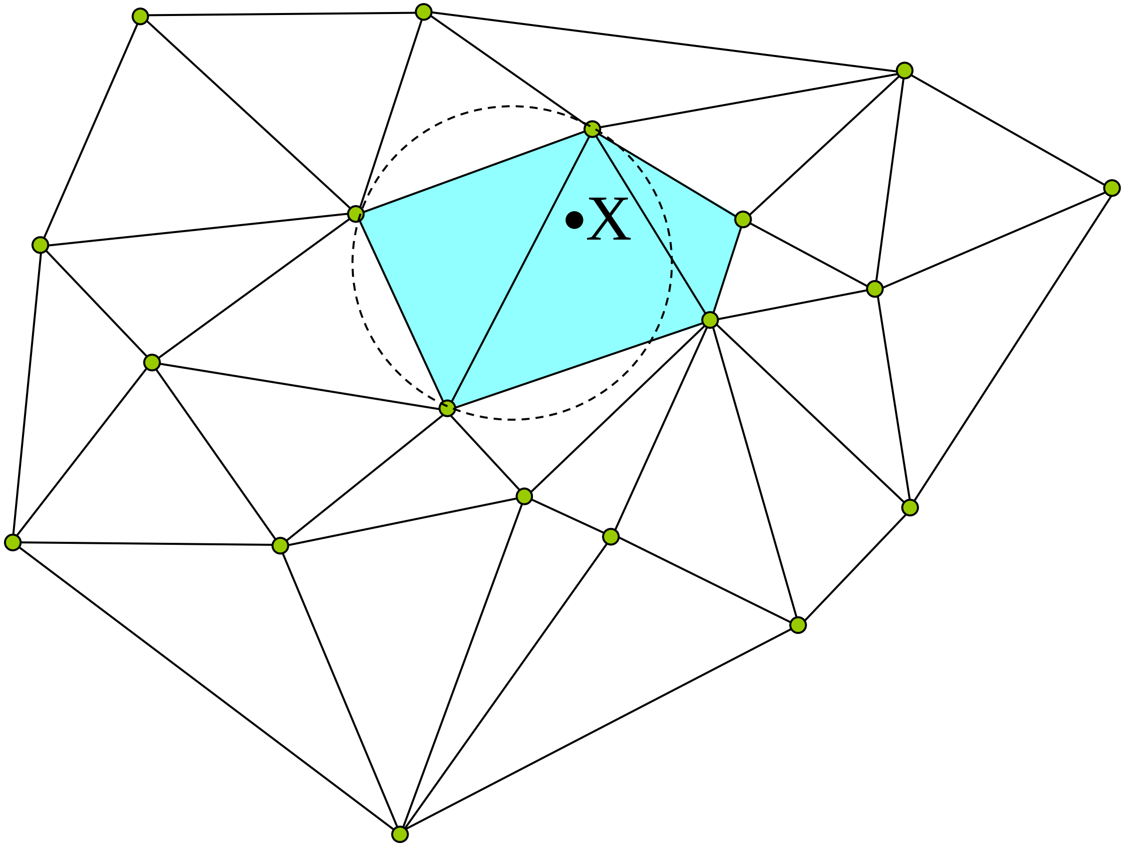
Bowyer-Watson Algorithm

• Znajdź trójkąt zawierający dany punkt.

• Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)

• Usuń znalezione trókąty

• Stwórz nowe trójkąty z uwzględnieniem danego punktu.



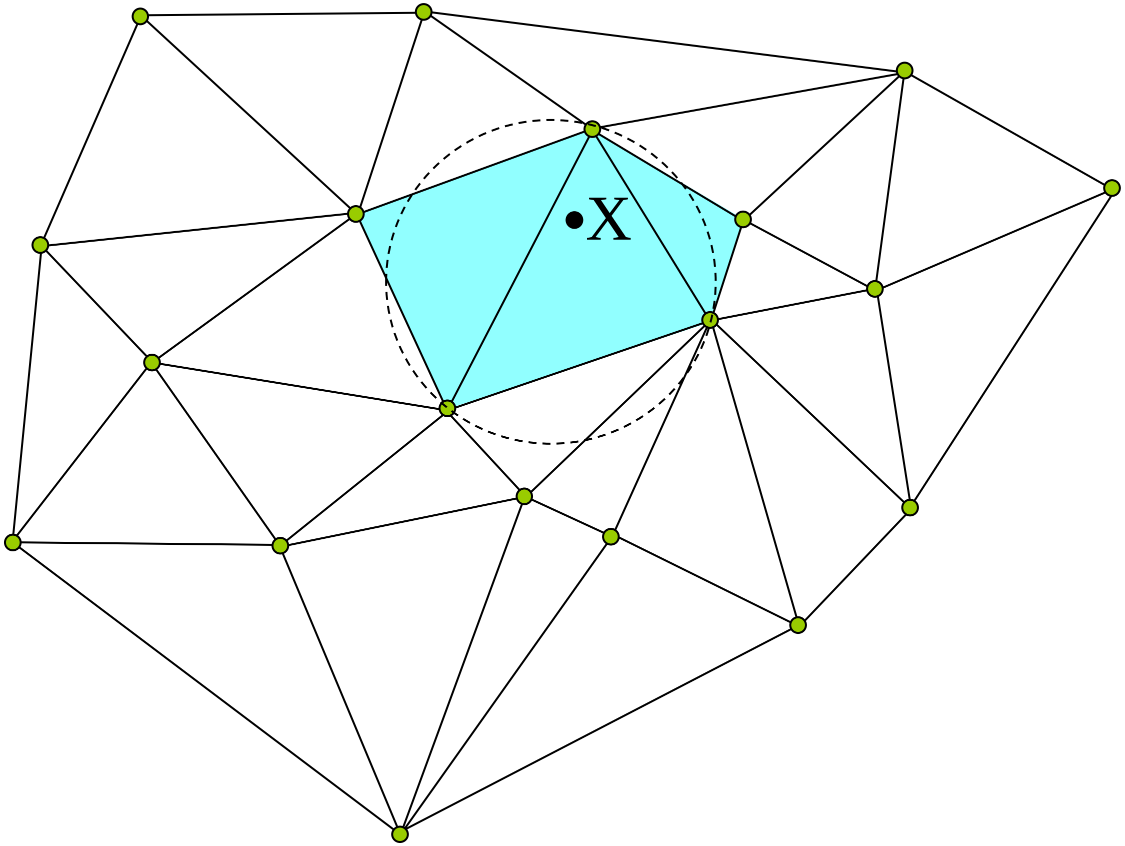
Bowyer-Watson Algorithm

• Znajdź trójkąt zawierający dany punkt.

• Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)

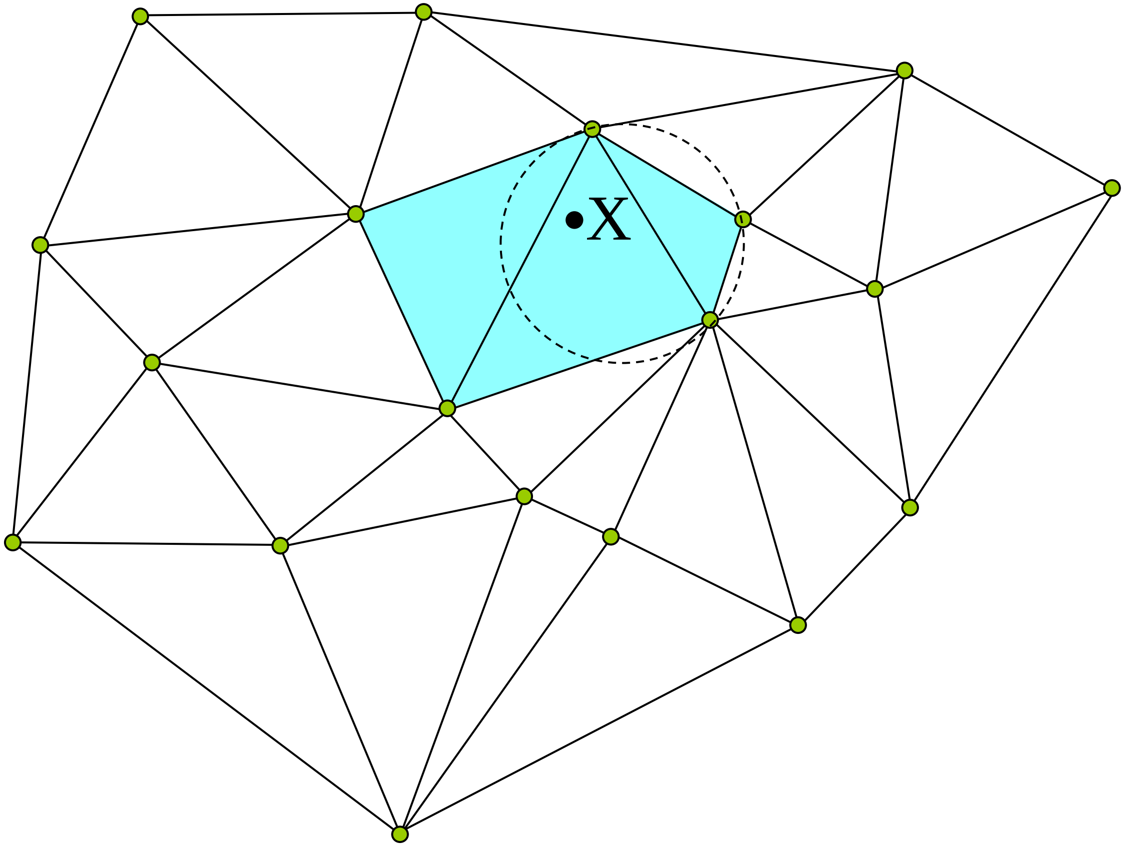
• Usuń znalezione trókąty

• Stwórz nowe trójkąty z uwzględnieniem danego punktu.



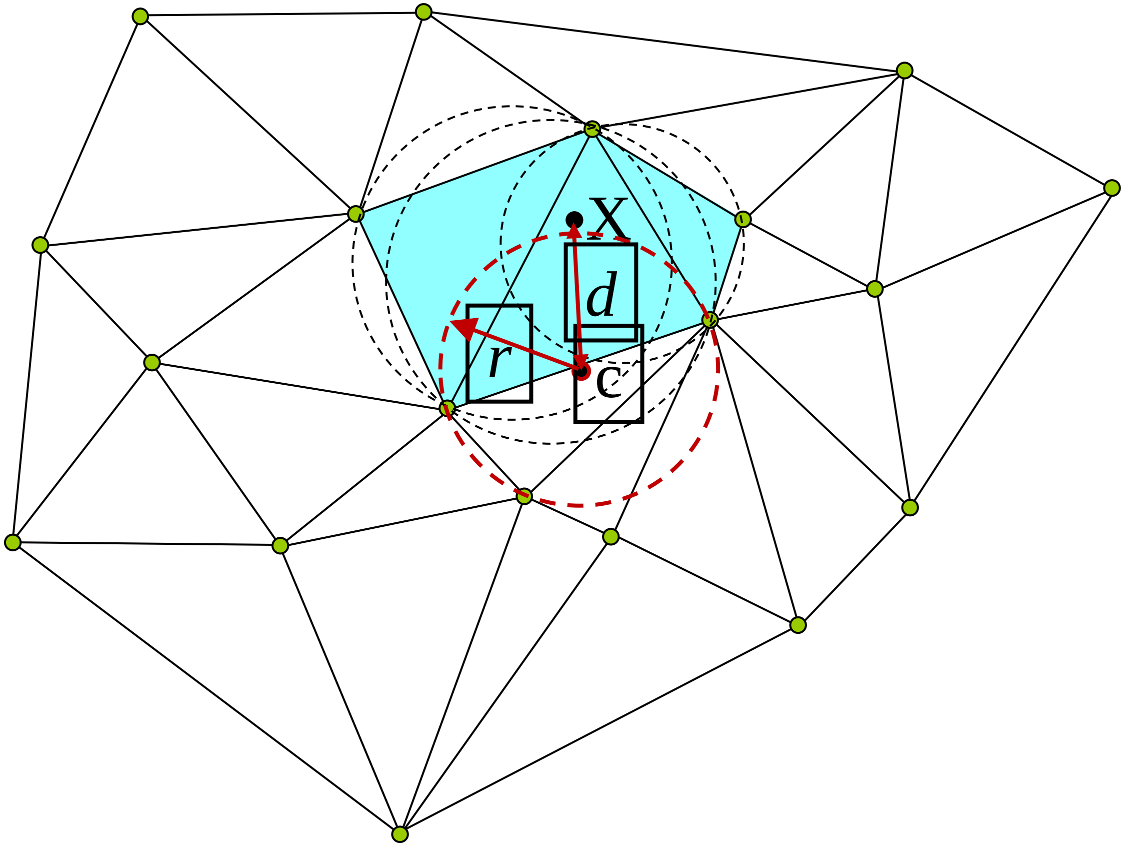
Bowyer-Watson Algorithm

- Znajdź trójkąt zawierający dany punkt.
- Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)
- Usuń znalezione trókąty
- Stwórz nowe trójkąty z uwzględnieniem danego punktu.



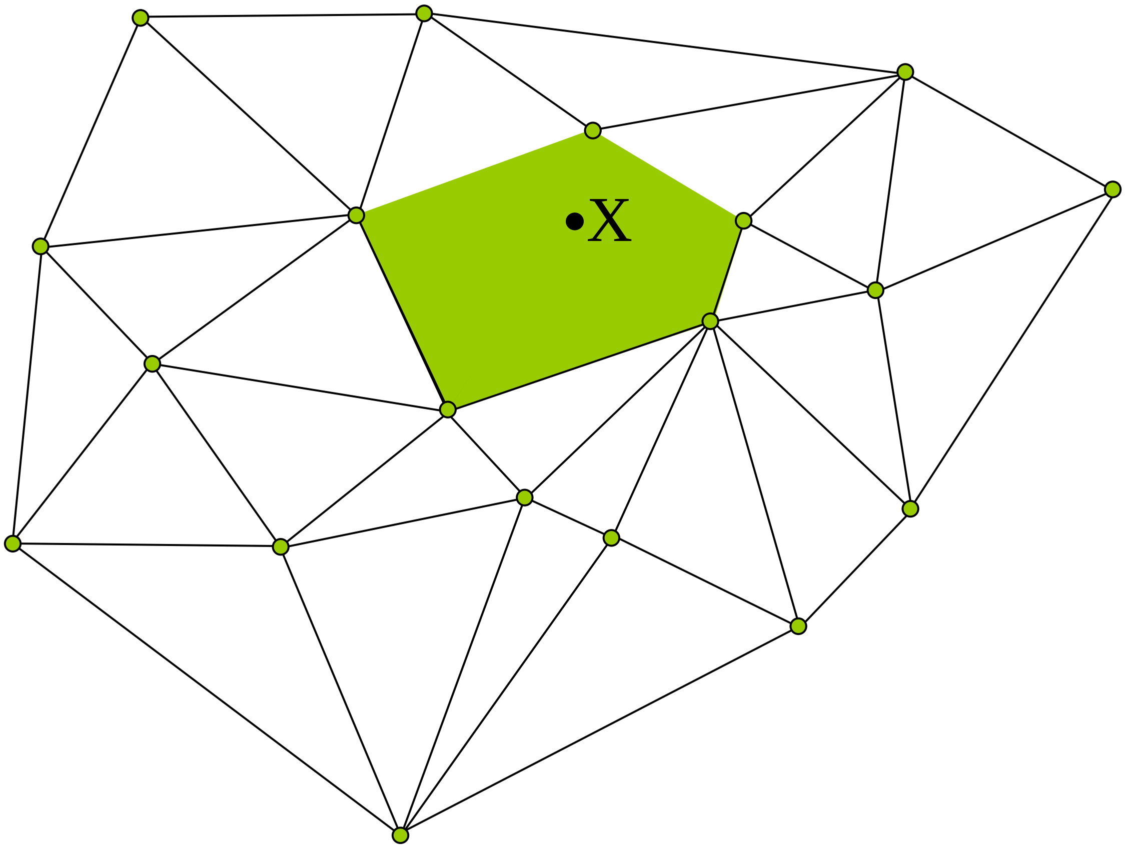
Bowyer-Watson Algorithm

- Znajdź trójkąt zawierający dany punkt.
- Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)
- Usuń znalezione trókąty
- Stwórz nowe trójkąty z uwzględnieniem danego punktu.



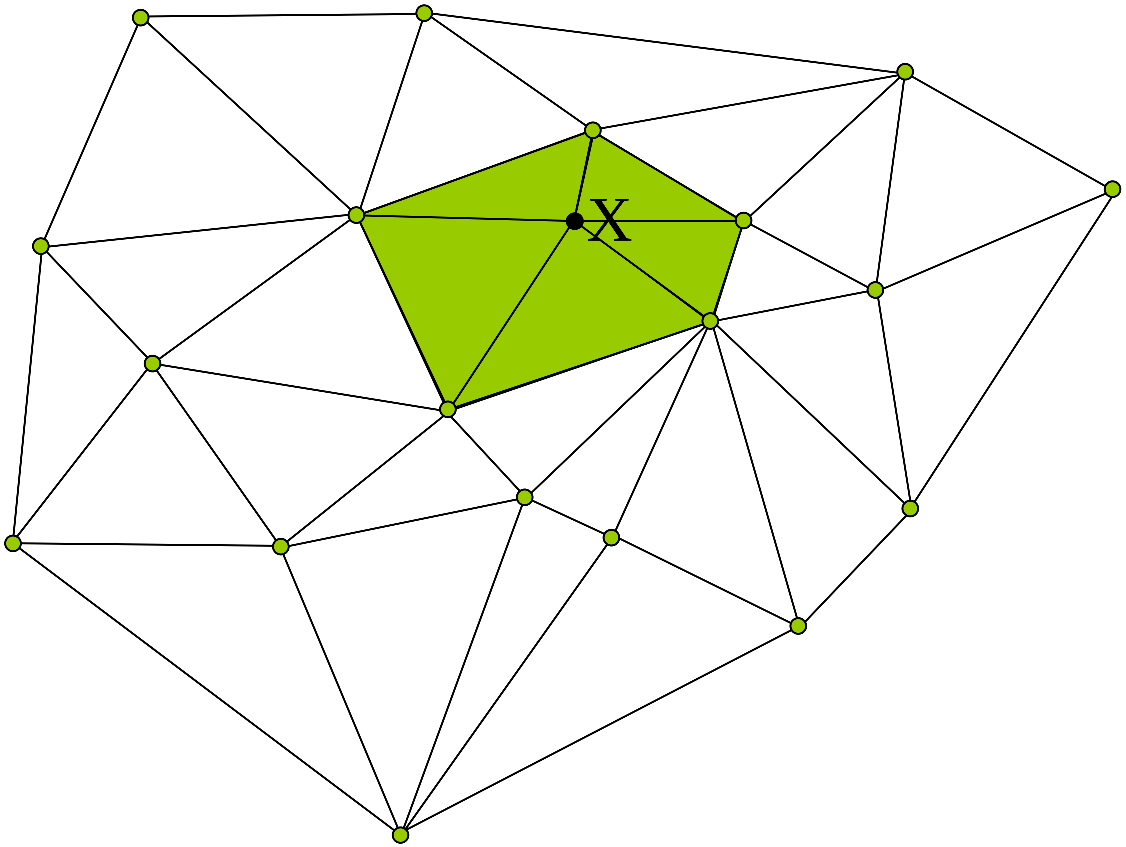
Bowyer-Watson Algorithm

- Znajdź trójkąt zawierający dany punkt.
- Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)
- Usuń znalezione trókąty
- Stwórz nowe trójkąty z uwzględnieniem danego punktu.



