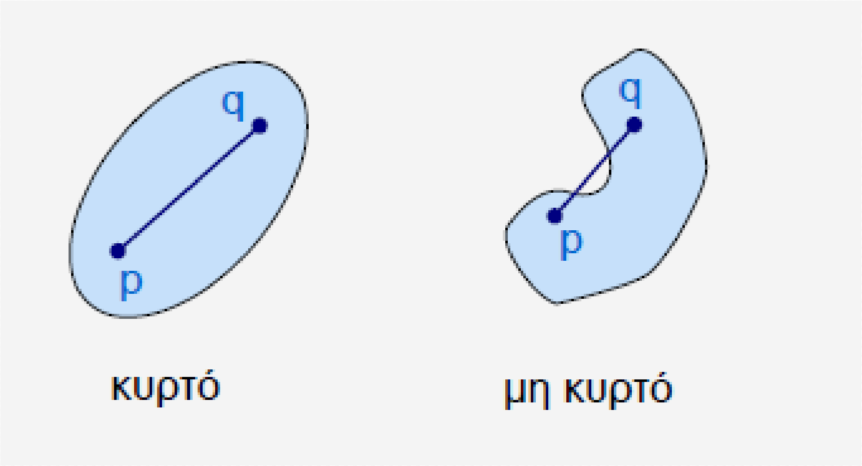
Πανεπιστήμιο Πελοποννήσου

Μεταπτυχιακή Εργασία

Μέθοδοι κατασκευής κυρτών περιβλημάτων και εφαρμογές

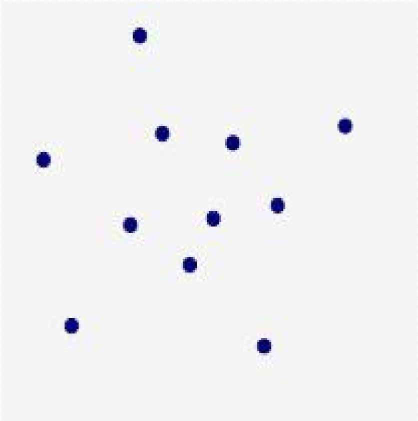
Κυρτό Σύνολο

- Ορισμός: Ένα σύνολο σημείων S στο επίπεδο καλείται κυρτό εάν για οποιαδήποτε δυο σημεία p, q που ανήκουν στο S, το ευθύγραμμο τμήμα pq ανήκει στο S.



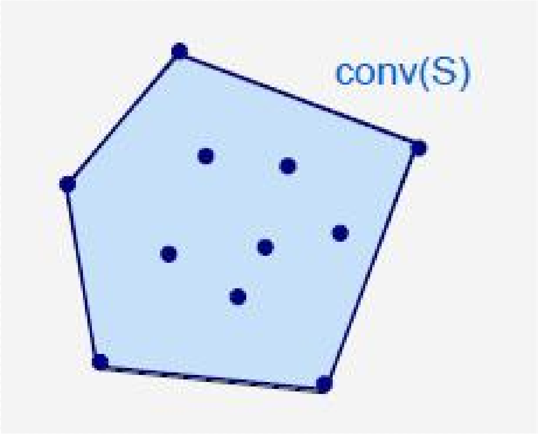
Κυρτό Περίβλημα

- Ορισμός: Το κυρτό περίβλημα ενός συνόλου σημείων S είναι το μικρότερο κυρτό σύνολο που περιέχει το S.
- Συμβολίζεται με conv(S).



Κυρτό Περίβλημα

- Ορισμός: Το κυρτό περίβλημα ενός συνόλου σημείων S είναι το μικρότερο κυρτό σύνολο που περιέχει το S.
- Συμβολίζεται με conv(S).



Αλγόριθμοι κατασκευής κυρτών περιβλημάτων σε 2D

- Αλγόριθμος του Graham
- Αλγόριθμος του Jarvis
- Αλγόριθμος του Chan

Αλγόριθμος του Graham

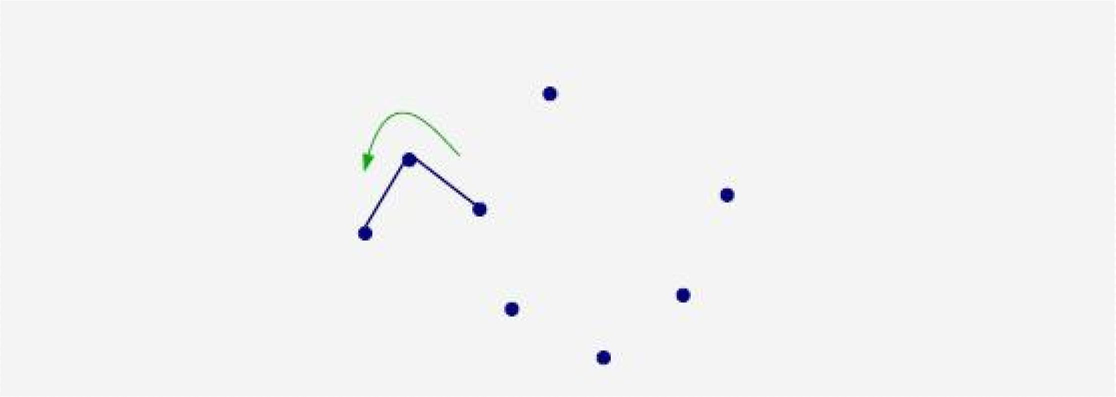
- Χρησιμοποιούμε μια στοίβα S με τις γνωστές πράξεις προσθήκης (push) και αφαίρεσης (pop)

- Έχουμε την δυνατότητα να προσπελάσουμε το πρώτο και το δεύτερο σημείο στην κορυφή της στοίβας S με τις πράξεις S. first και S. second

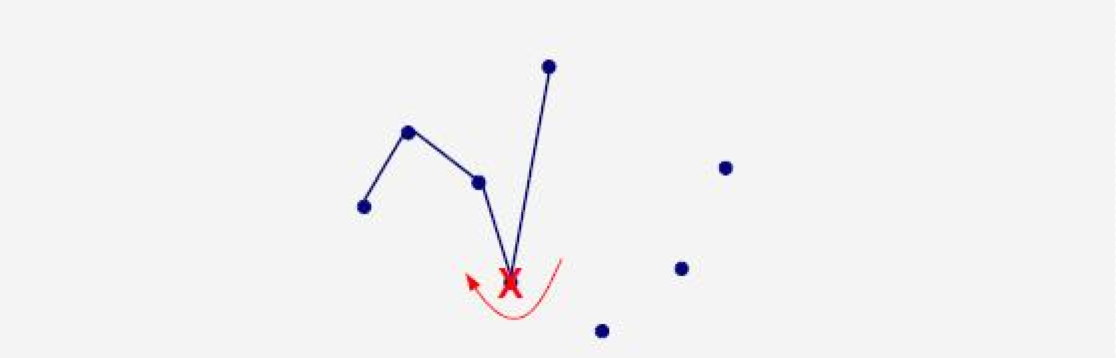
Αλγόριθμος του Graham

- Ταξινόμησε τα σημεία του P κατά αύξουσα x-συντεταγμένη (P1,P2, …, Pn)
- Πρόσθεσε το P1 και μετά το P2 στη στοίβα S.
- Για i= 3 έως n
- α. Ενόσω (μέγεθος (S) > 2 και Orient (pi, S.firts, S.second) < 0
  - Αφαίρεσε ένα σημείο από την στοίβα
- β. Πρόσθεσε το P1 στην S.

