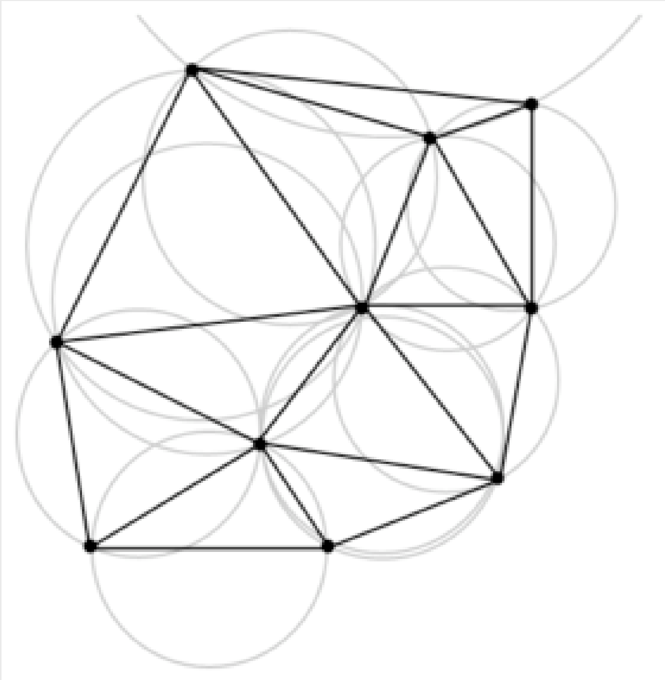
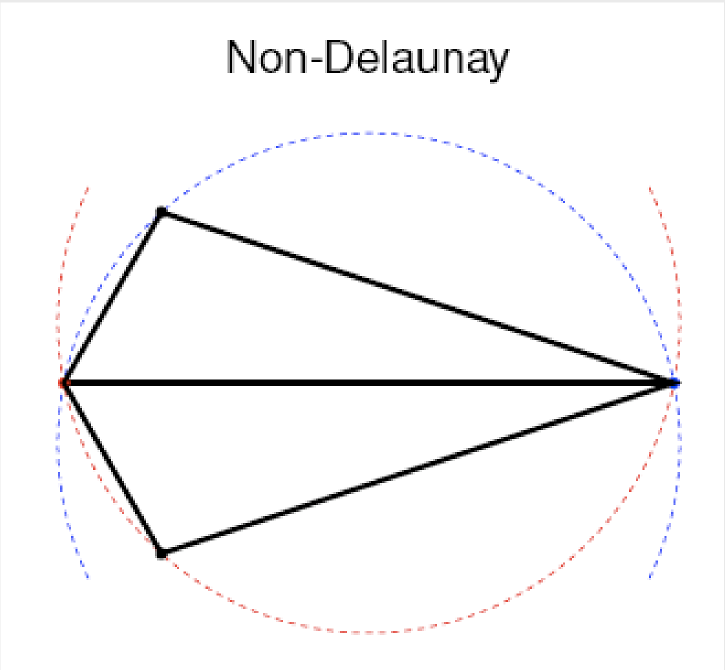
Algoritmi per la triangolazione di Delaunay

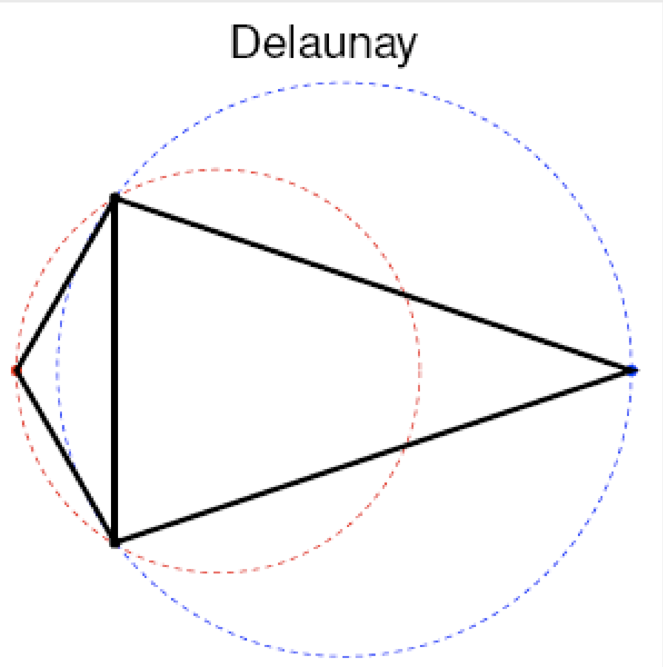
Definizione

* Dato un insieme di punti P, la triangolazione è univocamente determinata

* Per ogni circonferenza circoscritta ad un triangolo, nessun punto di P (oltre a quelli che formano il triangolo stesso) giace all'interno della circonferenza