Bowyer-Watson Algorithm

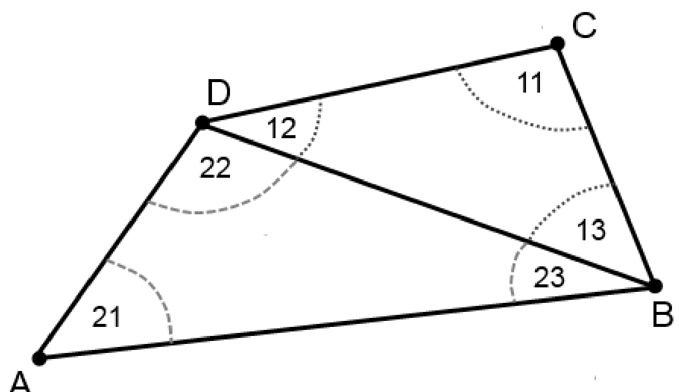
- Znajdź trójkąt zawierający dany punkt.
- Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)
- Usuń znalezione trókąty
- Stwórz nowe trójkąty z uwzględnieniem danego punktu.



Bowyer-Watson Algorithm

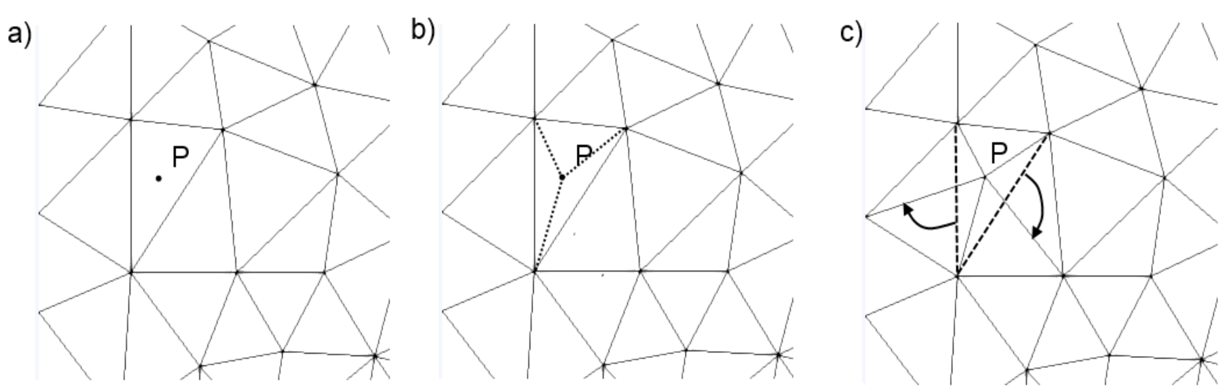
- Znajdź trójkąt zawierający dany punkt.
- Znajdź wszystkie trójkąty zwierające w swoim okręgu opisanym dany punkt ($d<r$)
- Usuń znalezione trókąty
- Stwórz nowe trójkąty z uwzględnieniem danego punktu.

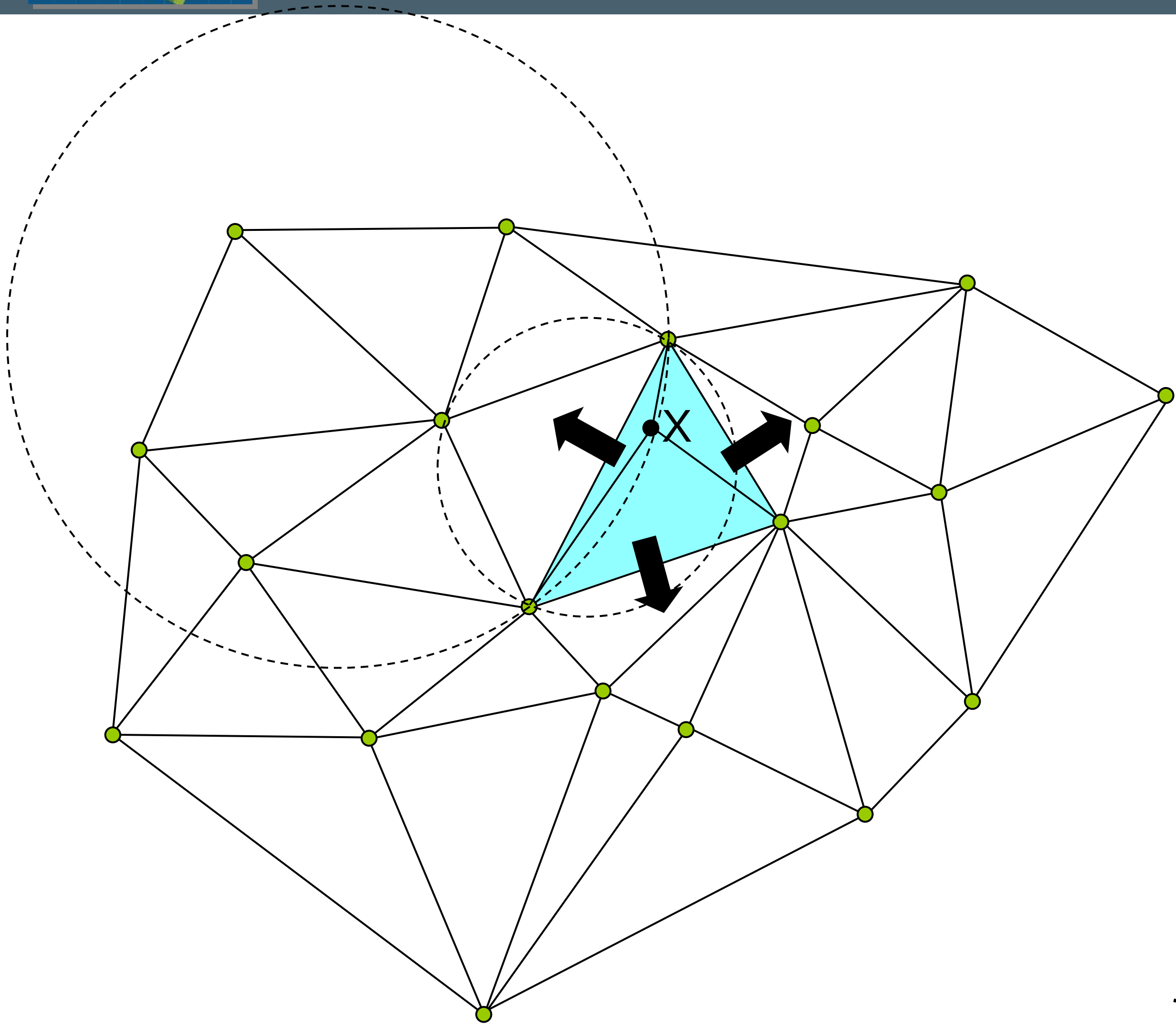
- Kryterium kątów wewnętrznych. Kryterium jest wykorzystywane podczas etapu powiększania minimalnych kątów generowanych trójkątów. W rezultacie trójkąty dążą do trójkątów o kształcie jak najbardziej zbliżonym do równobocznego. Po utworzeniu nowego trójkąta obliczana jest suma kątów przyległych do nowo utworzonej krawędzi i porównywana z sumą kątów nienależących do krawędzi.



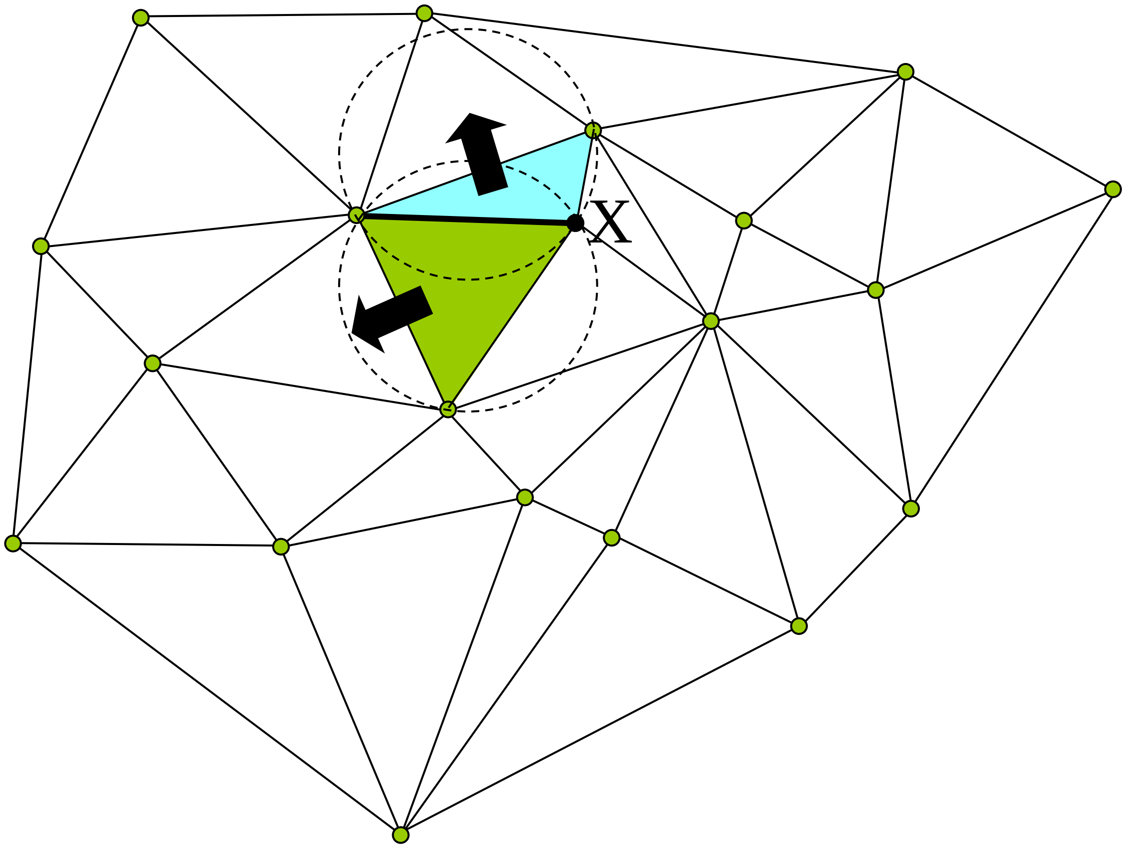
$$\alpha_{12}+\alpha_{13}+\alpha_{22}+\alpha_{23}\geq \alpha_{11}+\alpha_{21}$$

W przypadku gdy powyższe kryterium jest spełnione, krawędzie ulegają zamianie. Krawędź łącząca punkty (DB) zostaje usunięta, i utworzona zostaje krawędź łącząca punkty (AC).





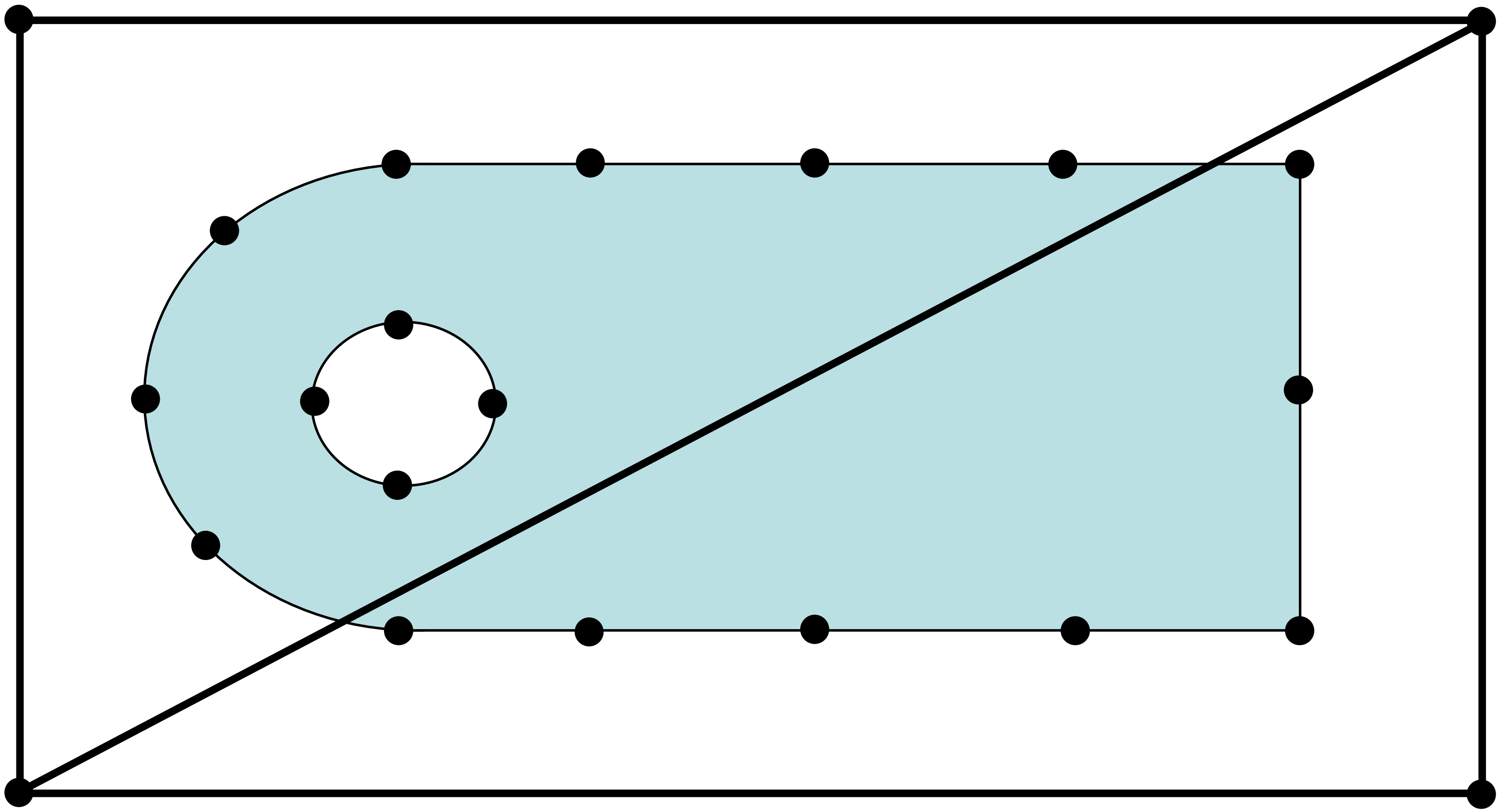
Algorytm Lawsona
- Znajdź trójkąt zwierający analizowany punkt X
- Podziel trójkąt
- Sprawdź kolejne trójkąty czy spełniają kryterium diagramu Dellanaya.
- Zamień krawędzie jeżeli jest taka konieczność.



Algorytm Lawsona

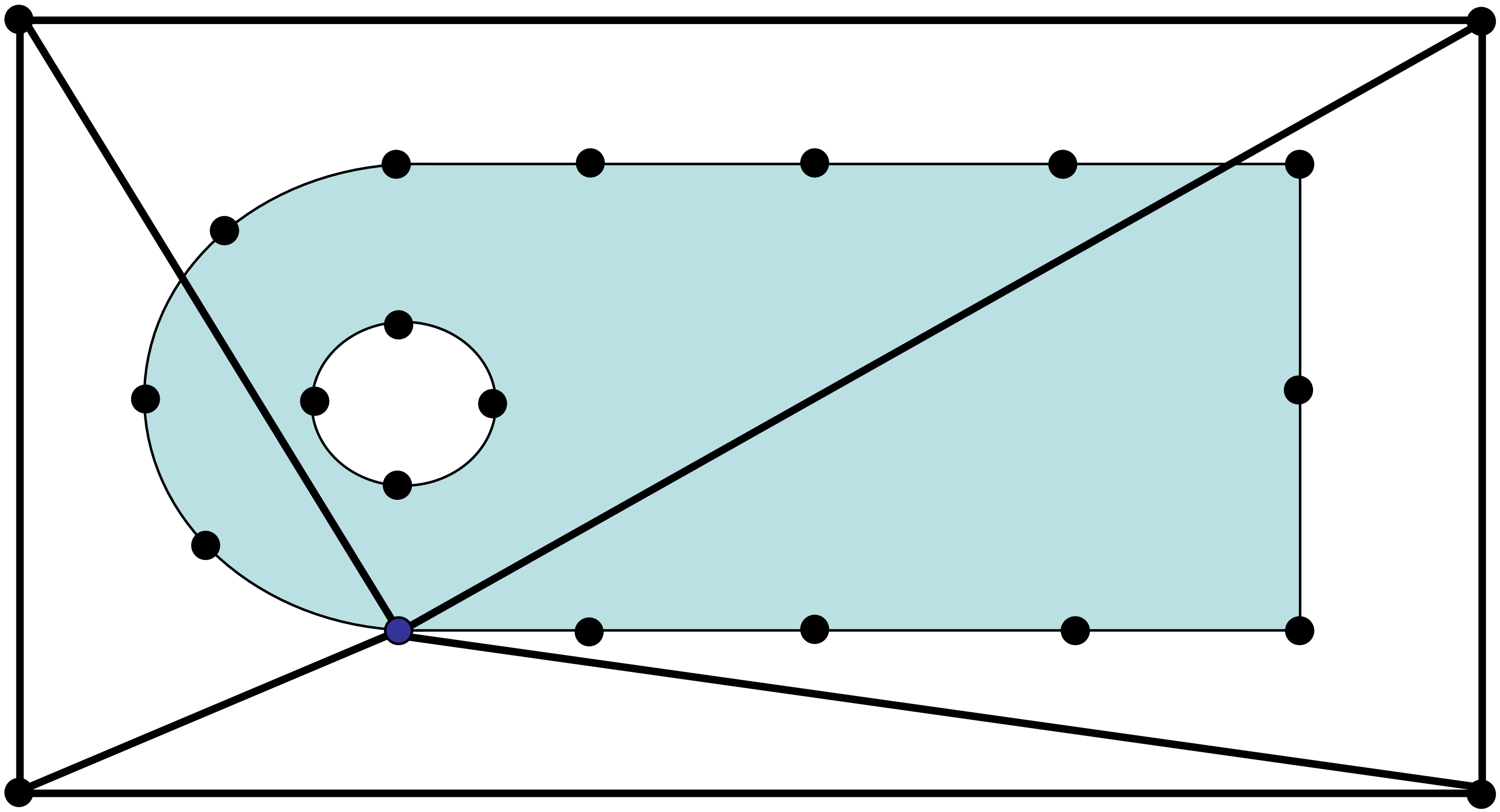
- Znajdź trójkąt zwierający analizowany punkt X
- Podziel trójkąt
- Sprawdź kolejne trójkąty czy spełniają kryterium diagramu Dellanaya.
- Zamień krawędzie jeżeli jest taka konieczność.