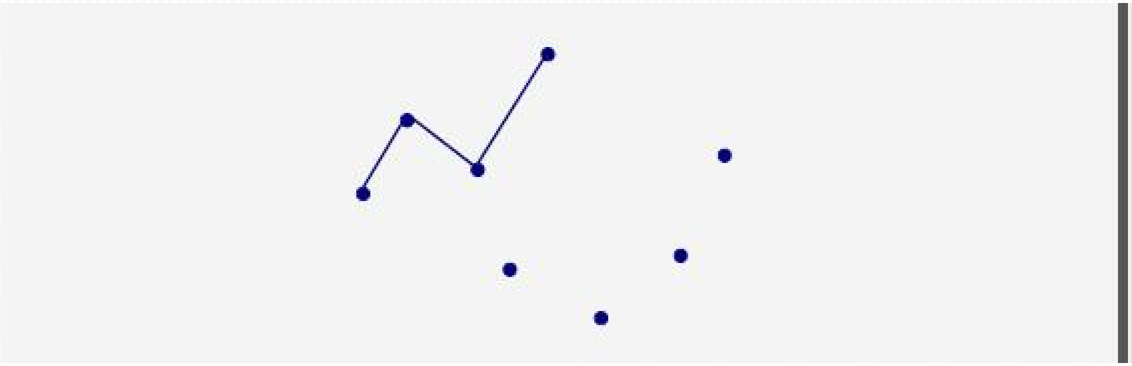
Αλγόριθμος του Graham



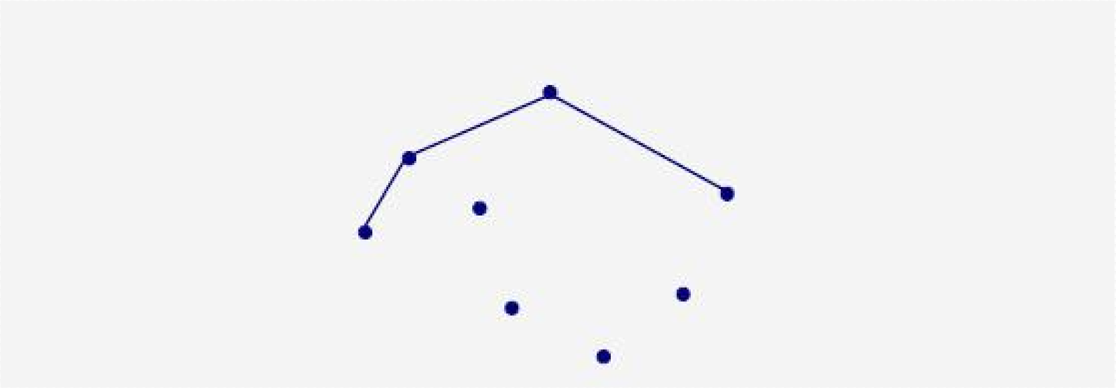
Αλγόριθμος του Graham



Αλγόριθμος του Graham



Αλγόριθμος του Graham



Αλγόριθμος του Graham

- Ο αλγόριθμος έχει χρόνο εκτέλεσης O (n log n)
- Η ταξινόμηση κατά x-συντεταγμένη μπορεί να γίνει σε χρόνο O (n log n)
- Πόσες φορές εκτελείται το βήμα 1 α ??
- Κάθε σημείο μπορεί να διαγραφεί μια φορά και υπάρχουν (n) σημεία, οπότε εκτελείται το πολύ O (n) φορές
- Η συνθήκη του βρόχου δεν μπορεί να είναι ψευδής πάνω από O(n) φορές
- Συνολικά για όλα τα (i) δαπανάται χρόνος O(n) + O(n) = O(n)