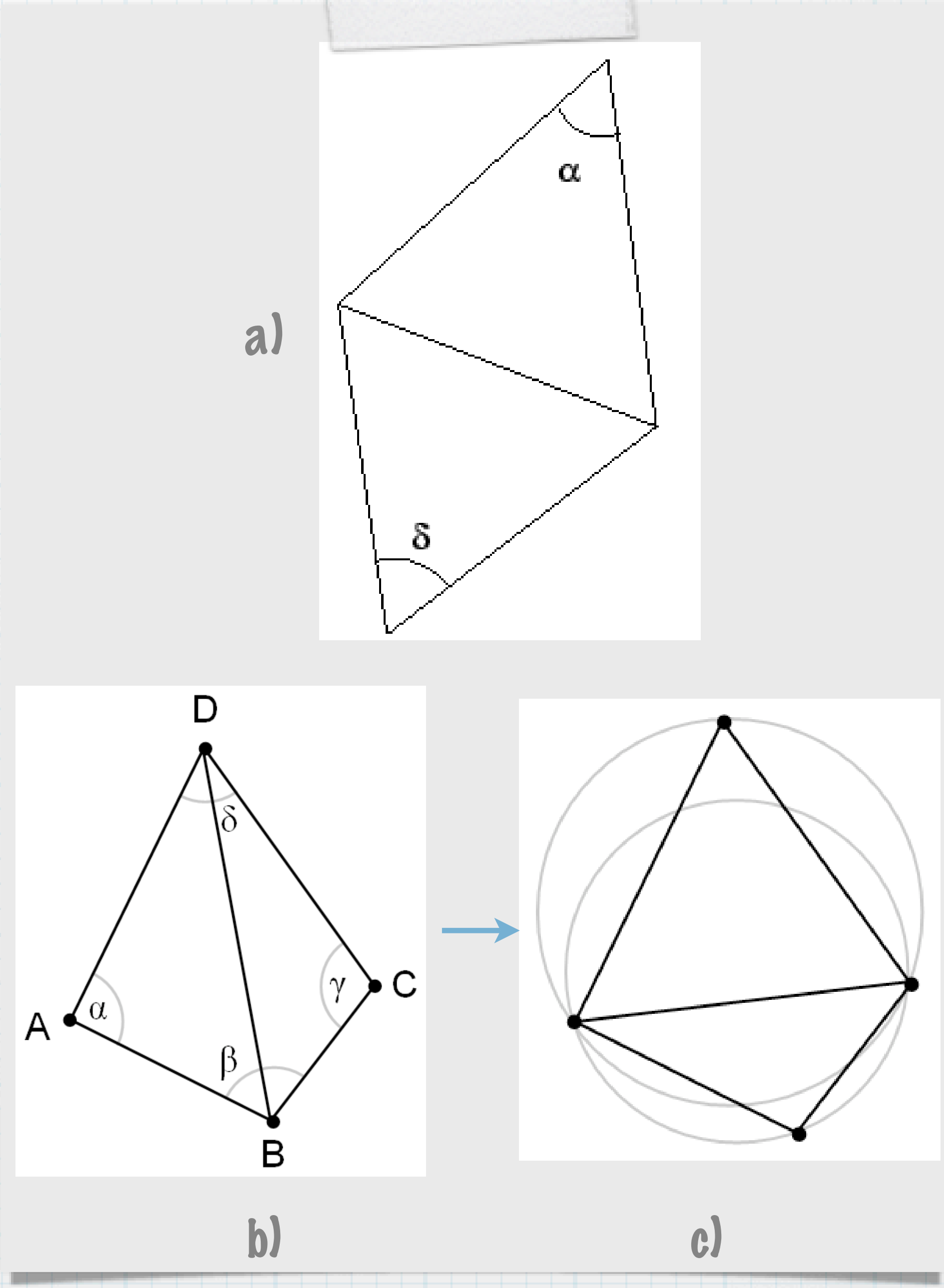
- E' possibile estendere il discordo al caso tridimensionale attraverso l'uso di sfere anzichè circonferenze (e più in generale al caso con n dim)