Delaunay



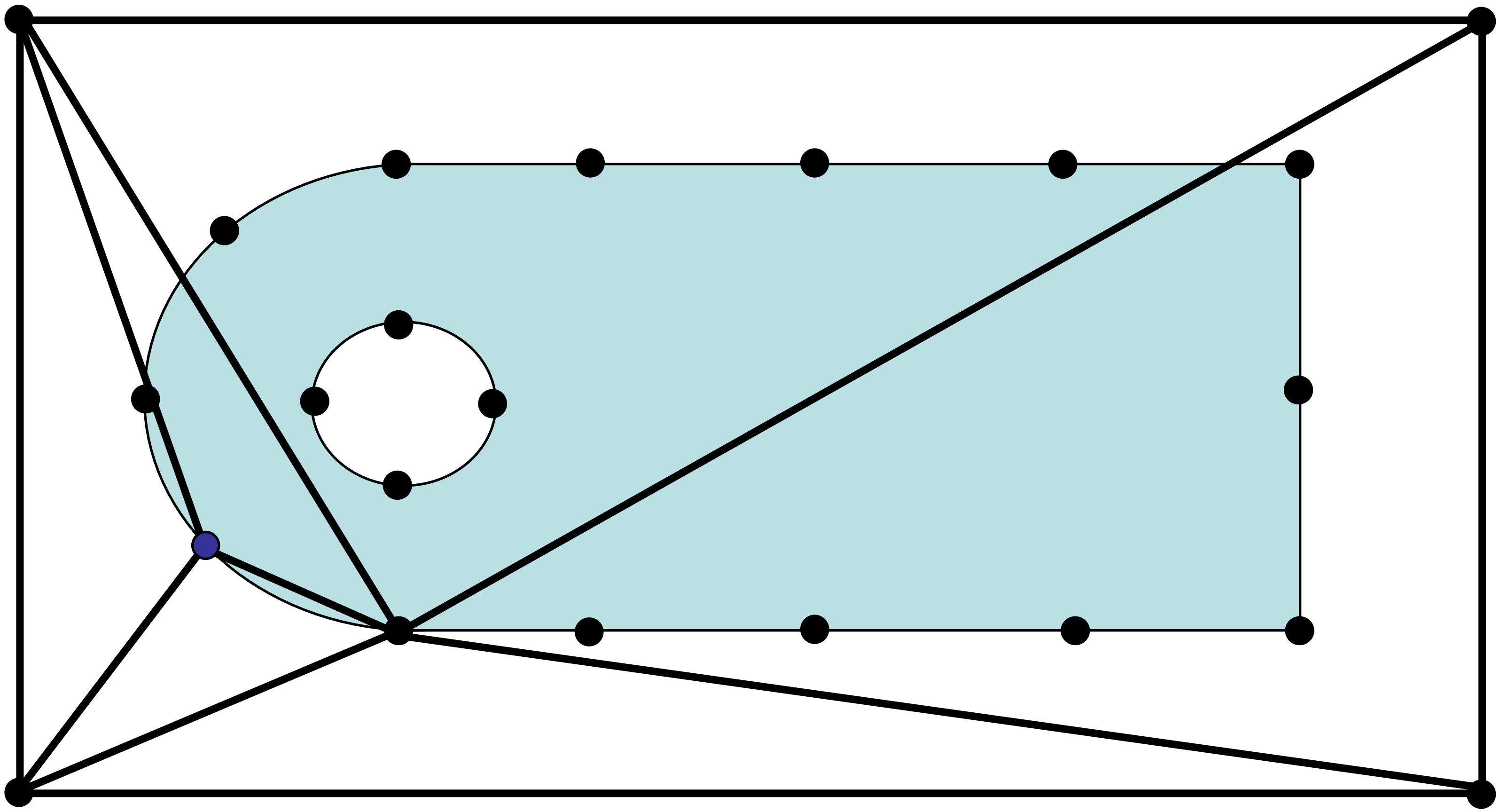
• stwórz super trójkąt/prostokąt

Delaunay



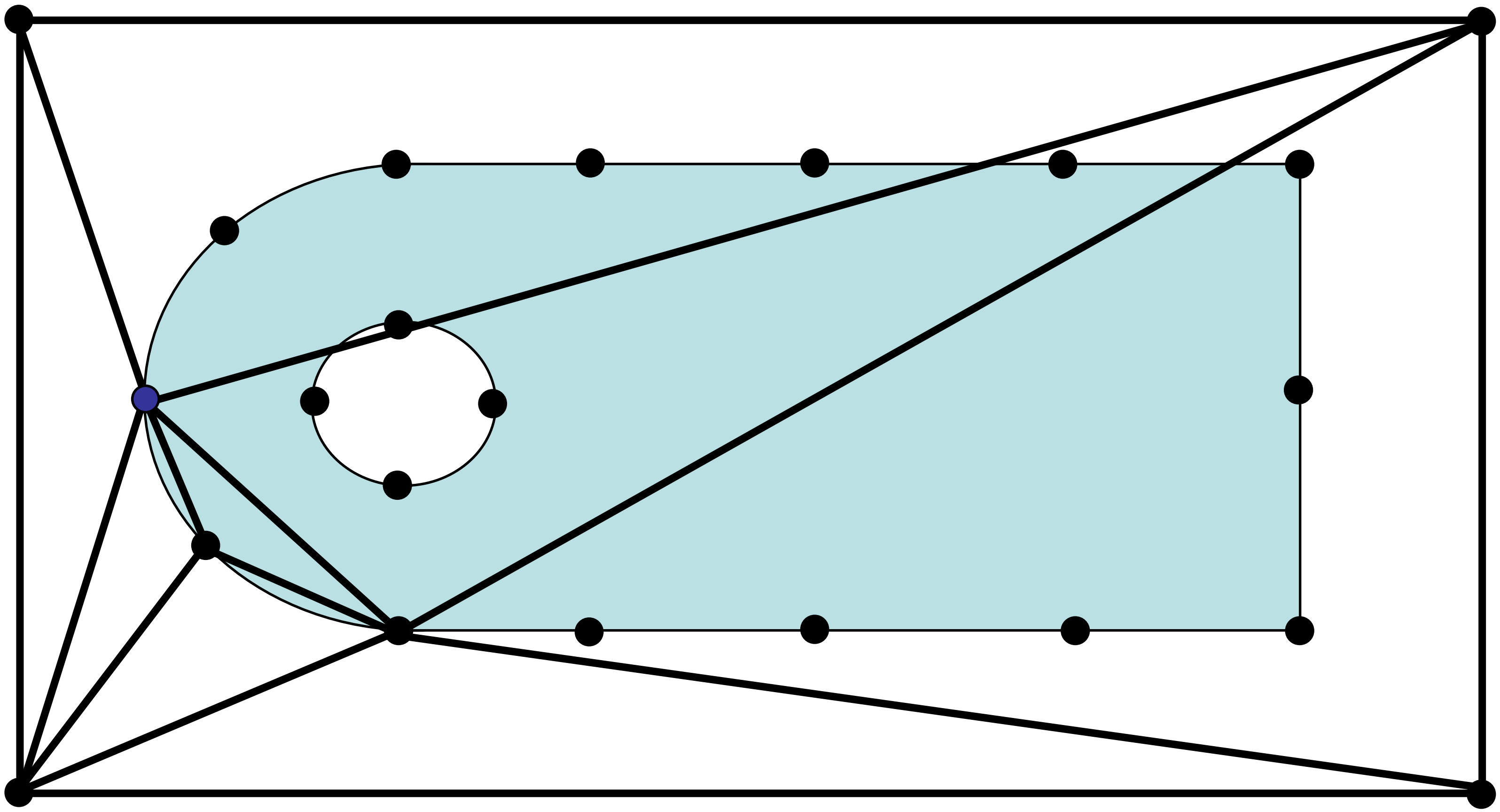
• wstaw nowy punkt korzystając z algorytmów Lawsona lub Bowyera-Watsona

Delaunay



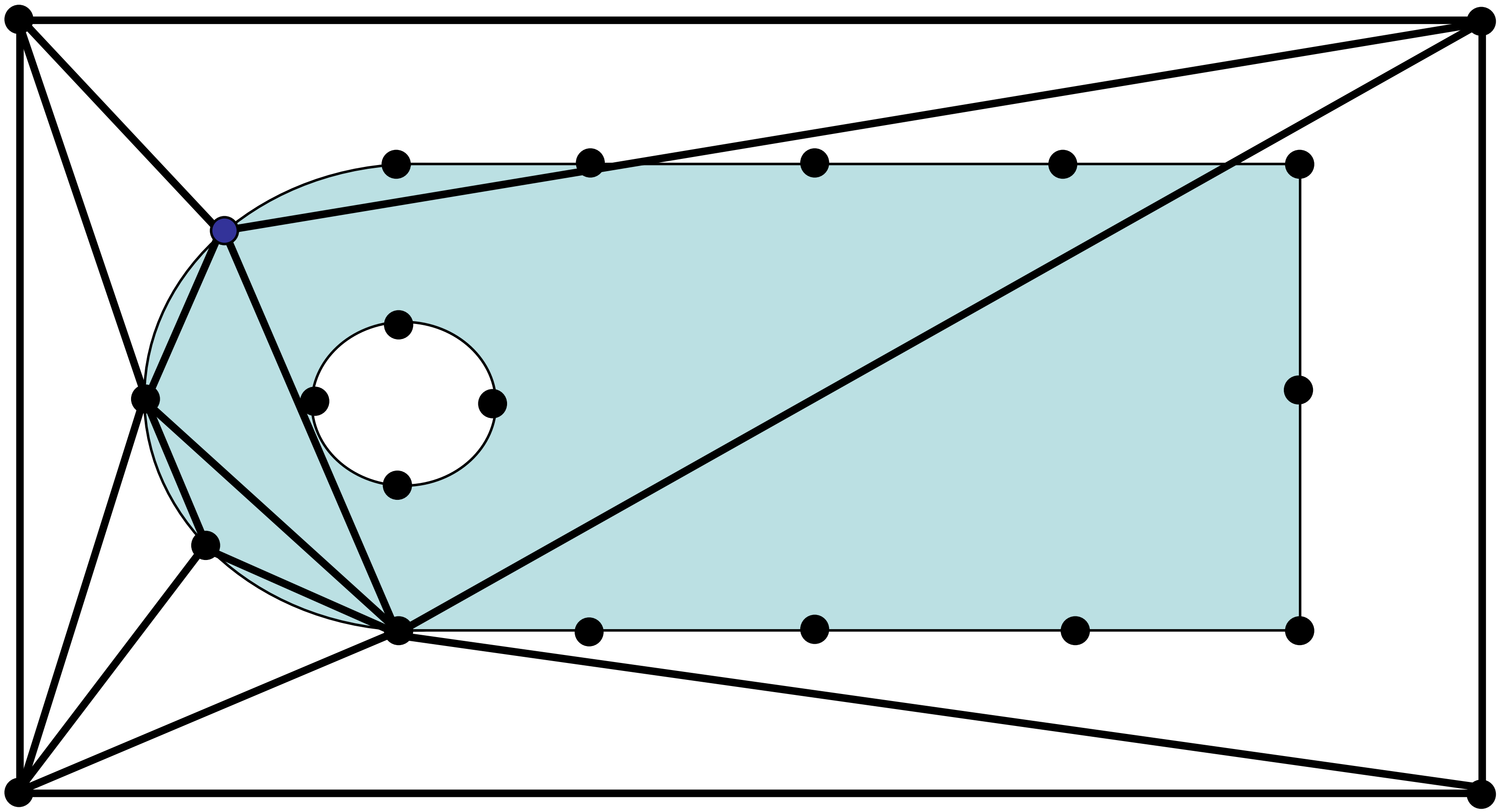
• wstaw nowy punkt korzystając z algorytmów Lawsona lub Bowyera-Watsona

Delaunay



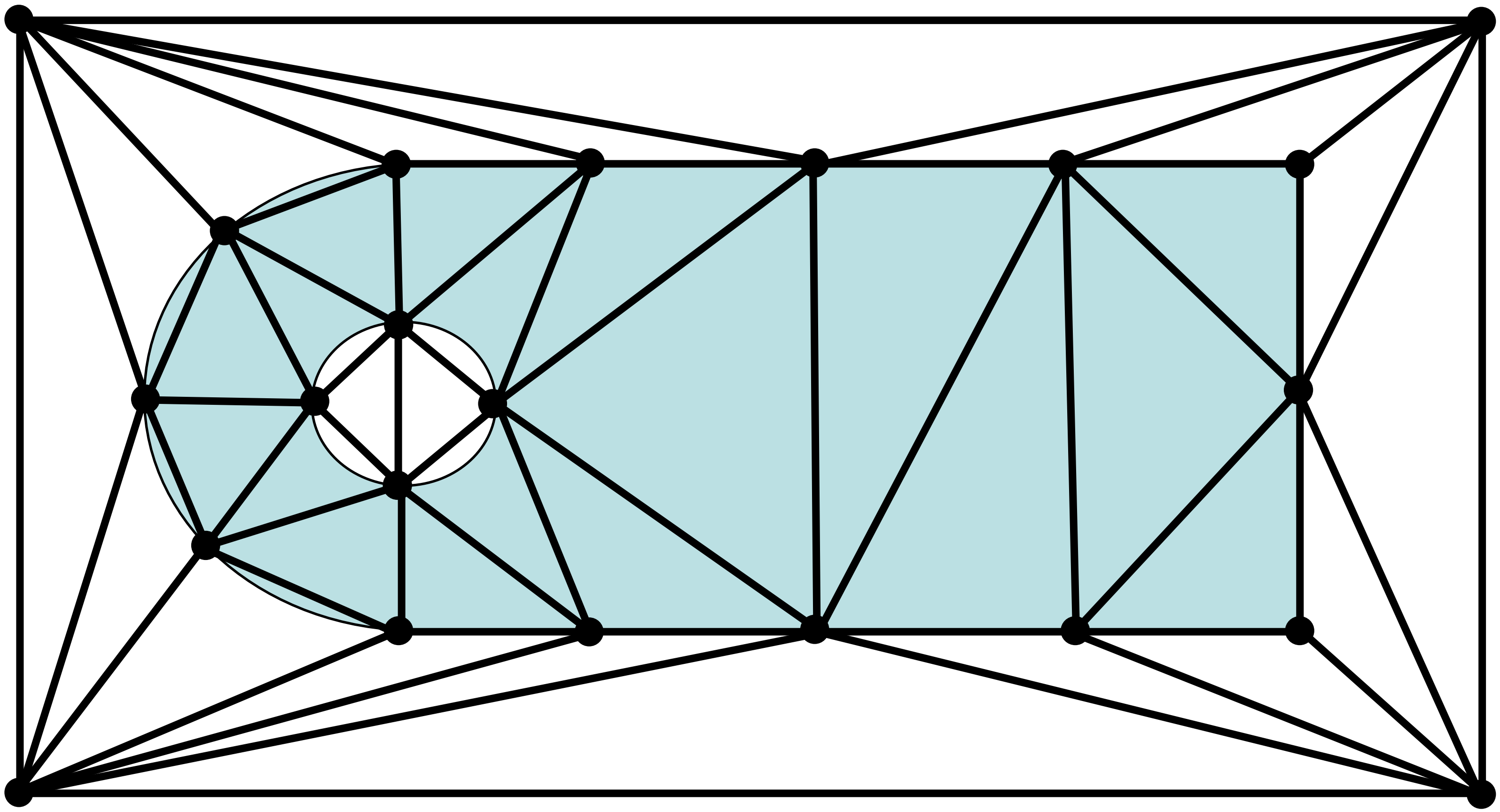
• wstaw nowy punkt korzystając z algorytmów Lawsona lub Bowyera-Watsona

Delaunay



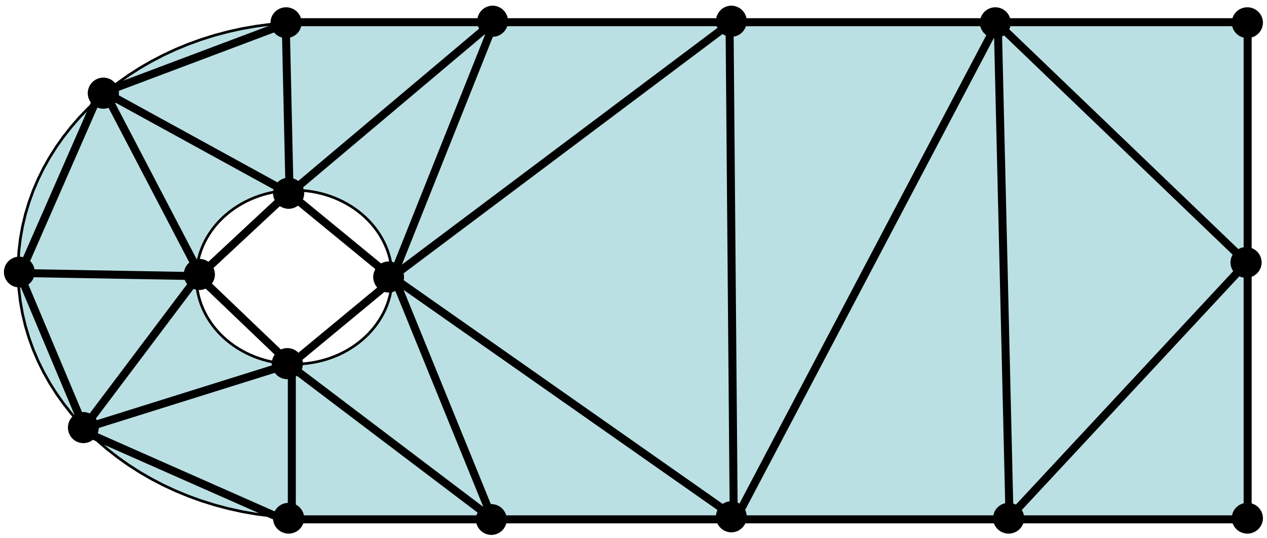
• wstaw nowy punkt korzystając z algorytmów Lawsona lub Bowyera-Watsona