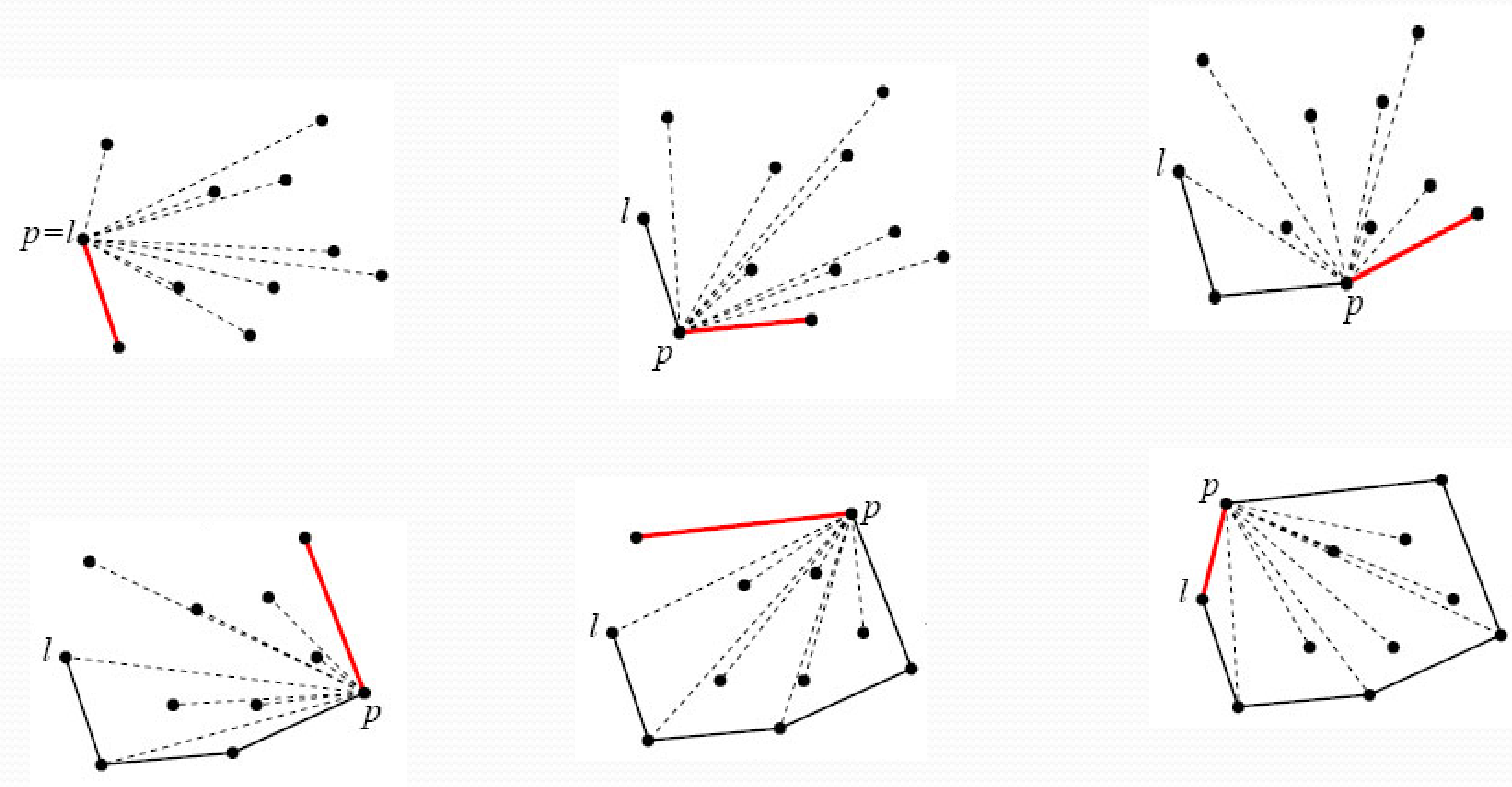
Αλγόριθμος του Jarvis

- Χρονική Πολυπλοκότητα O (nh)
  - n = αριθμός σημείων
  - h = αριθμών επαναλήψεων

Αλγόριθμος του Jarvis

- Ξεκινά με το σημείο που έχει την μικρότερη y-συντεταγμένη.
- Το ονομάζει P1 και συνεχίζει για την δεύτερη κορυφή φτιάχνοντας την πρώτη ακμή.
- Επισκέπτεται όλες τις κορυφές με αντίθετη φορά των δεικτών του ρολογιού μέχρι να ξαναβρεθεί στην αρχή.
- P1 αρχικό σημείο
- P2 μεγιστοποιεί την εσωτερική γωνία με κορυφή το P1
- P3 μεγιστοποιεί την εσωτερική γωνία με κορυφή το P2

Αλγόριθμος του Jarvis



Μπορούμε καλύτερα από αυτό ??

- Graham O(n log n) Jarvis O (nh)

- Chan O (n log h)

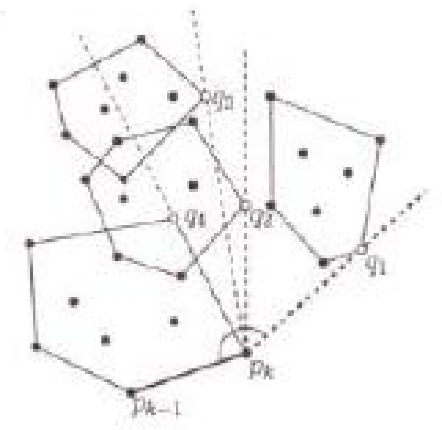
Αλγόριθμος του Chan

- Βέλτιστος εξόδου αλγόριθμος που υπολογίζει το κυρτό περίβλημα στις δυο διαστάσεις
- Αλγόριθμος που η χρονική πολυπλοκότητα του εξαρτάται από το μέγεθος του συνόλου των σημείων και από το μέγεθος των σημείων που συμμετέχουν στην κατασκευή του περιβλήματος
- Συνδυάζει τον αλγόριθμο του Graham και τον αλγόριθμο του Jarvis για να πετύχει χρονική πολυπλοκότητα $O(n \log h)$

Αλγόριθμος του Chan

- ΣΤΑΔΙΑ ΑΛΓΟΡΙΘΜΟΥ
- Α. Διαιρεί το σύνολο των σημείων P σε μικρότερα υποσύνολα
- Β. Κατασκευάζει το κυρτό περίβλημα για κάθε υποσύνολο
- Γ. Συνδυάζει τα κυρτά περιβλήματα κατασκευάζοντας το συνολικό κυρτό περίβλημα

Αλγόριθμος του Chan



Ο αλγόριθμος κατασκευής κυρτού περιβλήματος του Chan

Αλγόριθμος του Chan

- Α. διαιρεί τα σημεία σε ισομεγέθη ομάδες
  - Το σύνολο των ομάδων ισούνται με $r = n/m$
  - r = ακέραιο αριθμός
  - n = αριθμός σημείων
  - m = μέγεθος ομάδας

Αλγόριθμος του Chan

- Η παράμετρος m μας δείχνει μια εκτιμώμενη τιμή για το μέγεθος της κάθε ομάδας, δηλαδή κάθε ομάδα θα έχει το πολύ m σημεία.
- Η τιμή της m θα κυμαίνεται μεταξύ $h \le m \le h^2$

  το h μας δείχνει το πλήθος των σημείων που συμμετέχουν στην κατασκευή του κυρτού περιβλήματος
- Θα έχουμε (r – 1) ομάδες με μέγεθος m και η τελευταία ομάδα θα έχει μέγεθος (n – m), δηλαδή όσα σημεία απομένουν.

Αλγόριθμος του Chan

- Για κάθε ομάδα υπολογίζουμε το κυρτό περίβλημα χρησιμοποιώντας τον αλγόριθμο του Graham με χρονική πολυπλοκότητα $O(m \log m)$.

- Για τον υπολογισμό των ομάδων η συνολική χρονική πολυπλοκότητα θα είναι $O(rm \log m) = O(n \log m)$.

- Για την παράμετρο m ισχύει ότι $m \leq h^2$

- Η χρονική πολυπλοκότητα θα είναι $O(n \log h)$.

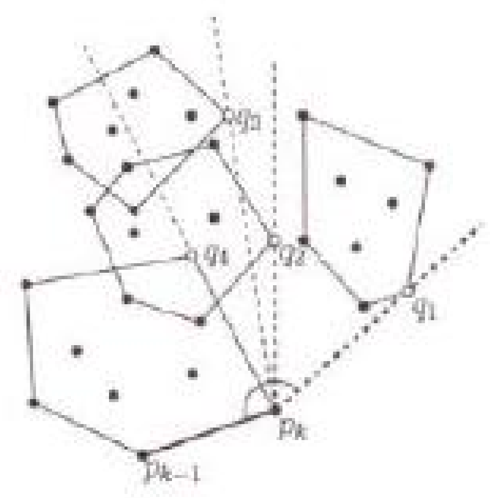
Αλγόριθμος του Chan

- Πως θα ενώσουμε αποδοτικά τα περιβλήματα των ομάδων ??

- Χρησιμοποιούμε τον αλγόριθμο του Jarvis, βρίσκοντας το πρώτο σημείο του περιβλήματος που έχει την μικρότερη y- συντεταγμένη.

- Κατόπιν παρατηρούμε ότι μπορούμε να βρούμε την κορυφή q που μεγιστοποιεί την γωνία $p_{k-1}p_k$ σε χρόνο log m.