Proprietà

* La triangolazione di D. massimizza l'angolo minimo (tra tutte le triangolazioni è quella che produce triangoli con angoli più piccoli)

* Nei triangoli che hanno un lato in comune la somma degli angoli opposti a tale lato è minore di 180°

* Questo fa si che se due triangoli (con lato in comune) non garantiscono la proprietà di Delaunay (fig. b) , sostituendo il lato in comune BD con AC avremo due triangoli che soddisfano le proprietà (l'arco viene "flippato")



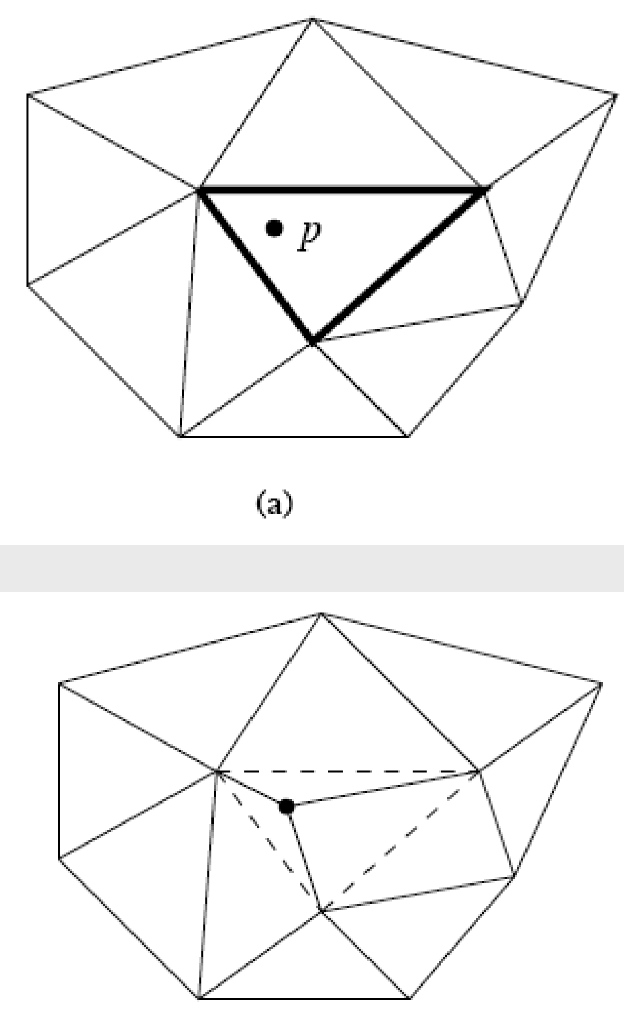
Algoritmi

* Incrementale in O(N^2)
  * Incremental Delaunay Triangulation - Dani Lischinski
    graphics.stanford.edu/courses/cs468-02-fall/readings/lischinski.ps
* Dividi et impera in O(NlogN)
  * DeWall: A Fast Divide and Conquer D. T. Algorithm
    vcg.isti.cnr.it/publications/papers/dewall.pdf
* Convex Hull in O(NlogN)

Incrementale

* I punti vengono aggiunti uno ad uno ed ad ogni passo le proprietà della triangolazione di D. vengono mantenute
* Si inizia costruendo un triangolo abbastanza grande da contenere tutti i punti
* I punti dell'insieme P vengono aggiunti uno alla volta:

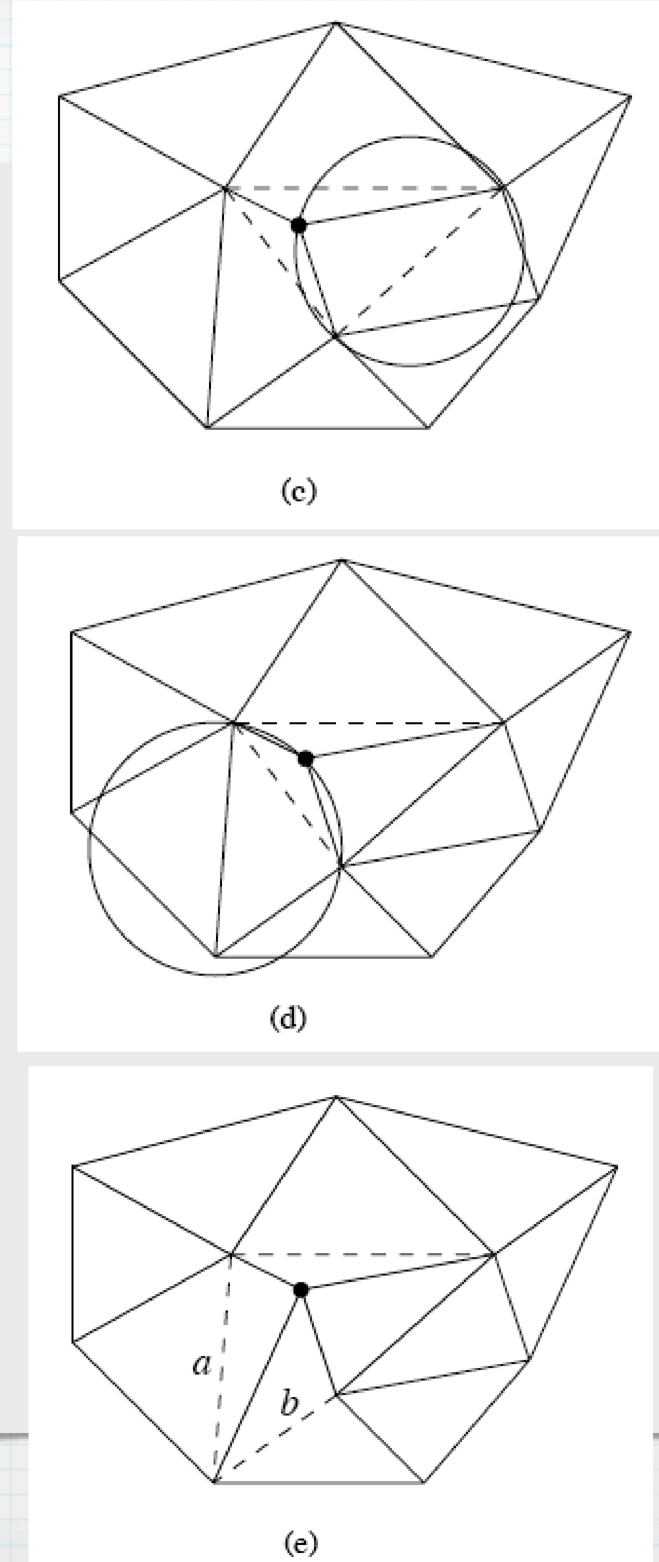
* Il nuovo punto p viene collocato all'interno della struttura (a)
* Nuovi archi vengono creati per collegare p ai vertici del triangolo che lo circonda (b)



(a)

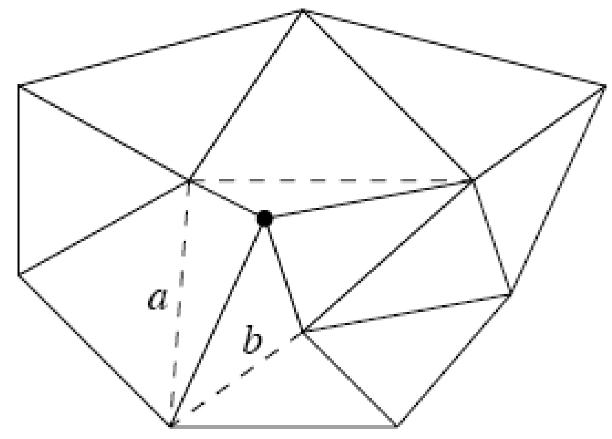
(b)

- Ogni “vecchio” arco viene controllato affinchè soddisfi ancora la proprietà della “circonferenza”
- Se mantiene la proprietà l’arco non viene toccato (c)
- Altrimenti l’arco viene rimpiazzato dalla diagonale del quadrilatero che si forma (d-e)