Delaunay



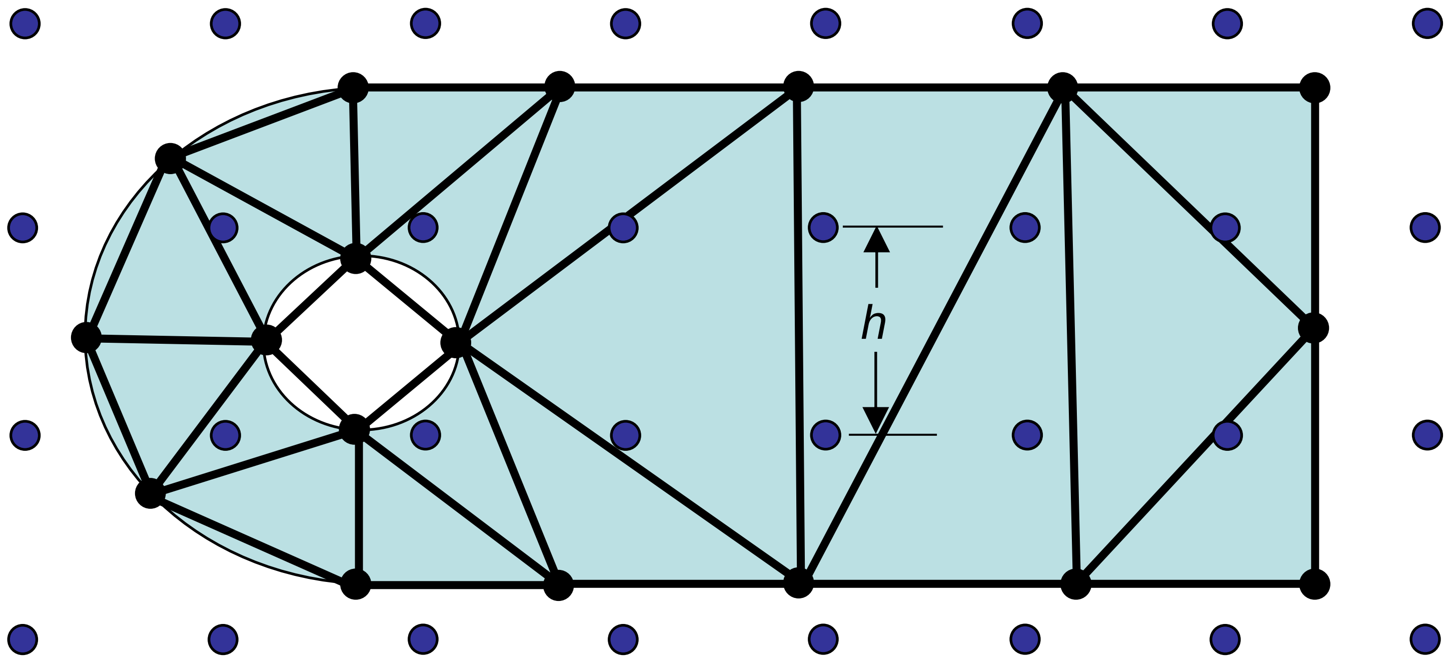
- wstaw nowy punkt korzystając z algorytmów Lawsona lub Bowyera-Watsona

Delaunay



- Usuń trójkąty zawierające węzły poza analizowanym obszarem Recover boundary
- Rozpocznij wstawianie węzłów wewnętrznych

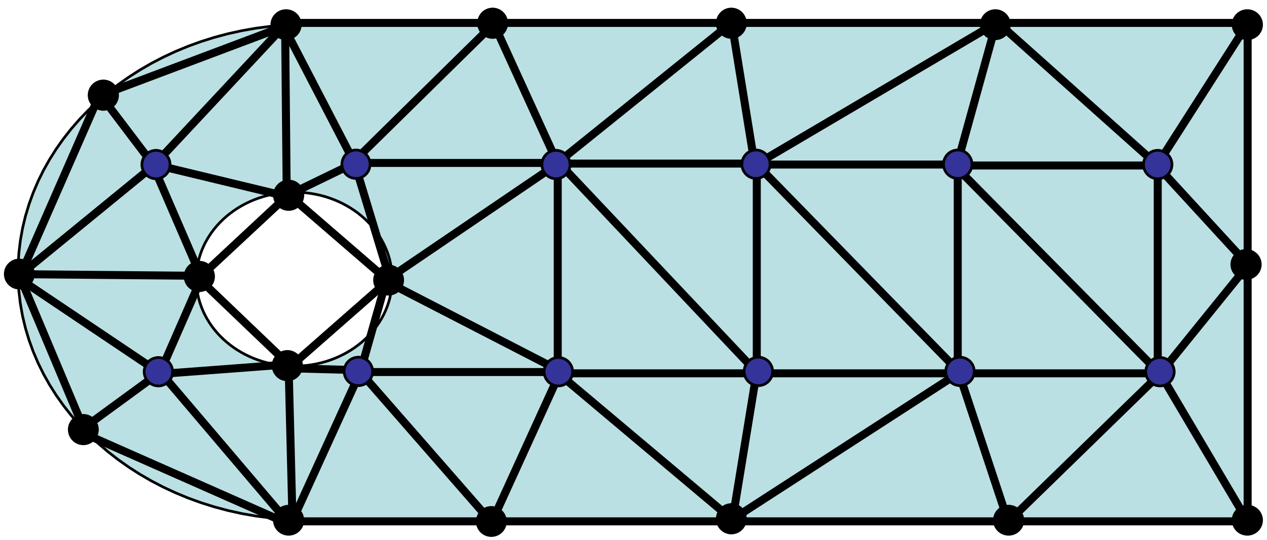
Delaunay



Wstawianie węzłów z wykorzystaniem predefiniowanej listy:

- Węzły wstawiane są na podstawie predefiniowanej listy np. regularnie rozłożonych punktów
- Rozłożenie punktów może być dowolne
- Punkty poza obszarem domeny obliczeniowej są ignorowane

Delaunay

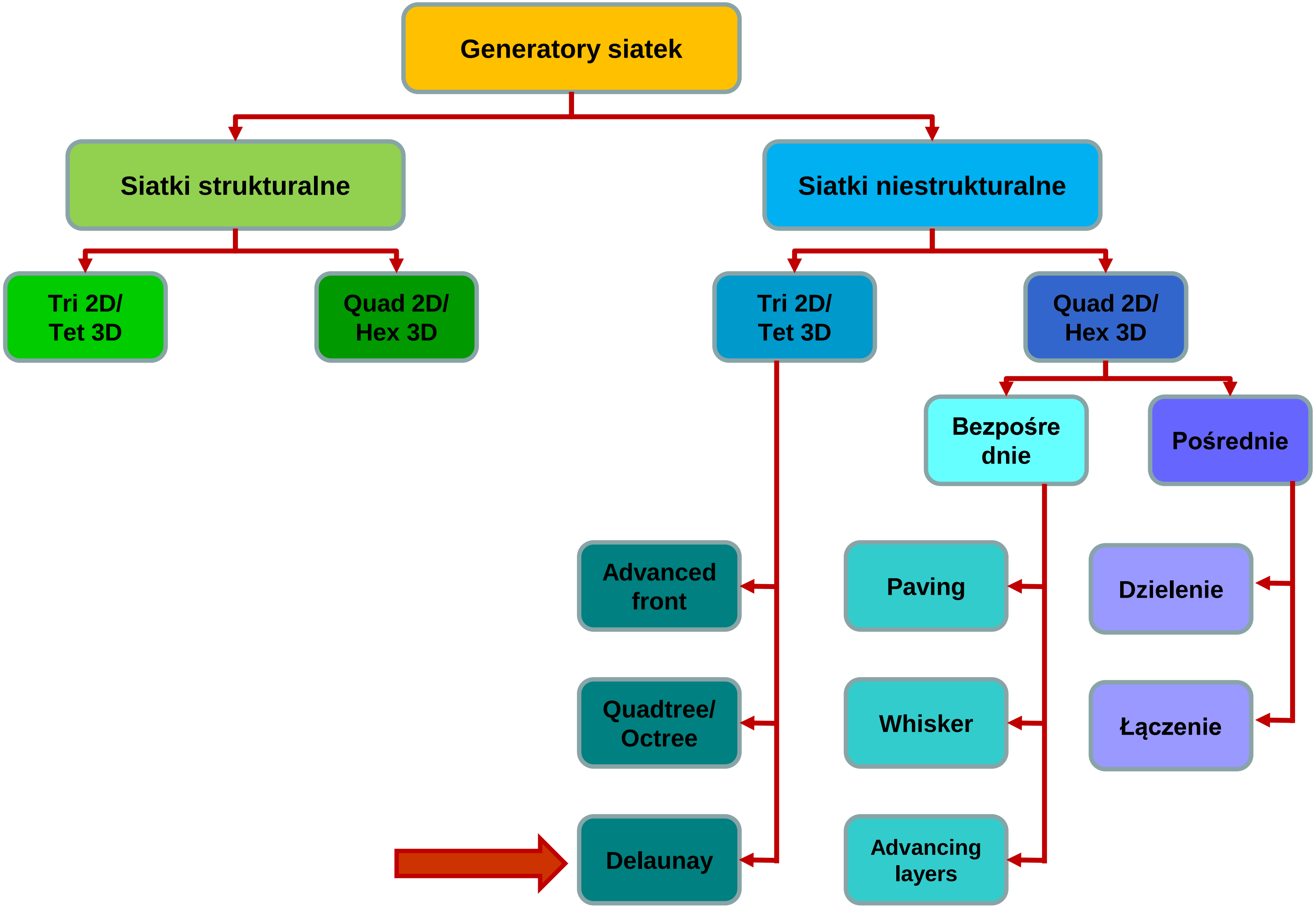


Wstawianie węzłów z wykorzystaniem predefiniowanej listy:

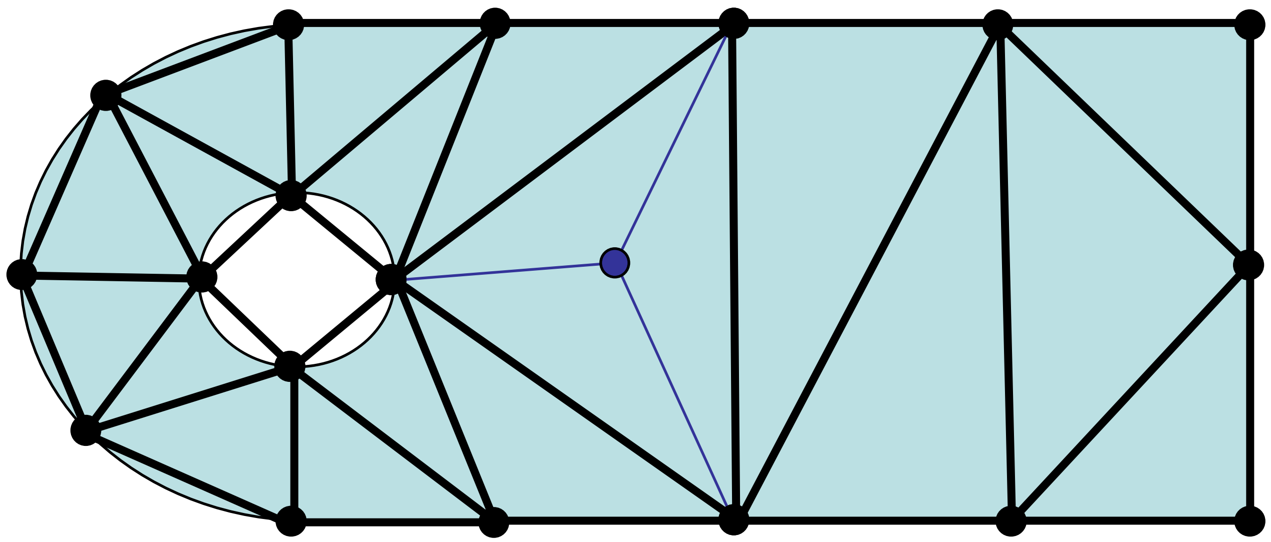
• Węzły wstawiane są na podstawie predefiniowanej listy np. regularnie rozłożonych punktów

• Rozłożenie punktów może być dowolne

• Punkty poza obszarem domeny obliczeniowej są ignorowane


Generatory siatek
Siatki strukturalne
Siatki niestrukturalne
Tri 2D/ Tet 3D
Quad 2D/ Hex 3D
Tri 2D/ Tet 3D
Quad 2D/ Hex 3D
Bezpośre dnie
Pośrednie
Advanced front
Quadtree/ Octree
Delaunay
Paving
Whisker
Advancing layers
Dzielenie
Łączenie

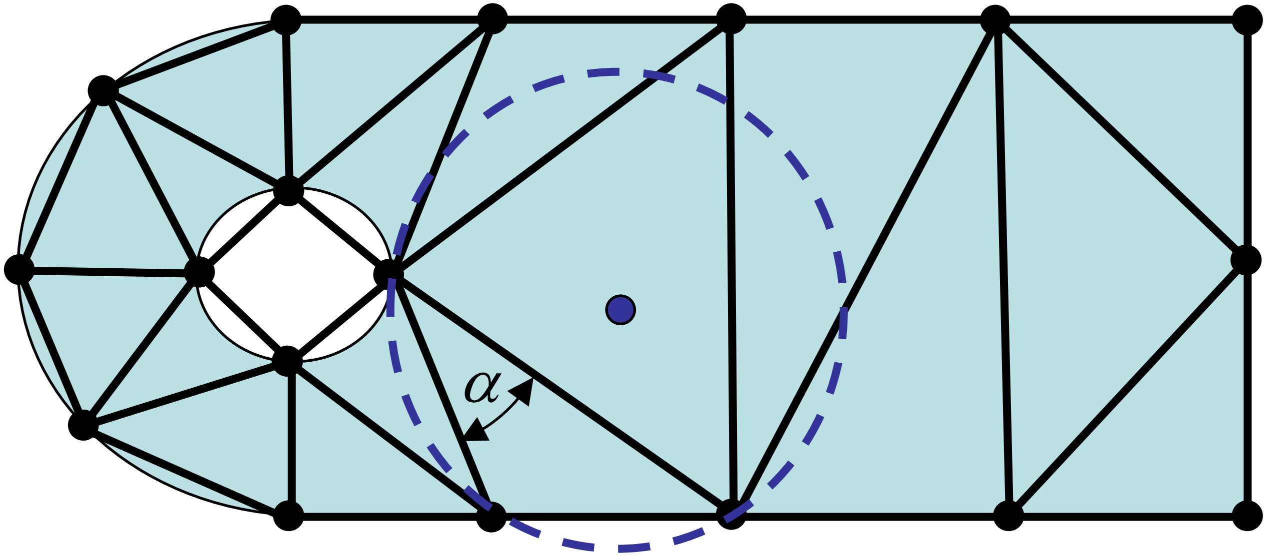
Delaunay



Wstawianie węzłów z wykorzystaniem środków ciężkości elementów:

• Węzły wstawiane są na podstawie określenia środków ciężkości kolejnych elementów.

Delaunay

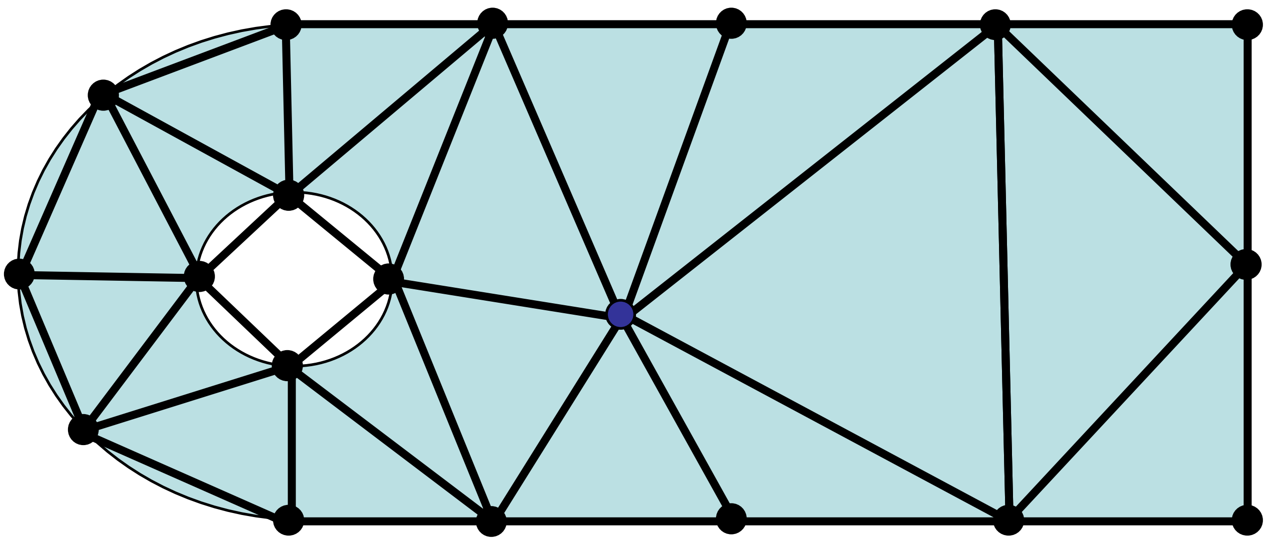


Wstawianie węzłów z wykorzystaniem środków okręgów opisanych:

• Węzły wstawiane są na podstawie określenia środków okręgów opisanych na kolejnych elementach.

• Wszystkie sąsiadujące trójkąty z najmniejszymi kątami mniejszymi od zdefiniowanej wartości krytycznej są usuwane. $(\alpha \approx 30^{\circ})$

Delaunay

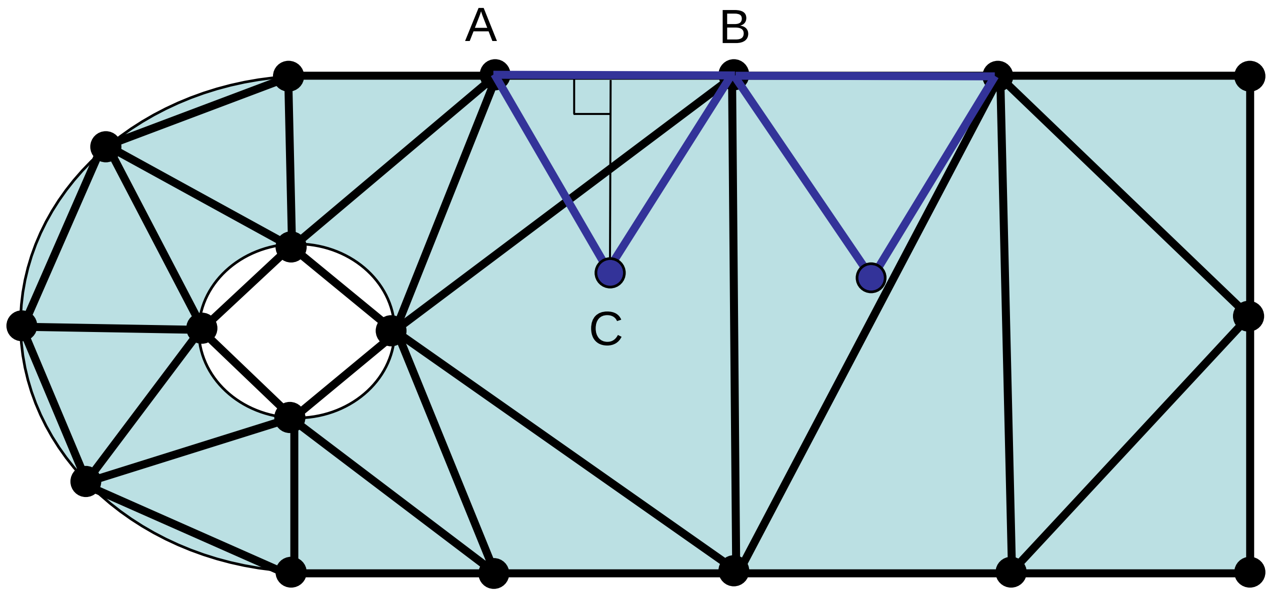


Wstawianie węzłów z wykorzystaniem środków okręgów opisanych:

• Węzły wstawiane są na podstawie określenia środków okręgów opisanych na kolejnych elementach.