Αλγόριθμος του Chan



Ο αλγόριθμος κατασκευής κυρτού περιβλήματος του Chan

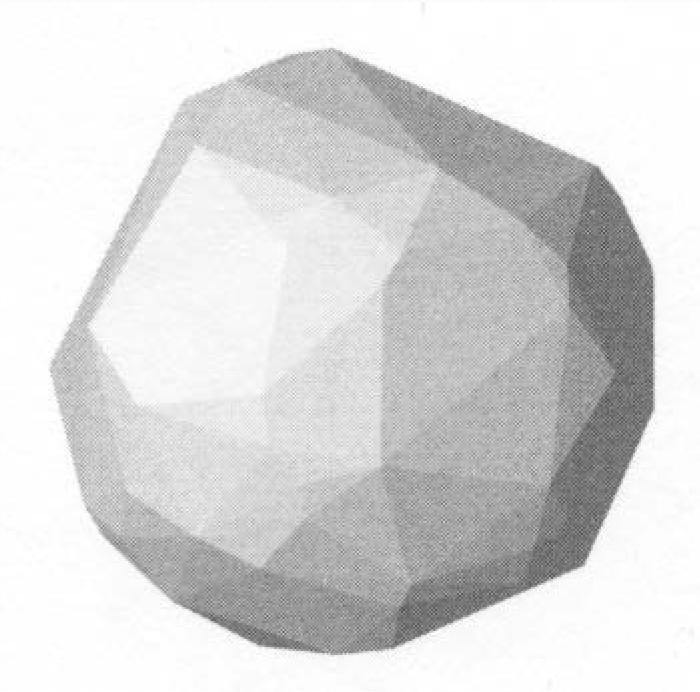
Αλγόριθμος του Chan

- Αναλύοντας το χρόνο εκτέλεσης ξέρουμε ότι ο αριθμός των σημείων στο τελικό περίβλημα είναι h γι’αυτό με τον αλγόριθμο του Jarvis χρειαζόμαστε h βήματα οπότε ο συνολικός χρόνος για την συγχώνευση των φάσεων του αλγορίθμου θα είναι O (n log h).

Αλγόριθμος του Chan

- Απομένει να βρούμε το μέγεθος της παραμέτρου m.
- Βάζουμε τιμές $m = 1,2,4,8, \ldots (2^0,2^1,2^2,2^3, \ldots)$ μέχρι να ικανοποιηθεί η συνθήκη $m \geq h$ αλλά η μέθοδος αυτή αργεί.
- Χρησιμοποιούμαι τη μέθοδο της δυαδικής αναζήτησης η οποία όμως έχει χρόνο εκτέλεσης $O(n \log n)$ που είναι αργός.
- Ο Chan διπλασίασε την προηγούμενη τιμή του m μέχρι να πετύχουμε το σωστό αποτέλεσμα $m = 2,4,16,\ldots (2^1,2^2,2^4, \ldots)$

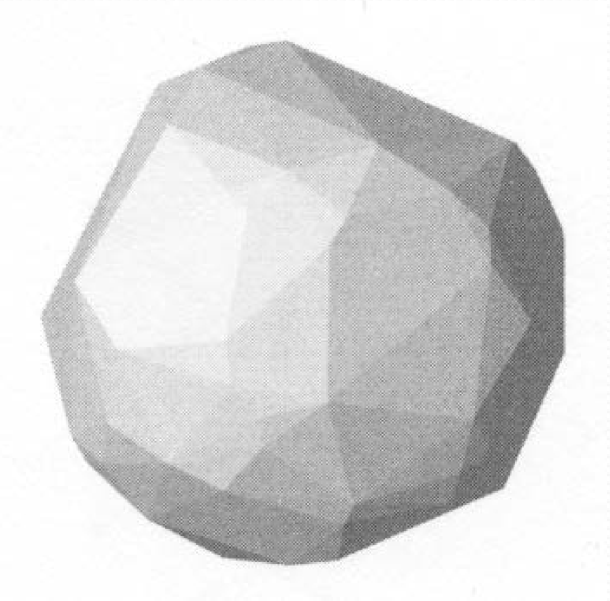
Κατασκευή κυρτών περιβλημάτων σε τρείς διαστάσεις



3d convex hull

- Μας δίδεται ένα σύνολο P που περιλαμβάνει n σημεία

- Και θα πρέπει
  - Να κατασκευάσουμε ένα κυρτό περίβλημα P: $\mathcal{CH}(P)$
  - Το μικρότερο πολύεδρο που περιλαμβάνει όλα τα στοιχεία του P στο εσωτερικό του $\mathcal{CH}(P)$.



3d convex hull - πολυπλοκότητα

- Η χρονική πολυπλοκότητα του *κυρτού περιβλήματος για* n *σημεία στις τρείς διαστάσεις είναι* $O(n)$
- *Λόγω του ότι ο αριθμός των ακμών στο κυρτό πολύεδρο με* n *κορυφές είναι το πολύ* $3n - 6$ *και ο αριθμός των εδρών είναι το πολύ* $2n - 4$ *ο γράφος ορίζεται από τις κορυφές και τις έδρες του κυρτού πολύεδρου στο επίπεδο*
- *Ο τύπος του Euler :* $n - n_e + n_f = 2$

n = κόμβοι n_e =*τόξα* nf = *έδρες*

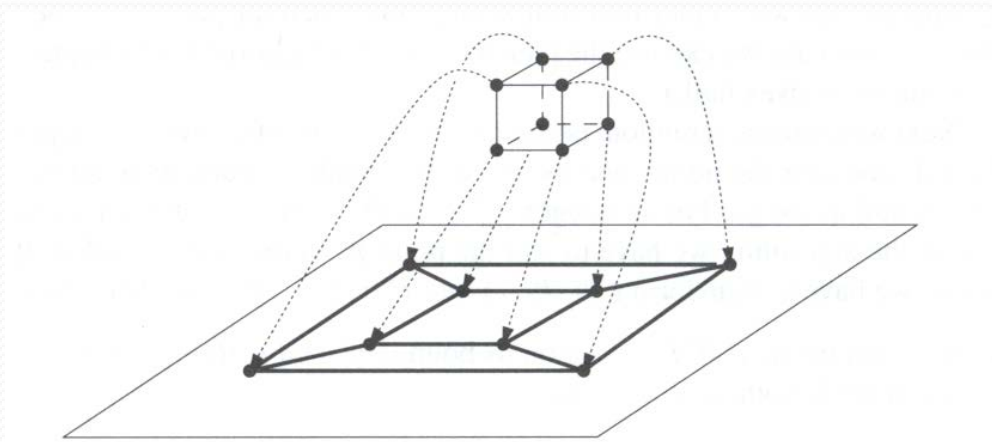
3d convex hull - πολυπλοκότητα

- Κάθε έδρα έχει 3 τόξα το πολύ
- Κάθε τόξο προσπίπτει σε δύο έδρες

$$2n_e \geq 3n_f$$

- Χρησιμοποιώντας το τύπο του Euler:

$$n_f \leq 2n\text{-}4 \qquad n_e \leq 3n\text{-}6$$



3d convex hull - αλγόριθμος

- Randomized incremental algorithm

- Βήματα:

- Ξεκινάει ο αλγόριθμος
- Προσθέτει το p_r *στο κυρτό περίβλημα του* $P_{r\text{-}1}$ *για να μετασχηματίσει το κυρτό περίβλημα* $\mathcal{CH}(P_{r\text{-}1})$ στο $\mathcal{CH}(P_r)$

[για κάθε ακέραιο $r \geq 1$, θέσε $P_r := \{p_1, \ldots, p_r\}$]

3d convex hull - αλγόριθμος

- Χρειάζεται ένα κυρτό περίβλημα για να ξεκινήσει
- Κατασκευάζει ένα τετράεδρο χρησιμοποιώντας 4 σημεία του P
- Ξεκινάει με δυο διαφορετικά σημεία του P (P1 και P2)
- Αναζητεί στο σύνολο P το σημείο P3, τέτοιο ώστε να μην είναι στην ίδια ευθεία με τα (P1 και P2)
- Βρίσκει το σημείο P4, τέτοιο ώστε να μην είναι στην ίδια ευθεία με τα (P1,P2, P3)
- Εάν δεν υπάρχει τέτοια περίπτωση σημείων τότε τα σημεία βρίσκονται στο ίδιο επίπεδο
- Υπολόγισε μια τυχαία μετάθεση για τα υπόλοιπα σημεία P5,… Pn

Εισαγωγή σημείων στο $\mathcal{CH}$

- Πρόσθεσε το p_r στο κυρτό περίβλημα του P_{r-1} και μετασχημάτισε το $\mathcal{CH}(P_{r-1})$ σε $\mathcal{CH}(P_r)$

  [για ακέραιο $r \geq 1$, θέσε $P_r := \{p_1, \ldots, p_r\}$]

- Δυο περιπτώσεις:

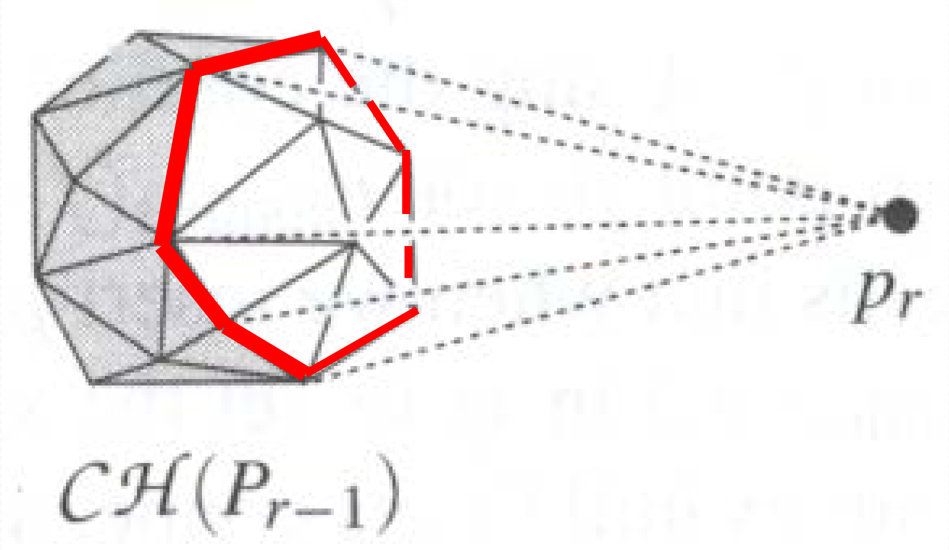
  1) Το σημείο P_r βρίσκεται εσωτερικά του κυρτού περιβλήματος $\mathcal{CH}(P_{r-1})$

     Τότε ισχύει: $\mathcal{CH}(P_r) = \mathcal{CH}(P_{r-1})$

  2) Το σημείο P_r βρίσκεται εξωτερικά του $\mathcal{CH}(P_{r-1})$

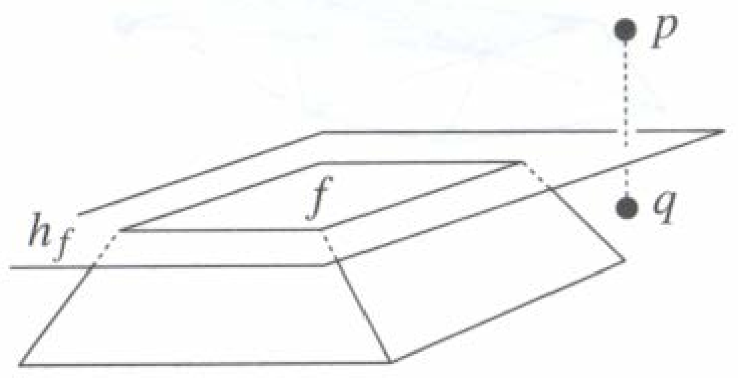
Περίπτωση 2: το P_r βρίσκεται εξωτερικά του $\mathcal{CH}(P_{r-1})$

- Προσδιορίζουμε τον ορίζοντα του p_r στο $\mathcal{CH}(P_{r-1})$
  - Μια κλειστή καμπύλη ακμών πριλαμβάνουν την ορατή περιοχή του p_r στο $\mathcal{CH}(P_{r-1})$



Ορατότητα

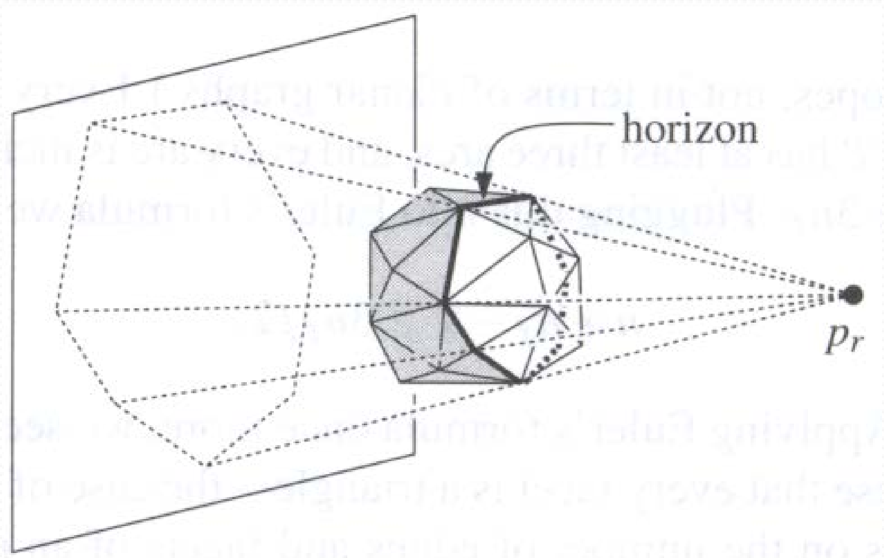
- Ας θεωρήσουμε ότι η επίπεδη πλευρά h_f περιλαμβάνει μια έδρα f του $\mathcal{CH}(P_{r-1})$
- Η έδρα f είναι ορατή από το σημείο P εάν το σημείο βρίσκεται στο επάνω μισό την πλευράς h_f