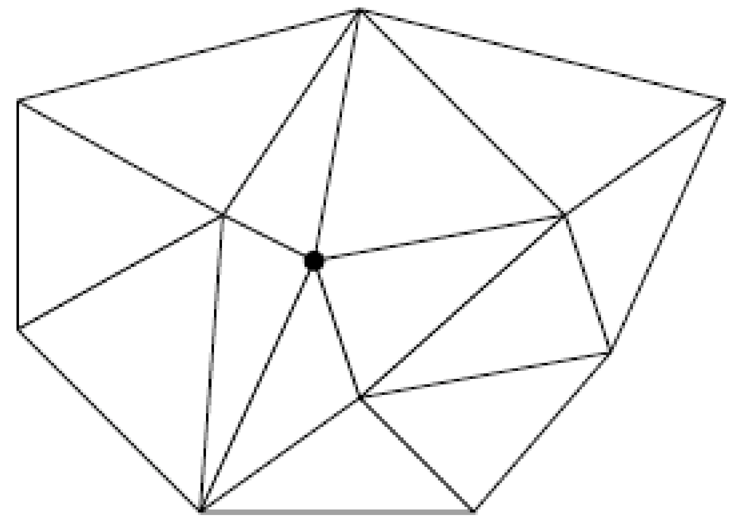


(c)
(d)
(e)

* Vengono controllati i lati dei “nuovi” triangoli che si sono formati
* Si procede con un altro punto



(e)



(f)

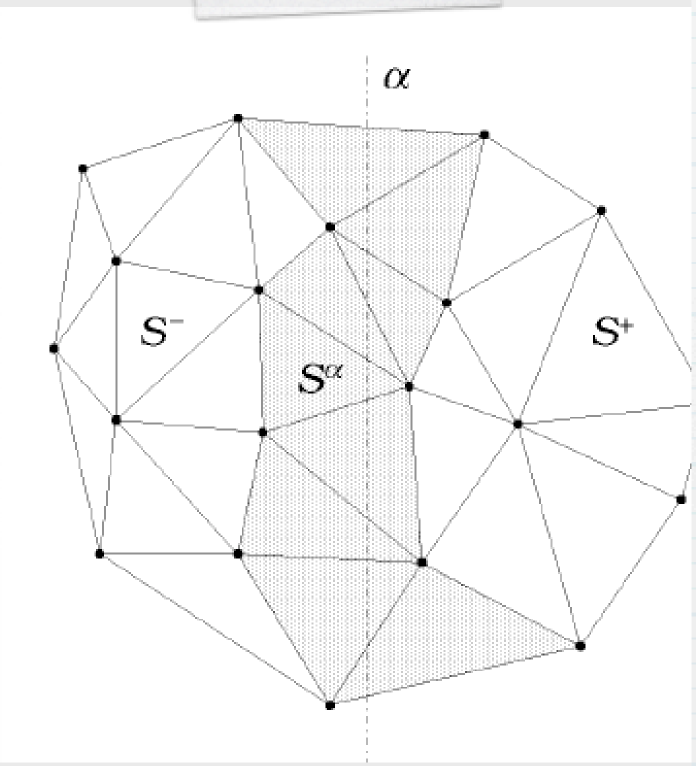
Dividi et impera

* Algoritmo DeWall
* Dato un insieme di punti permette di costruire una T. di D. in uno spazio a 'd' dimensioni

DeWall

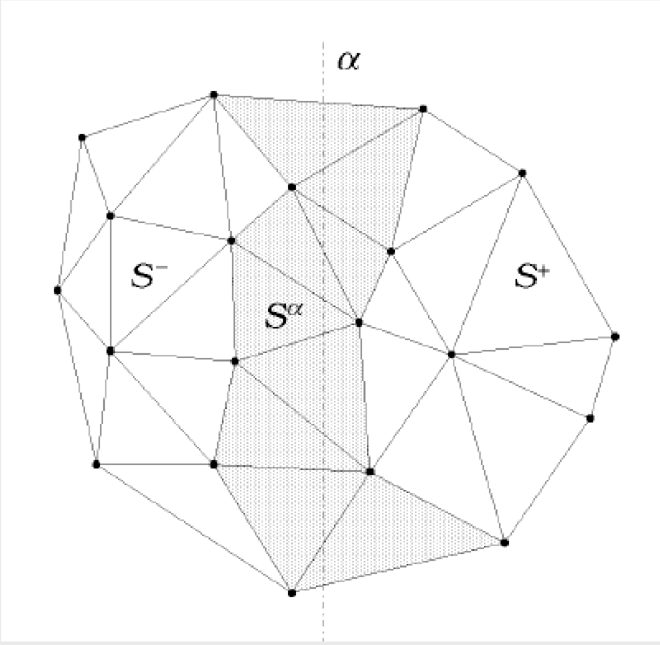
* Gli altri algoritmi (di triangolazione) "dividi et impera" separano l'insieme dei punti in due parti, costruiscono la triangolazione su ciascuna delle parti e poi unisco le due strutture