• Wszystkie sąsiadujące trójkąty z najmniejszymi kątami mniejszymi od zdefiniowanej wartości krytycznej są usuwane.

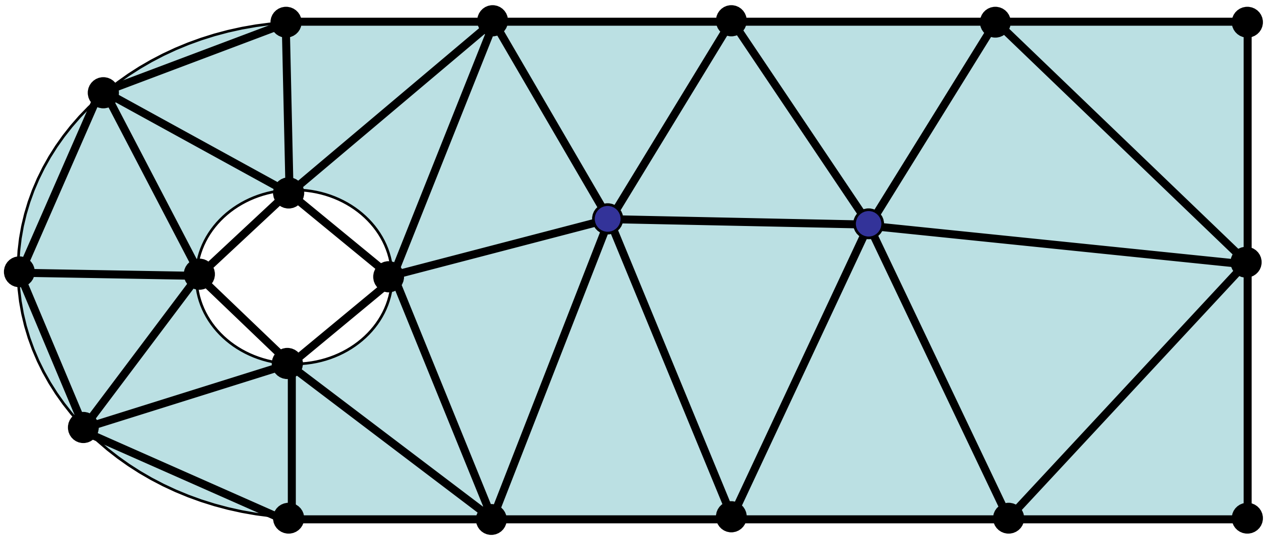
Delaunay



Wstawianie węzłów z wykorzystaniem postępującego frontu:

• Węzły wstawiane w „idealne” położenia na kolejnych odcinkach frontu.

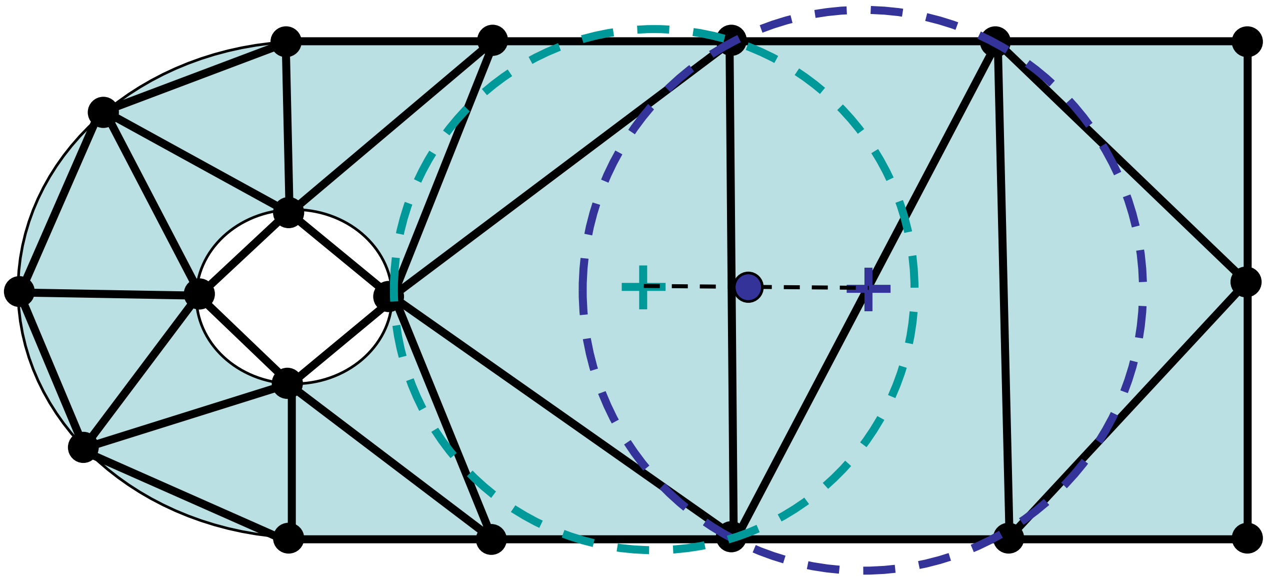
Delaunay



Wstawianie węzłów z wykorzystaniem postępującego frontu:

• Węzły wstawiane w „idealne” położenia na kolejnych odcinkach frontu.

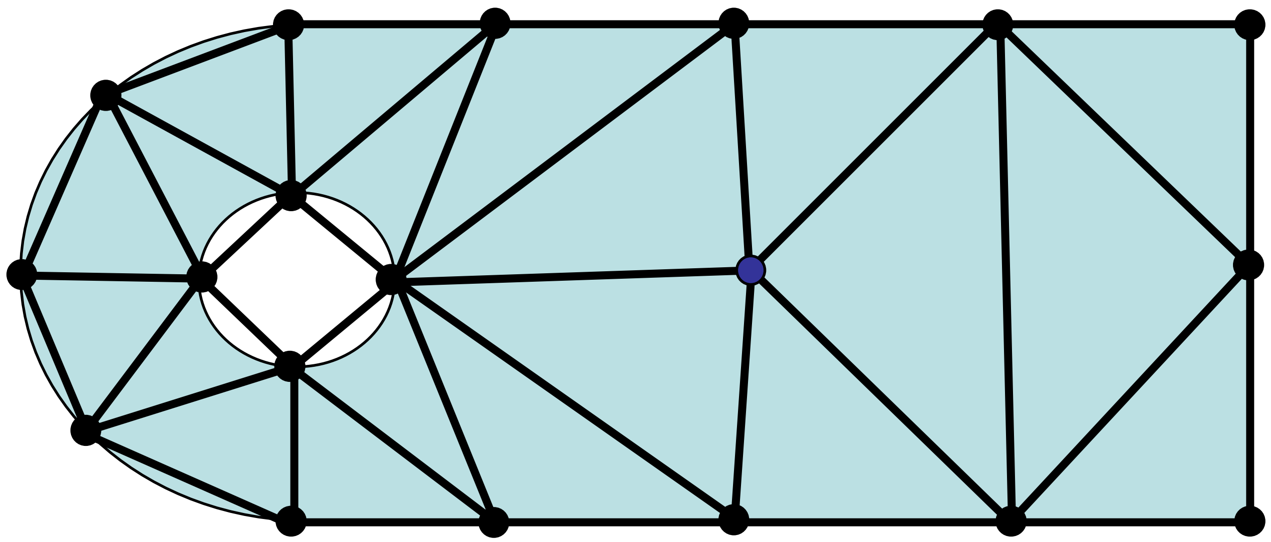
Delaunay



Wstawianie węzłów na bazie Segmentu-Voronoia.

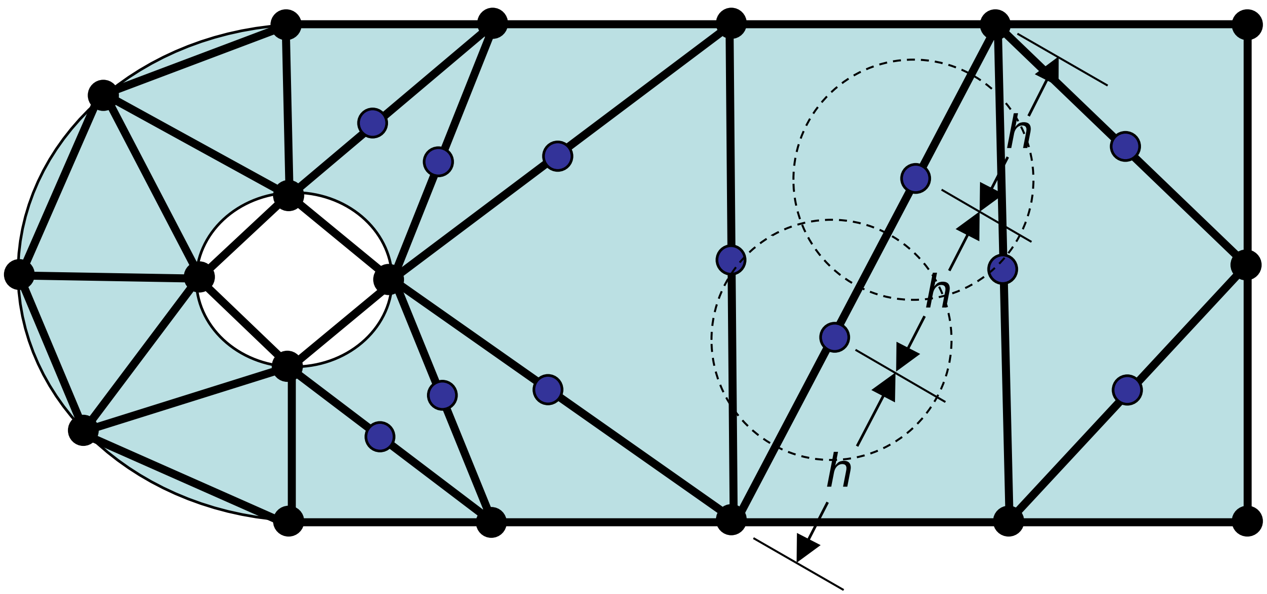
•węzły wstawiane w na środku odcinka (segment Voronoia) łączącego dwa środki okręgów opisanych na sąsiadujących trójkątach.

Delaunay



Wstawianie węzłów na bazie Segmentu-Voronoia.
•węzły wstawiane w na środku odcinka (segment Voronoia) łączącego dwa środki okręgów opisanych na sąsiadujących trójkątach.

Delaunay

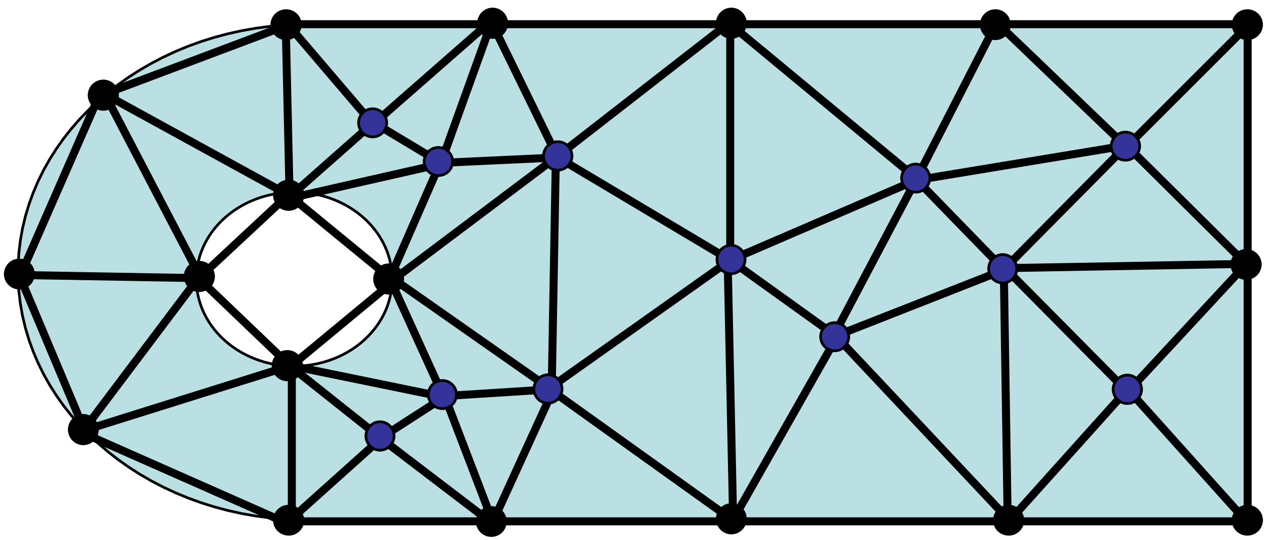


Wstawianie węzłów na brzegach istniejących trójkątów.

• węzły wstawiane są na brzegach istniejących trójkątów z założoną gęstością.

• wymagane jest sprawdzenie czy węzły na sąsiednich brzegach nie są zlokaliwoane zbyt blisko siebie.

Delaunay

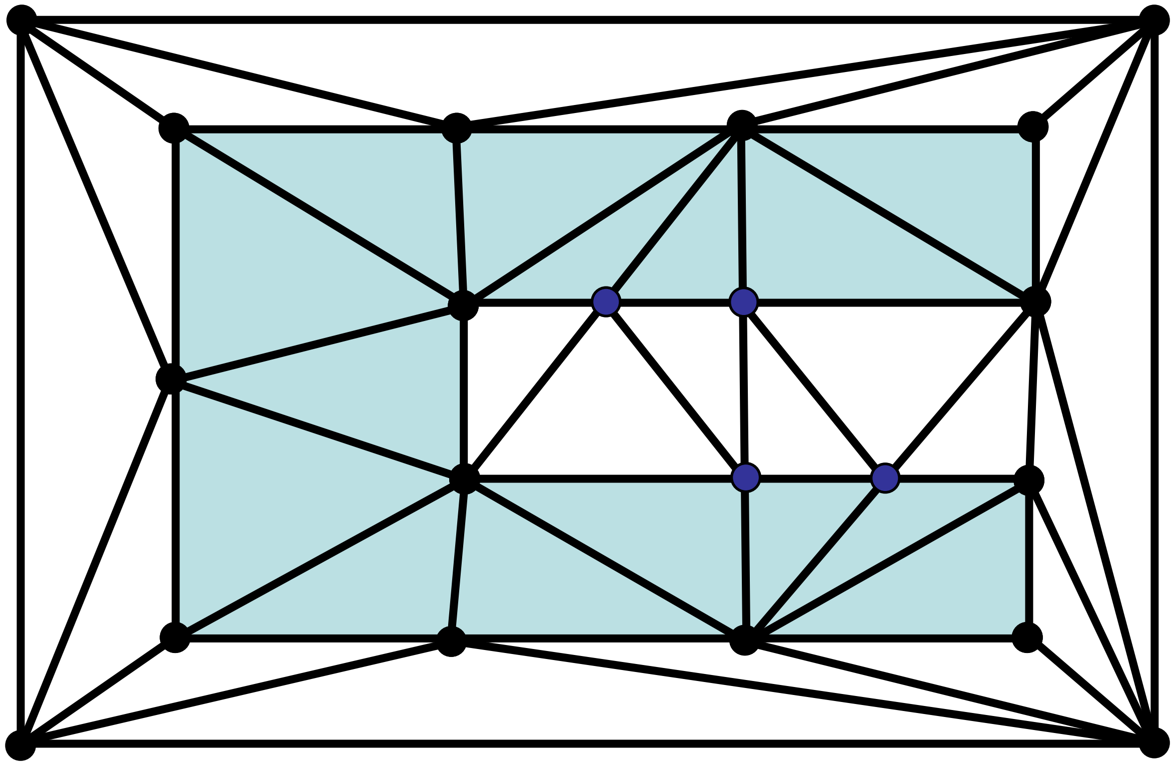


Wstawianie węzłów na brzegach istniejących trójkątów.

- węzły wstawiane są na brzegach istniejących trójkątów z założoną gęstością.
- wymagane jest sprawdzenie czy węzły na sąsiednich brzegach nie są zlokaliwoane zbyt blisko siebie.

Delaunay

Rekonstrukcja brzegu

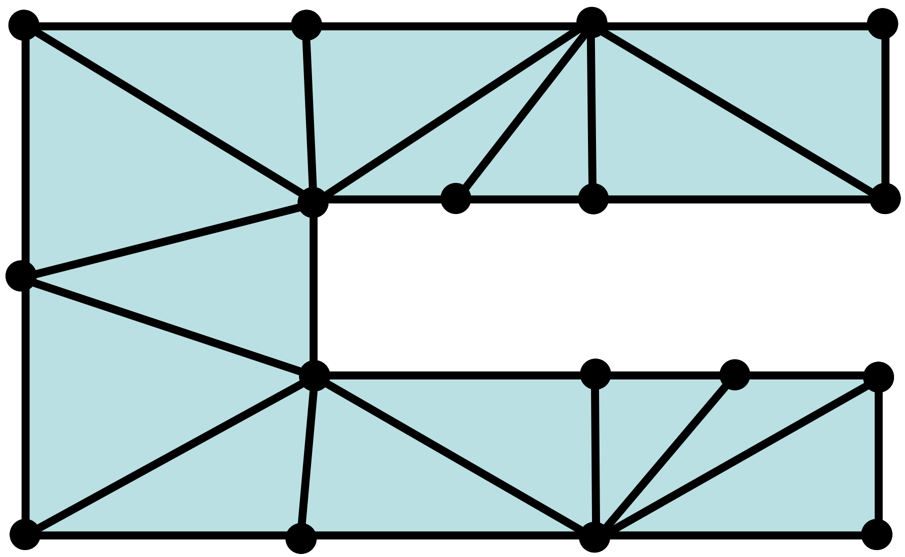


Wstawianie węzłów w punktach przecięcia trójkątów z brzegiem domeny obliczeniowej

- węzły wstawiane są w punktach w których trójkąty przecinają domenę obliczeniową.
- należy usunąć elementy leżące poza domeną obliczeniową.