f is visible from p,
but not from q

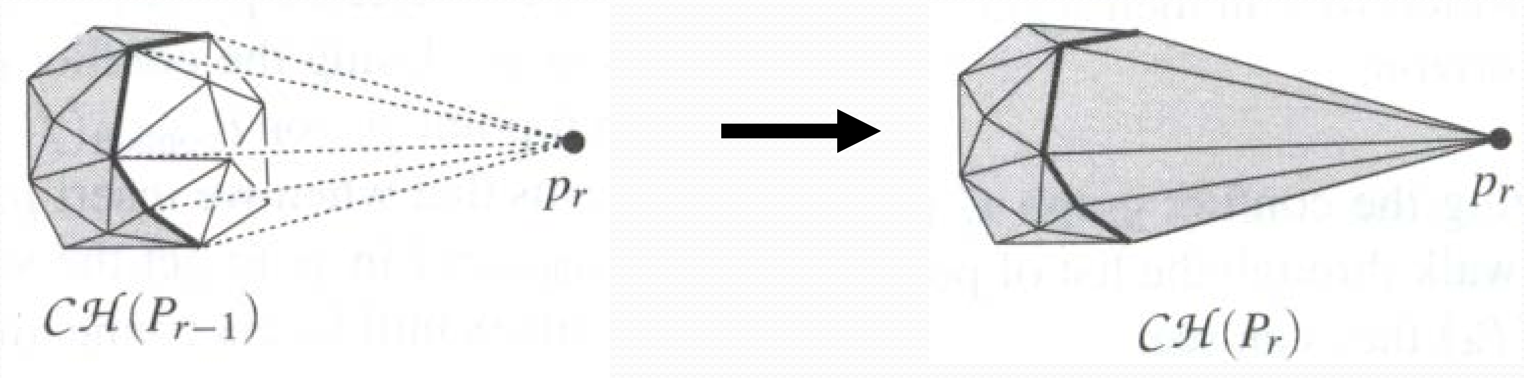
Ξανά βλέποντας τον ορίζοντα

- Το σύνορο του πολύγωνου επιτυγχάνεται προβάλλοντας το $\mathcal{CH}(P_{r-1})$ σε ένα επίπεδο με το p_r ως κέντρο της προβολής



$\mathcal{CH}(P_{r-1}) \rightarrow \mathcal{CH}(P_r)$

- Αφαιρούμε τις ορατές έδρες από το $\mathcal{CH}(P_{r-1})$
- Βρίσκουμε τον ορίζοντα: Κλειστή καμπύλη από ακμές του $\mathcal{CH}(P_{r-1})$
- Σχηματίζουμε το $\mathcal{CH}(P_r)$ συνδέοντας κάθε έδρα του ορίζοντα με το σημείο p_r για να δημιουργήσουμε μια καινούργια τριγωνική έδρα.



Ο αλγόριθμος μέχρι τώρα…

- Ξεκινά
  - Σχηματίζει ένα τετράεδρο $\mathcal{CH}(P_4)$ χρησιμοποιώντας 4 σημεία του P
  - Υπολογίζει μια τυχαία μετάθεση των υπολοίπων σημείων.
- Για κάθε σημείο του P που παραμένει
  - p_r είναι ένα σημείο που πρέπει να εισαχθεί
  - Εάν το p_r είναι εξωτερικά του $\mathcal{CH}(P_{r-1})$ τότε
    - Προσδιόρισε την ορατή περιοχή
    - Βρες τον ορίζοντα και αφαίρεσε τις ορατές έδρες
    - Πρόσθεσε καινούργιες έδρες συνδέοντας τις ακμές του ορίζοντα στο p_r

Πως όμως καθορίζουμε την ορατή περιοχή?

Πως βρίσκουμε την ορατή περιοχή

- Αφελής προσέγγιση:
  - Δοκίμασε κάθε έδρα στο p_r
  - *Χρόνος πολυπλοκότητας* $O(n^2)$

- Κολπάκι για να συνεχίσουμε:

  Διατηρούμε τις πληροφορίες για να τις προσθέτουμε στις ορατές έδρες

Λίστες συγκρούσεων

- Για κάθε έδρα f που διατηρούμε

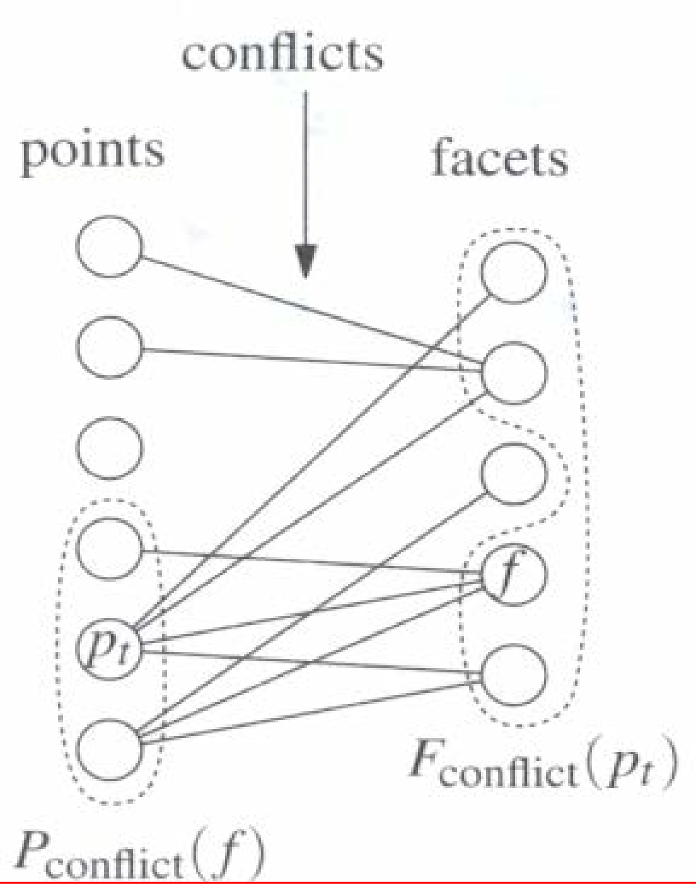
  $P_{\text{conflict}}(f) \subseteq \{p_{r+1}, \ldots, p_n\}$

  περιλαμβάνουμε τα σημεία που πρέπει να εισαχθούν για να μπορούμε να βλέπουμε την f
- Για κάθε p_t, όπου $t > r$, παραμένει $F_{\text{conflict}}(p_t)$

  συνδέουμε τις έδρες του $\mathcal{CH}(P_r)$ που είναι ορατές από το p_t

- p και f βρίσκονται σε σύγκρουση διότι δεν μπορούν να συνυπάρχουν στο ίδιο κυρτό περίβλημα