* Qui si cerca di costruire prima la parte "a cavallo" (chiamata S^{α}), attraverso una più complicata fase di divisione, e poi si costruisco le altre parti iterando il procedimento

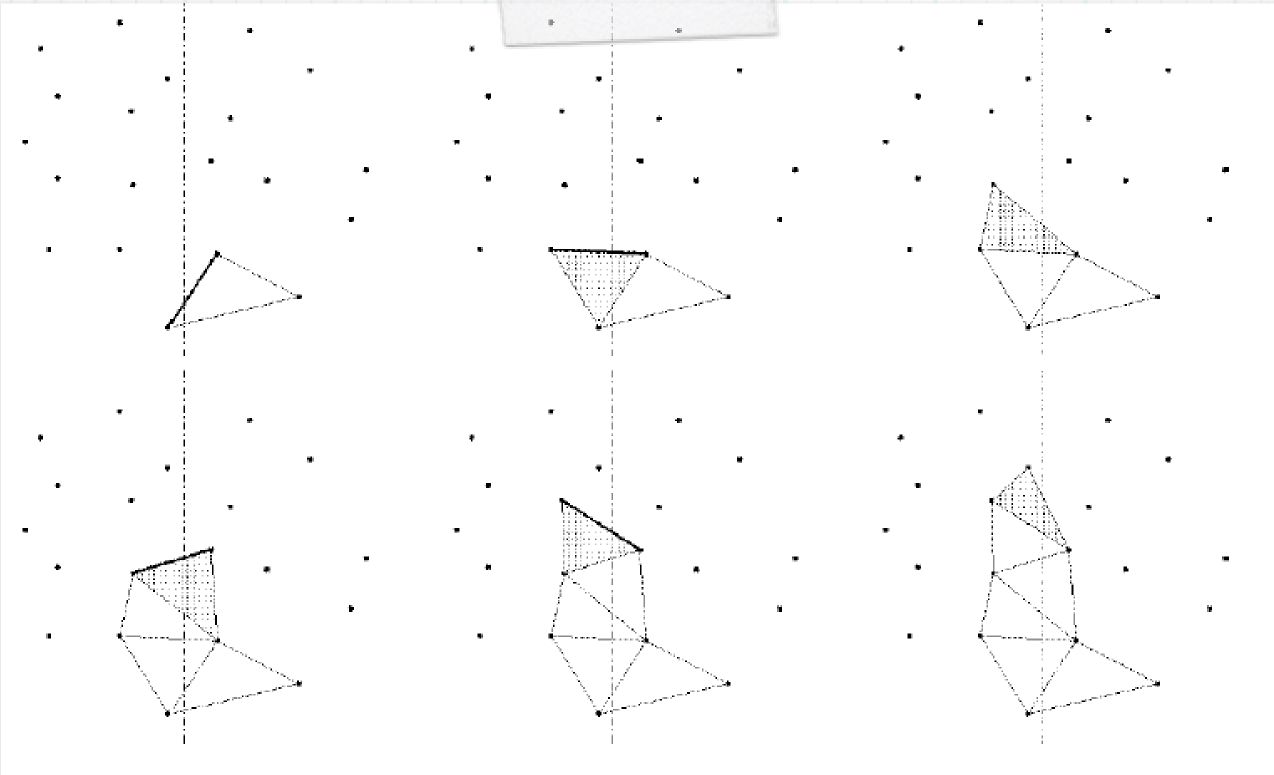


DeWall

* I tre passi dell'algoritmo sono:
  * Selezionare i piano α, dividere l'insieme dei punti in due parti, e costruire S^α
  * Applicare ricorsivamente l'algoritmo a S^+ e S^-
  * Restituire l'unione dei tre