Delaunay

Rekonstrukcja brzegu

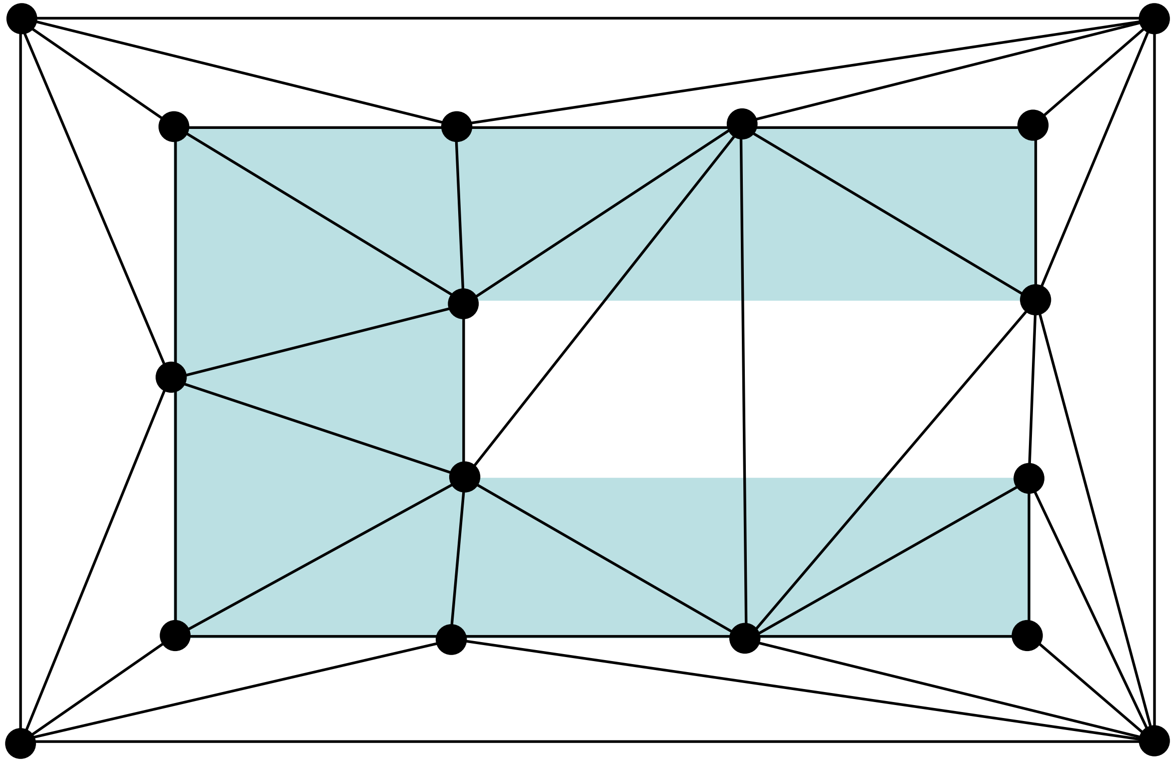


Wstawianie węzłów w punktach przecięcia trójkątów z brzegiem domeny obliczeniowej

- węzły wstawiane są w punktach w których trójkąty przecinają domenę obliczeniową.
- należy usunąć elementy leżące poza domeną obliczeniową.

Delaunay

Rekonstrukcja brzegu

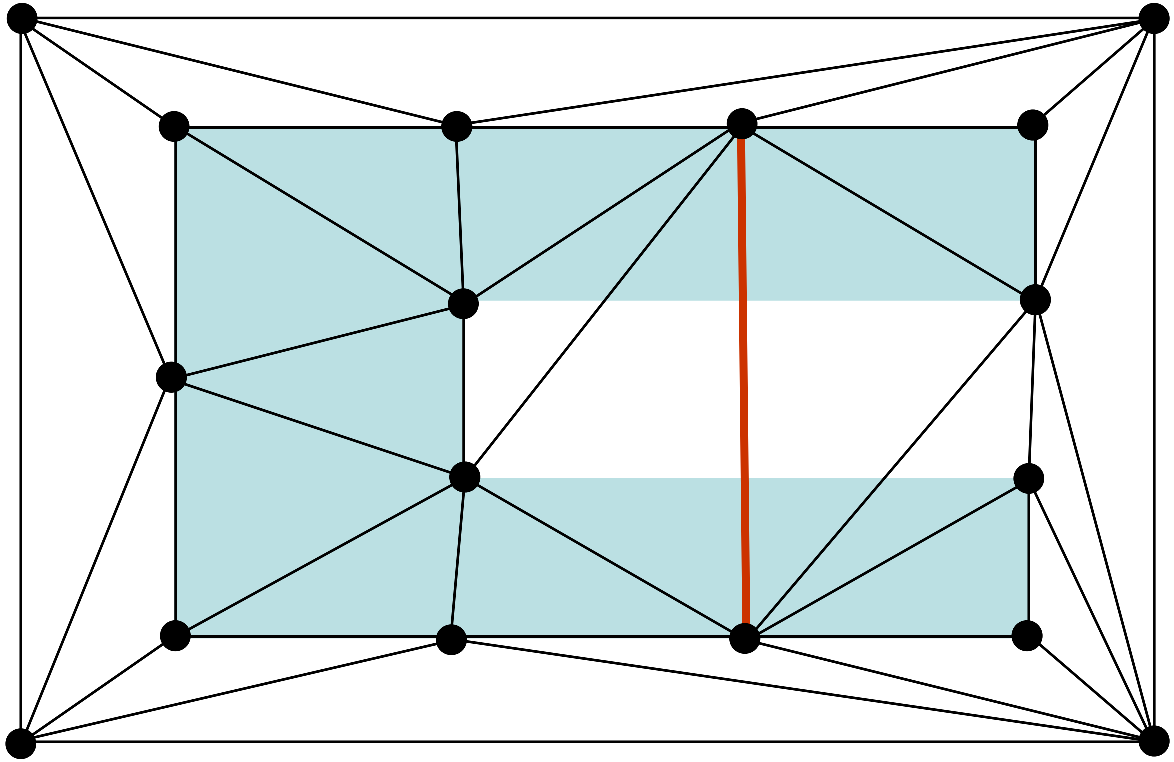


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

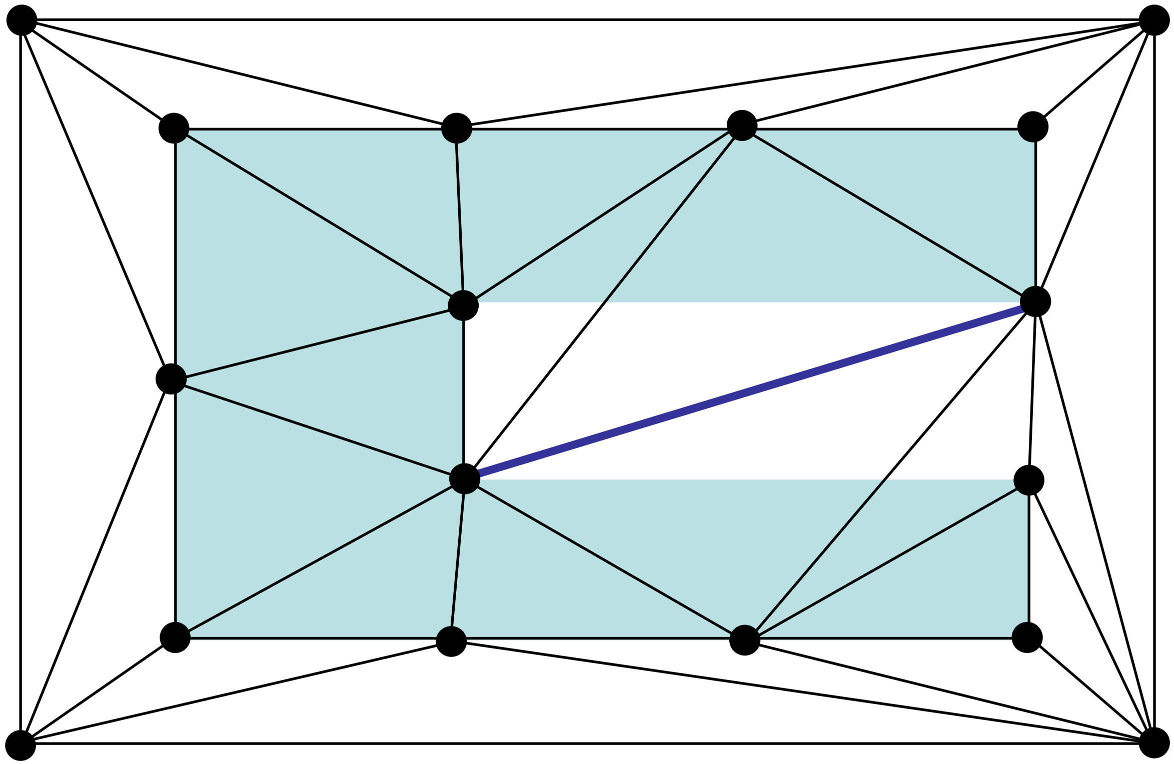


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

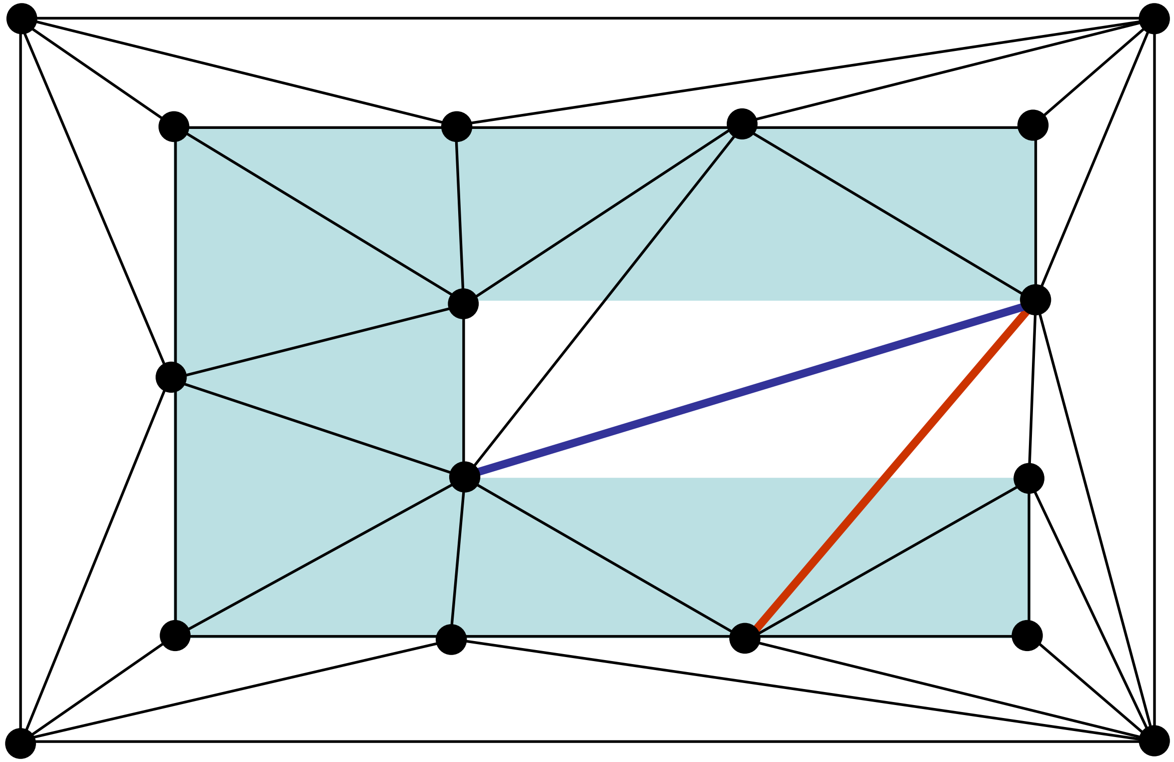


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

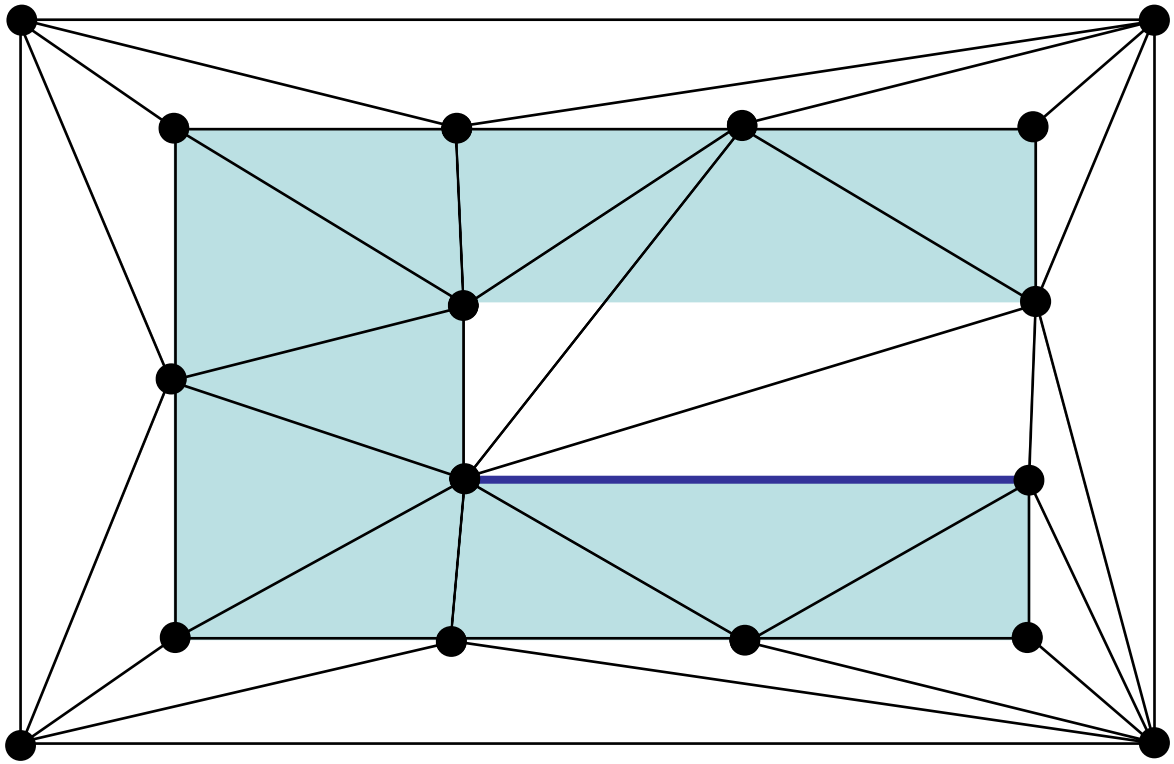


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

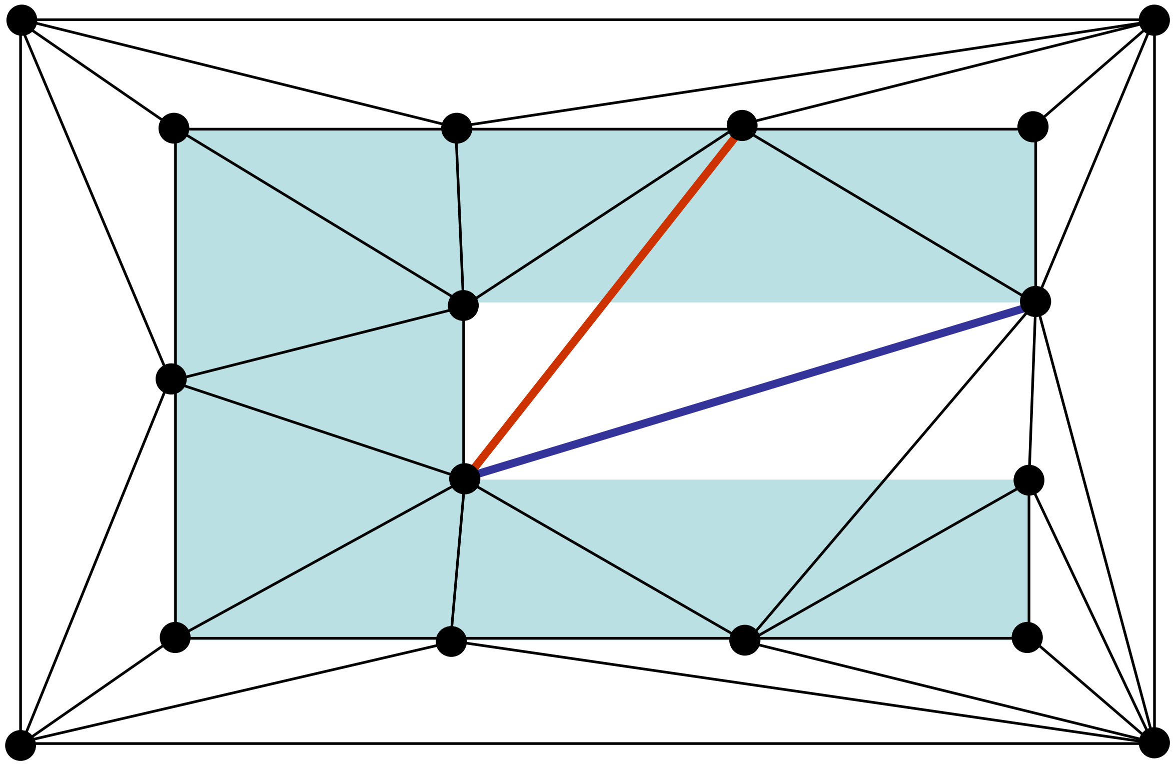


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

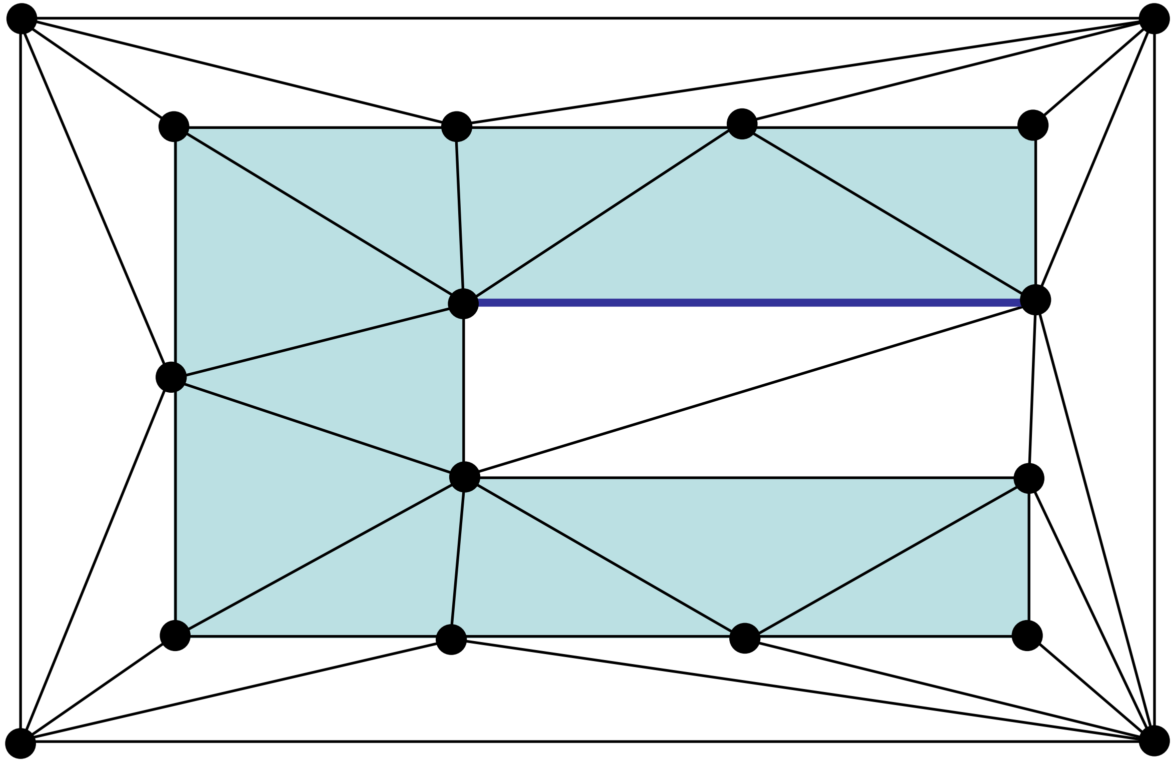


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

Delaunay

Rekonstrukcja brzegu

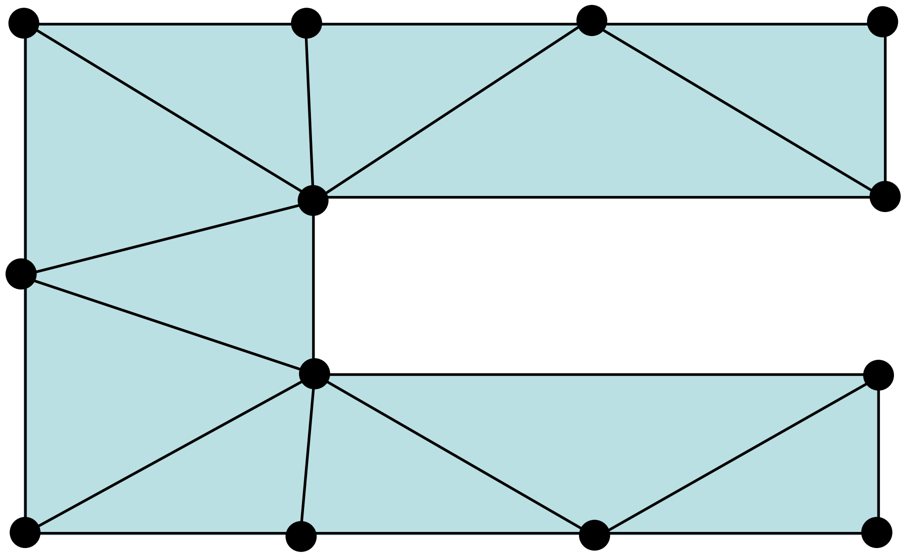


Zamiana krawędzi

• Krawędzie ulegają zamianie do momentu kiedy nie pokryją się z odcinkiem granicy domeny obliczeniowej.