Γράφος συγκρούσεων $\mathcal{G}$

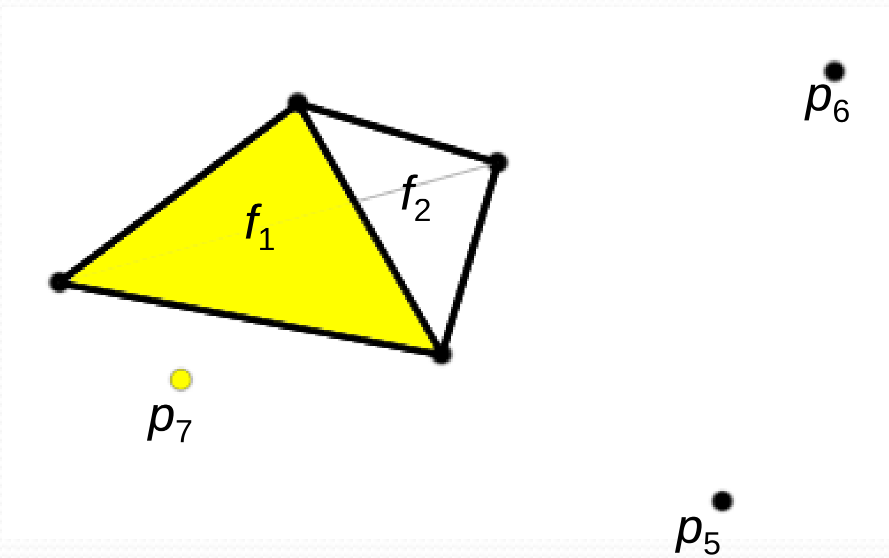


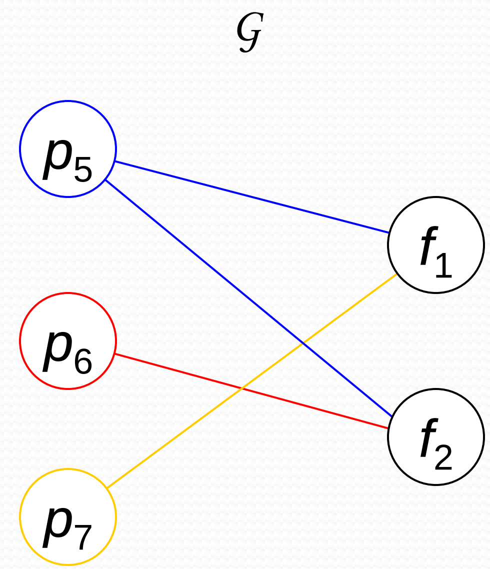
- Διμερής γράφος
  - Τα σημεία δεν έχουν ακόμα εισαχθεί
  - Οι έδρες ανήκουν στο $\mathcal{CH}(P_r)$
- Arc for every point-facet conflict
- Η λίστα συγκρούσεων για κάθε σημείο ή έδρα μπορεί να επιστρέφει σε γραμμικό χρόνο

Σε κάθε βήμα του αλγόριθμου, γνωρίζουμε όλες τις συγκρούσεις μεταξύ των σημείων που παραμένουν και των εδρών στο υπάρχον $\mathcal{CH}$

Ξεκινώντας $\mathcal{G}$

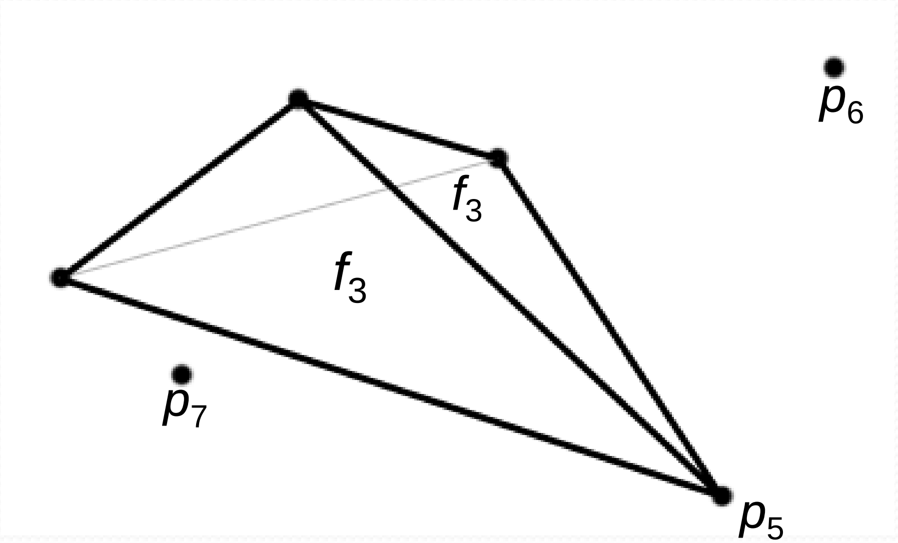
- Ξεκινώντας το $\mathcal{G}$ with $\mathcal{CH}(P_4)$ σε γραμμικό χρόνο
- Προσπελαύνουμε τα σημεία $P_{5\text{-}n}$ για να καθορίσουμε ποιες έδρες κάθε σημείο μπορεί να δεί



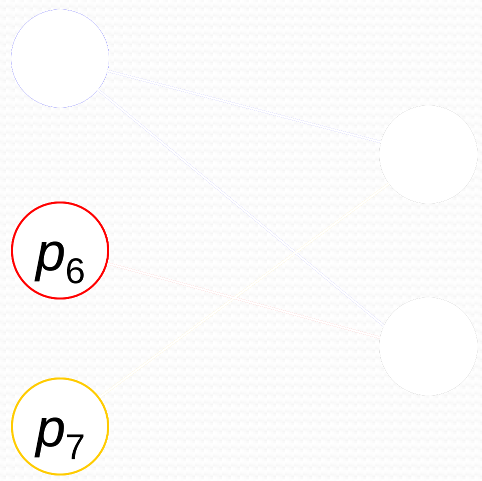


Ενημερώνοντας το $\mathcal{G}$

- Απορρίπτουμε τις ορατές έδρες p_r απομακρύνοντας τους γείτονες του p_r στο $\mathcal{G}$
- Απομακρύνουμε το p_r από το $\mathcal{G}$
- Καθορίζουμε την καινούργια λίστα συγκρούσεων