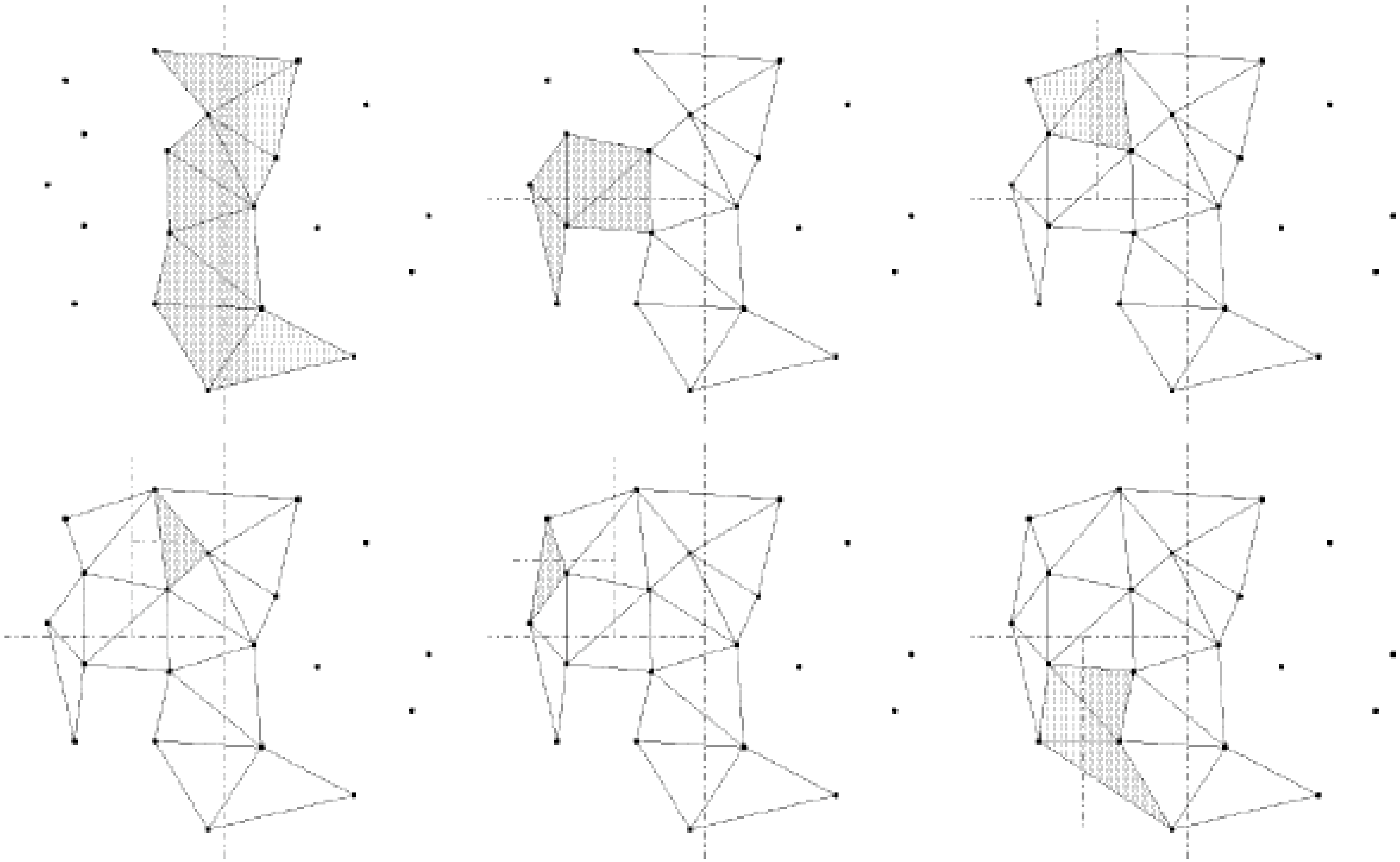


Costruzione di S^{α}



* Dopo aver stabilito il piano α, si costruisce il primo triangolo

* partendo dalla faccia (lato in 2d) che interseca α si costruisce il triangolo successivo scegliendo il punto, dall'altra parte di α, che minimizza il raggio della circonferenza circoscritta



Link

* Applet algoritmi:
  * http://www.cse.unsw.edu.au/~lambert/java/3d/delaunay.html
* Test di concavità:
  * http://local.wasp.uwa.edu.au/~pbourke/geometry/clockwise/