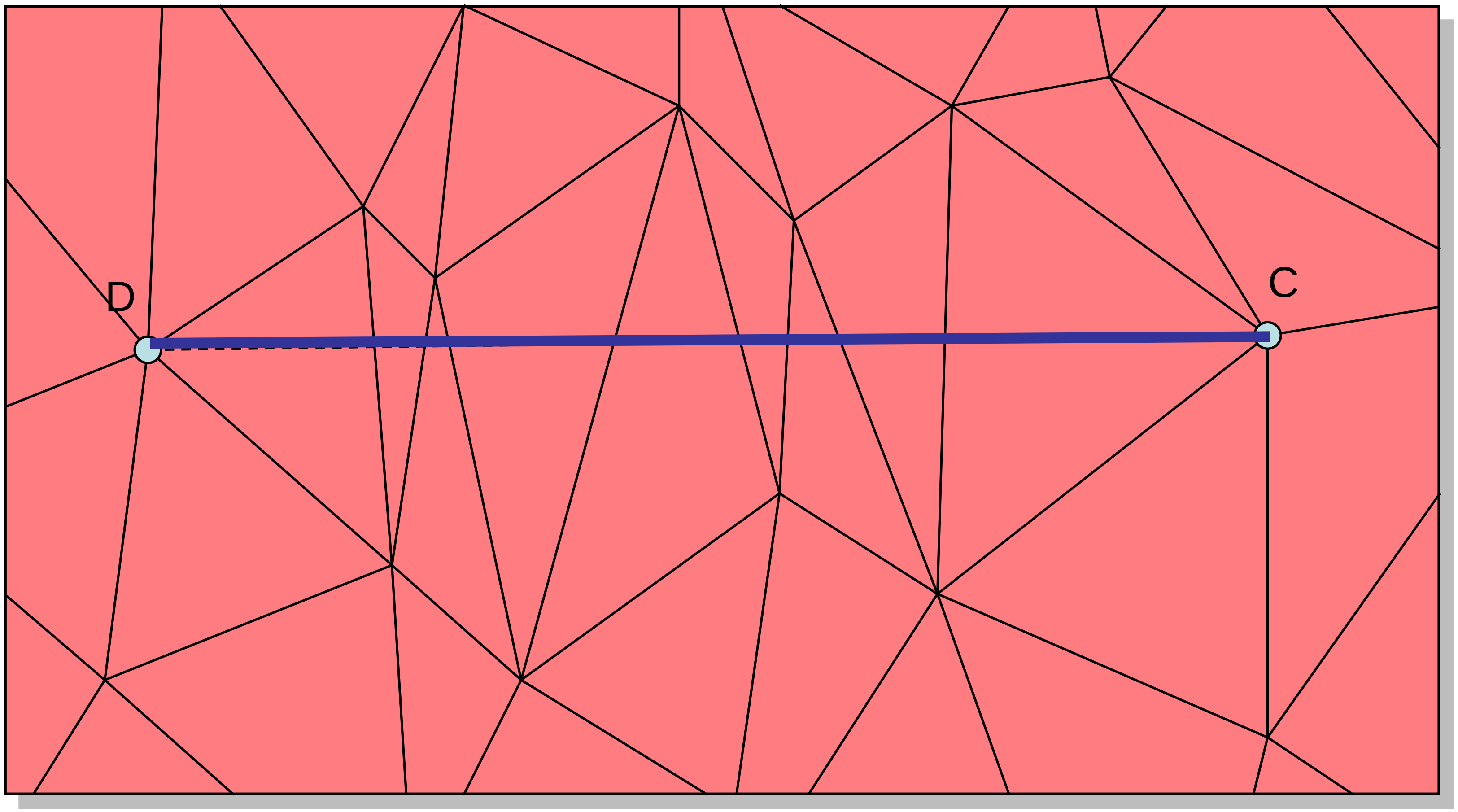
• należy usunąć elementy leżące poza domeną obliczeniową.

Rekonstrukcja brzegu



Delaunay

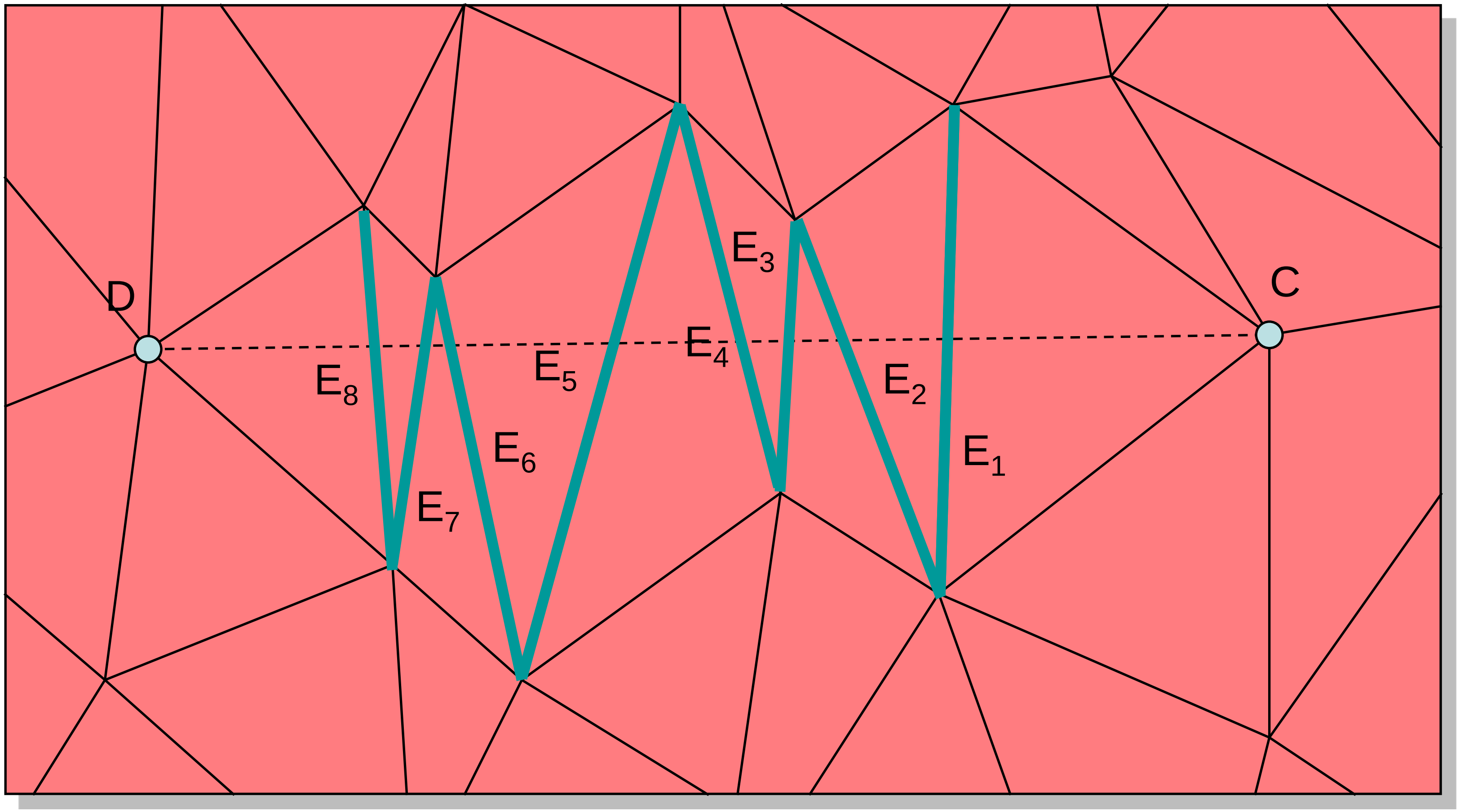
Rekonstrukcja krawędzi np. CD



Zamian krawędzi:

- Wyznacz linię łączącą analizowane punkty CD.

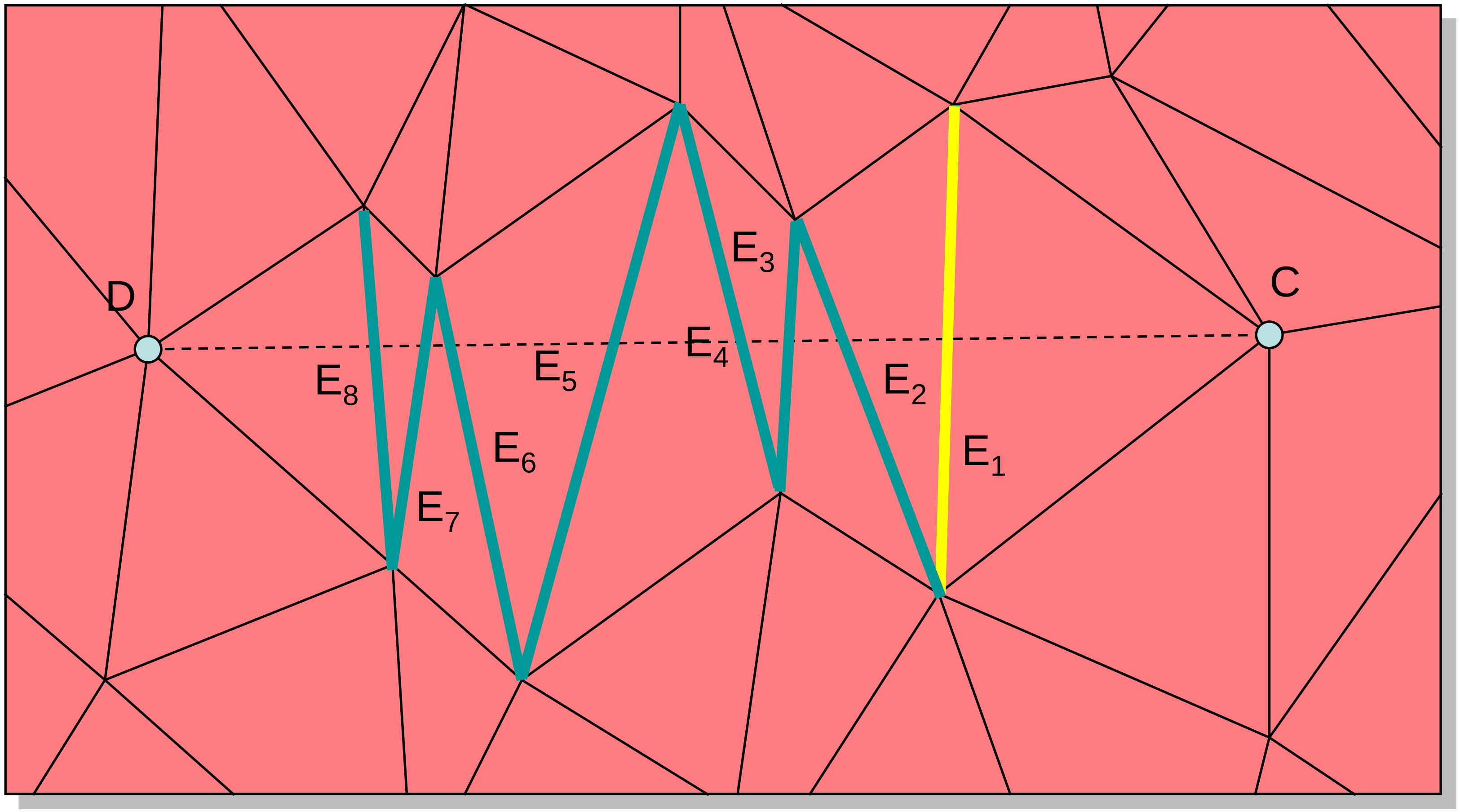
Delaunay



Zamian krawędzi:

- Wyznacz wszystkie krawędzie E_i przecinające odcinek CD.

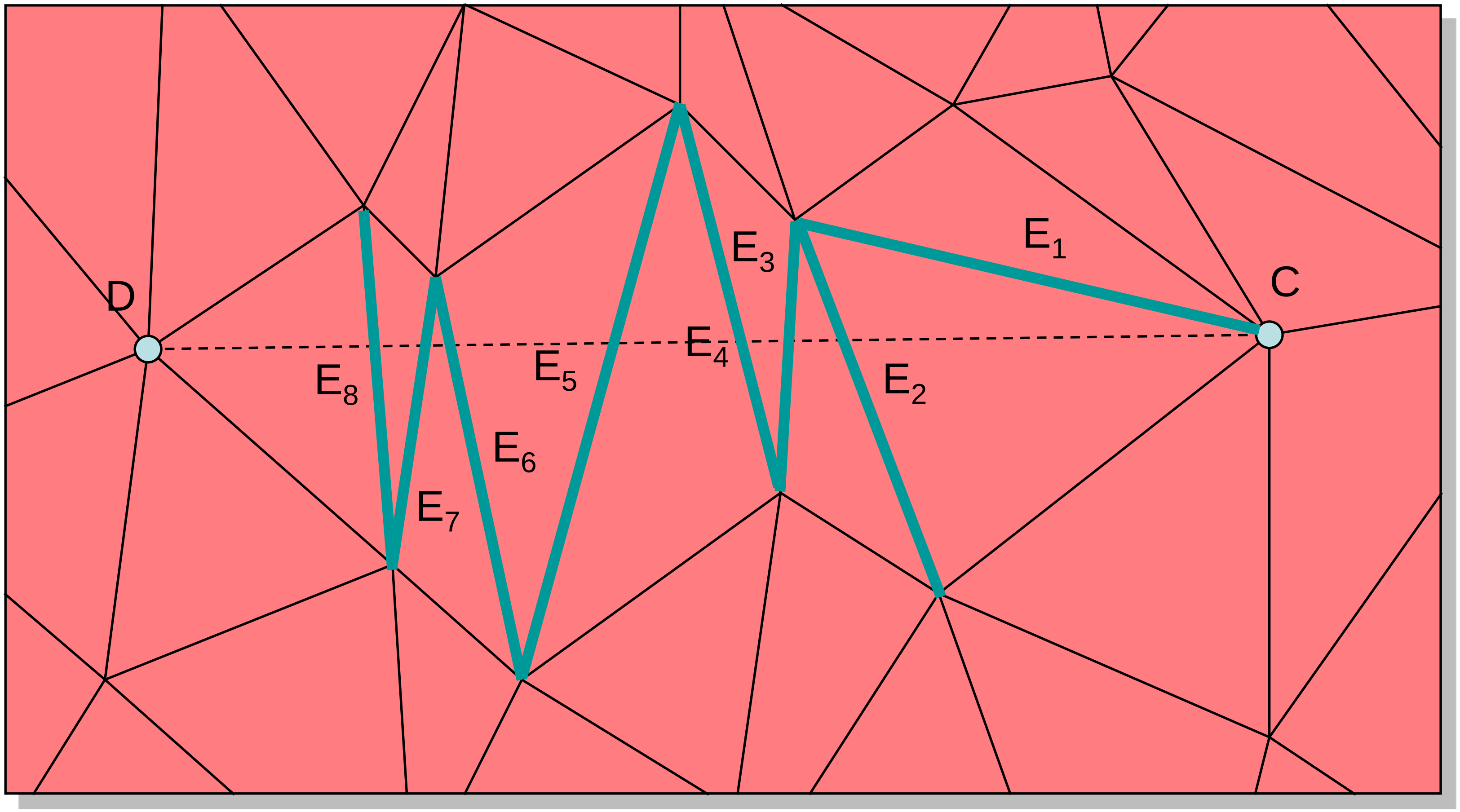
Delaunay



Zamian krawędzi:

- Zamień krawędź łączącą kolejne przylegle do siebie trójkąty.

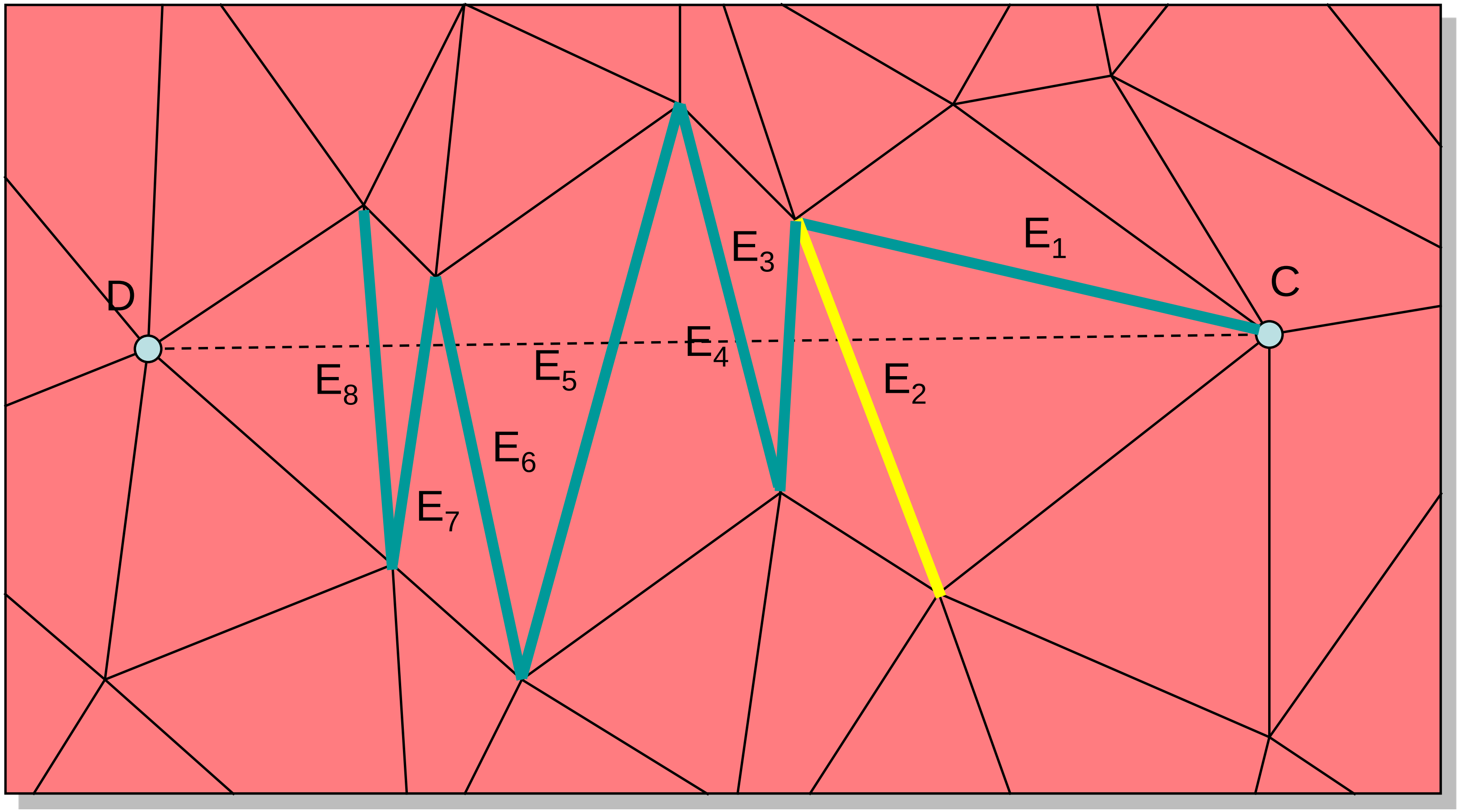
Delaunay



Zamian krawędzi:

• Zamień krawędź łączącą kolejne przylegle do siebie trójkąty.

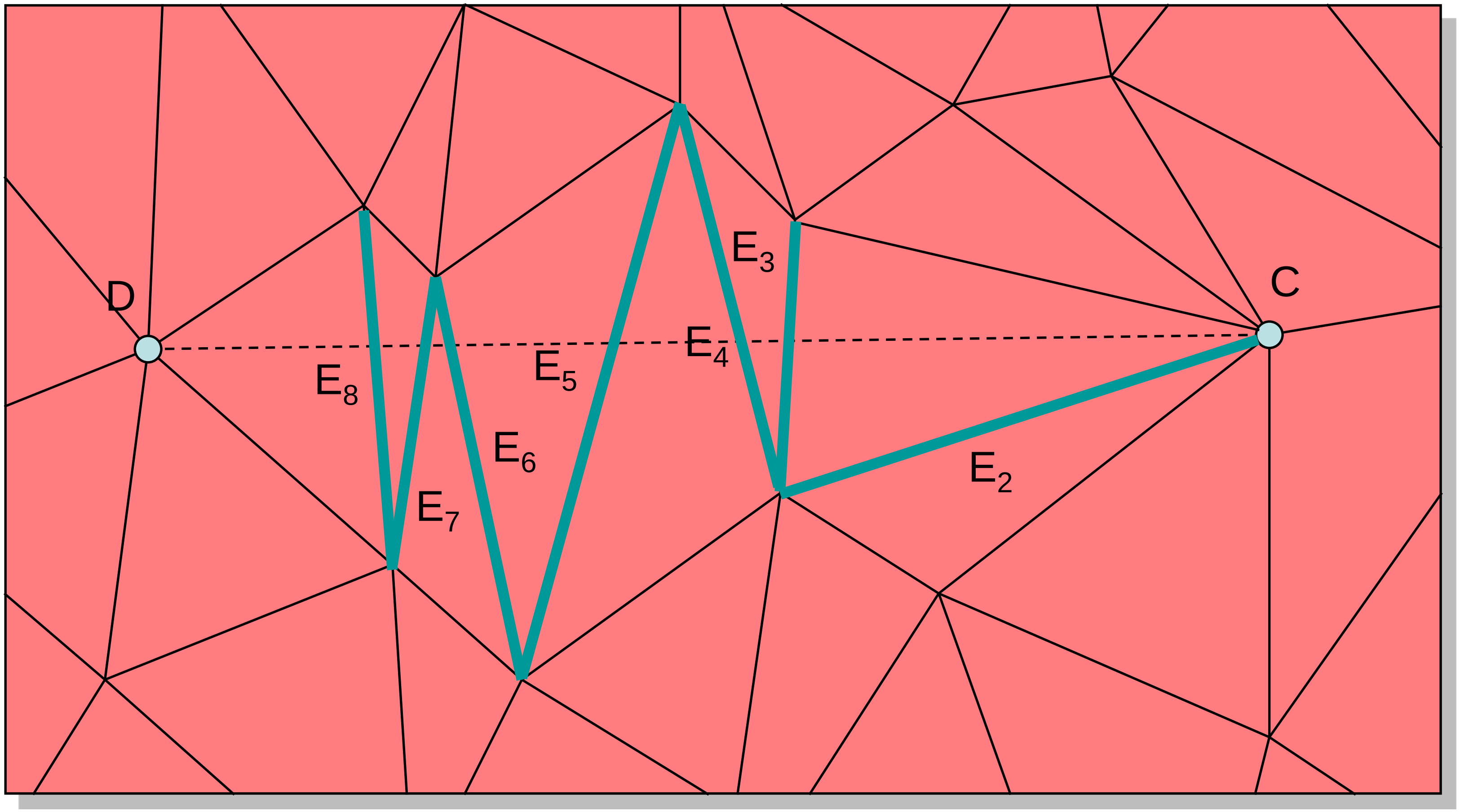
Delaunay



Zamian krawędzi:

- Zamień krawędź łączącą kolejne przylegle do siebie trójkąty.

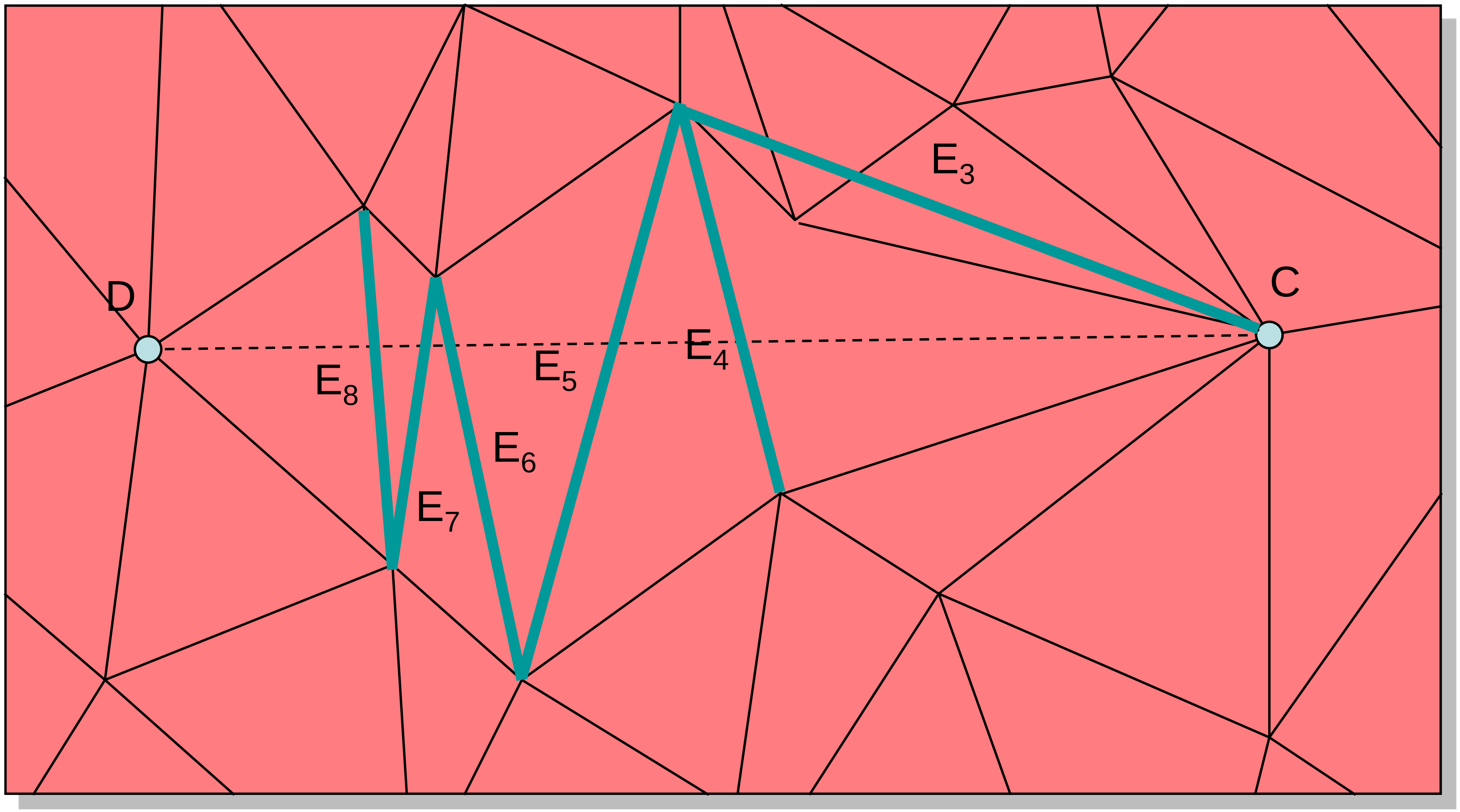
Delaunay



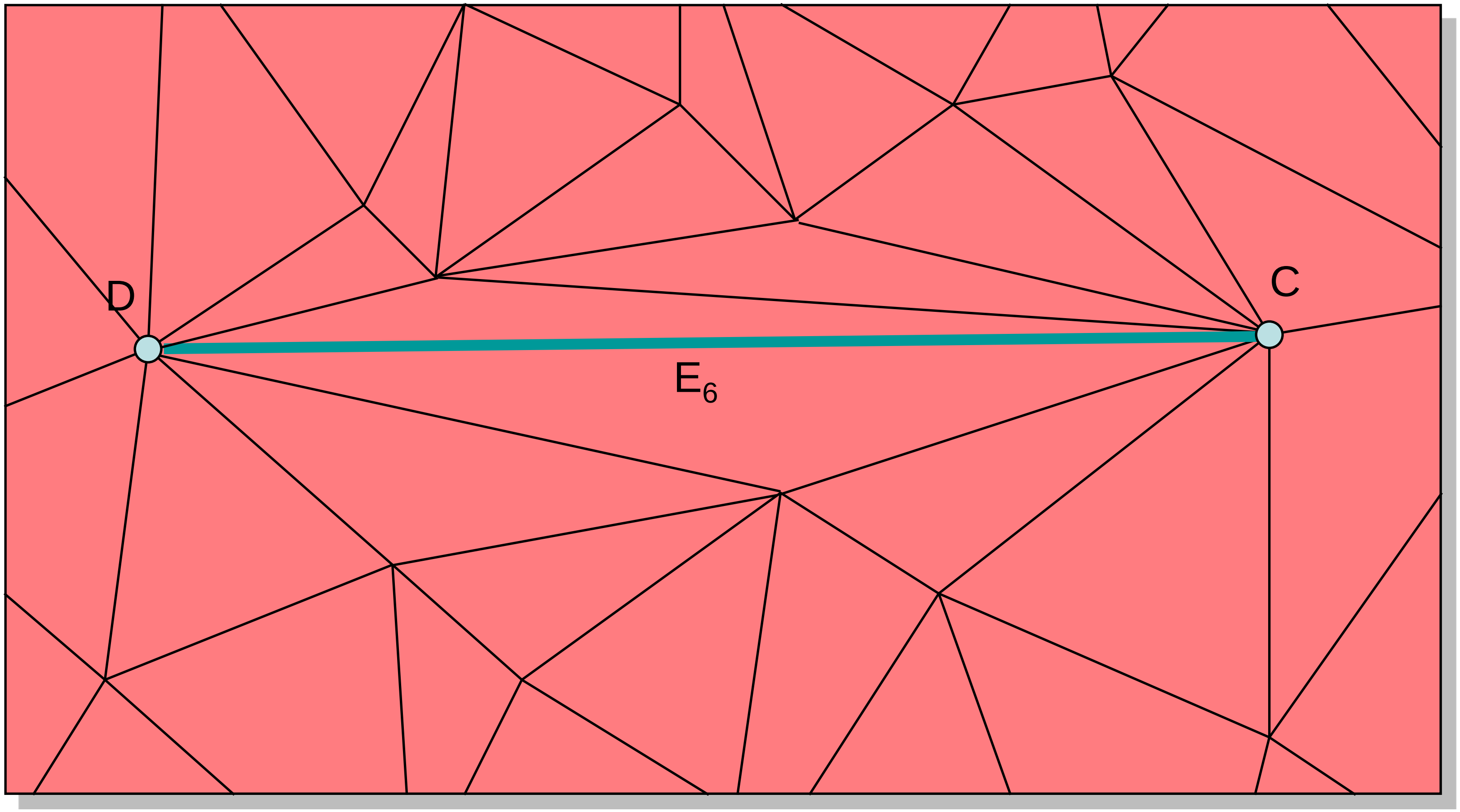
Zamian krawędzi:

• Sprawdź czy nowo utworzona krawędź nie przecina już istniejących krawędzi. Jeżeli tak pomiń tę zamianę.

Delaunay



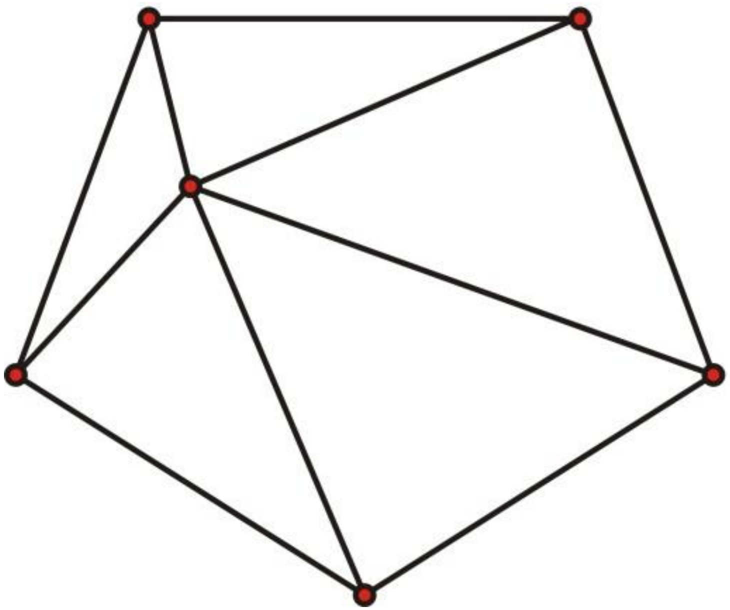
Delaunay

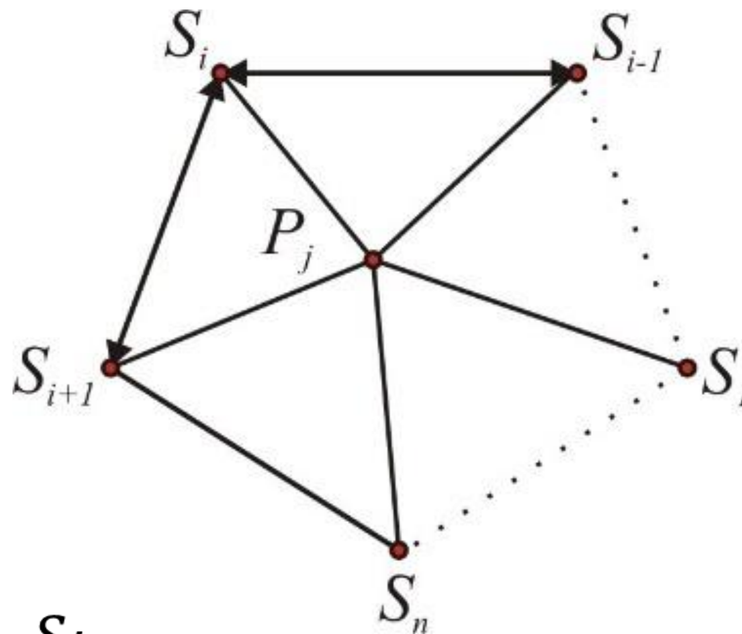


Taka siatka nie spełnia kryterium Delaunay

Wygładzanie Laplace’a

Wygładzanie Laplace’a polega na przesuwaniu węzłów do środka ciężkości figury utworzonej z wszystkich trójkątów do których ten węzeł należy.



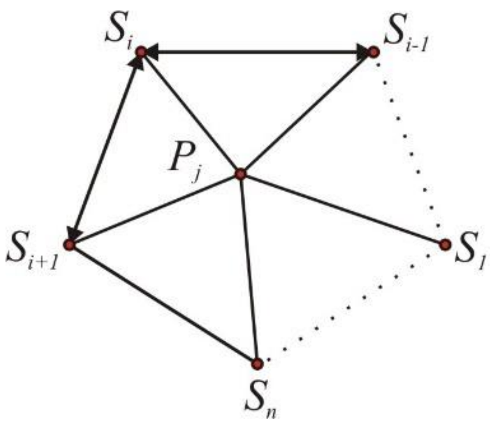


$$P_j = \frac{\sum_{i=1}^{n} S_i}{n}$$

W przypadku wygładzania elementów zmodyfikowaną metodą Laplace, każdy z punktów S otrzymuje wagi proporcjonalnie do odległości pomiędzy wspólnymi punktami, punktu P dla którego obliczane jest położenie a punktu dla którego obliczana jest waga.

Każda waga dla każdego punktu obliczana jest osobno.

Następnie wszystkie wagi punktów dzielone są przez średnią arytmetyczną wag. Aby zwiększać lub zmniejszać wpływ wagi na obliczenie pozycji punktu dodatkowo można wprowadzić dwa współczynniki v1 i v2.



$$P_j = \frac{\sum_{i=1}^{n} \left(S_i w_i + S_i v_2 \right)}{n \left(v_2 + 1 \right)} \qquad \forall j \in \{1, \ldots, N\}$$

$$w_i = \left(\left| S_i S_{i+1} \right| + \left| S_i S_{i-1} \right| \right)^{v_1}$$

http://www.cs.cmu.edu/~quake/triangle.html