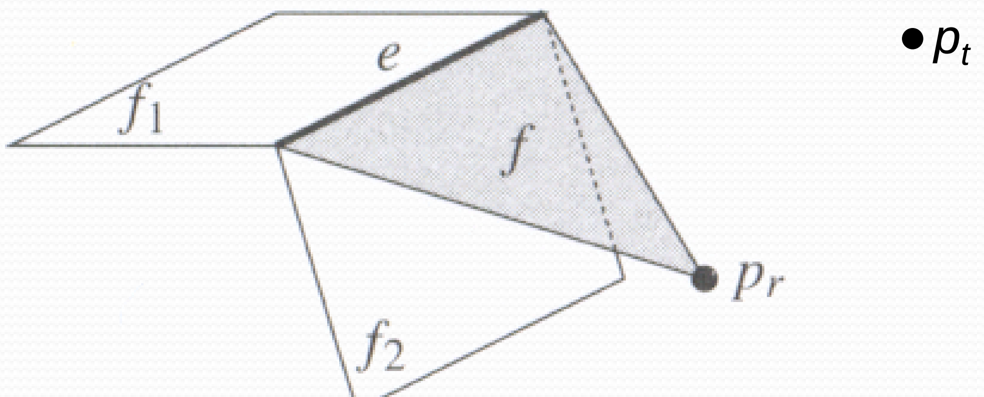


$\mathcal{G}$



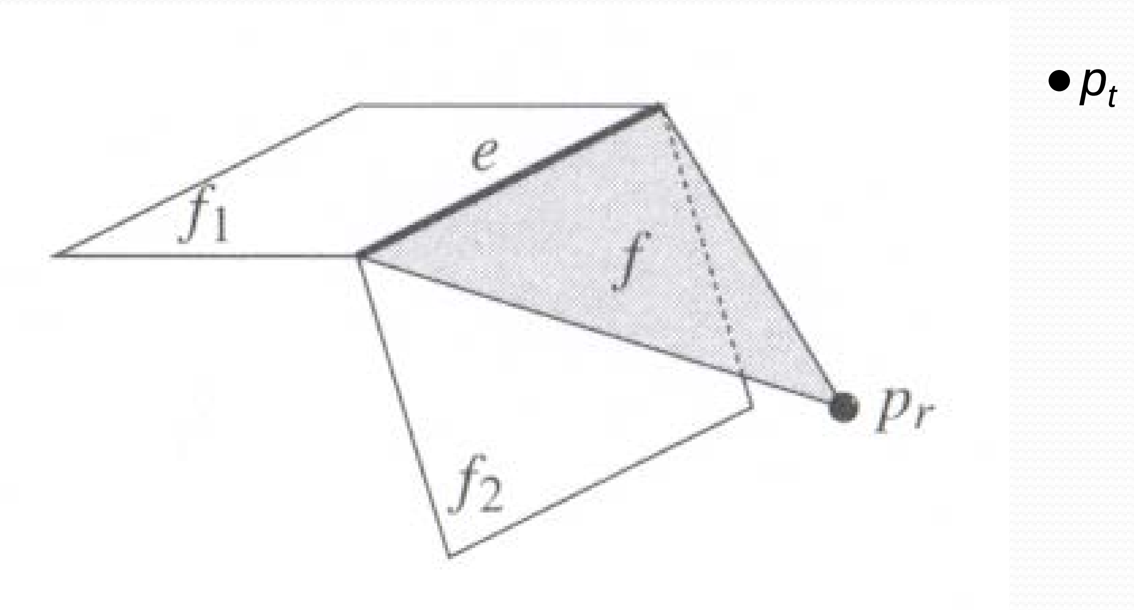
Προσδιορίζοντας την καινούργια λίστα συγκρούσεων

- Εάν p_t μπορεί να δει την καινούργια f, τότε μπορεί να δει την ακμή e του f.
- *Η* e ανήκει στον ορίζοντα του p_r, έτσι η e *ανήκει και είναι ορατή από το* p_t στο $\mathcal{CH}(P_{r\text{-}1})$
- Εάν το σημείο p_t βλέπει την e, θα βλέπει είτε την f_1 ή την έδρα f_2 στο $\mathcal{CH}(P_{\mathrm{r}\text{-}1})$
- P_t ήταν μέσα στην λίστα $P_{\mathrm{conflict}}(f_1)$ οή στην λίστα $P_{\mathrm{conflict}}(f_2)$ του $\mathcal{CH}(P_{\mathrm{r}\text{-}1})$



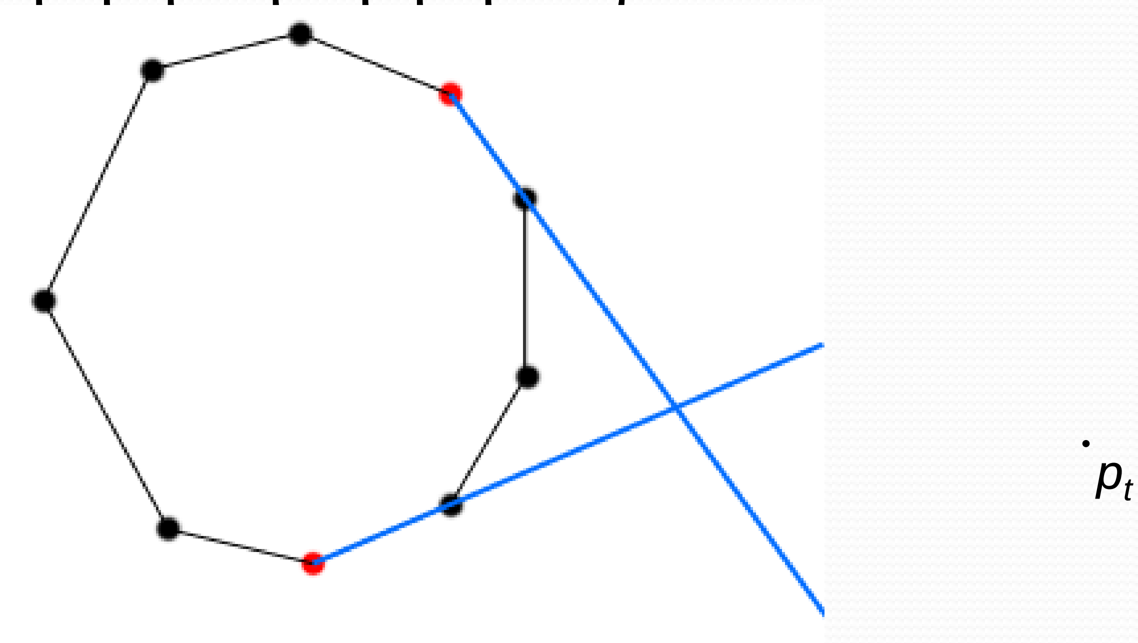
Προσδιορίζοντας την καινούργια λίστα συγκρούσεων

- Η λίστα συγκρούσεων του f μπορεί να βρεθεί δοκιμάζοντας τα σημεία από τις λίστες συγκρούσεων των f_1 και f_2 συγκρινόμενη με την ακμή του ορίζοντα e του $\mathcal{CH}(P_{r-1})$



Και οι υπόλοιπες έδρες ??

- *Η λίστα συγκρούσεων* $P_{\text{conflict}}(f)$ για κάθε f που δεν επηρεάζεται από την p_r παραμένει χωρίς να την πειράξουμε
- Οι έδρες που δεν άνηκαν στον ορίζοντα και διαγράφηκαν ήδη προσμετρήθηκαν για



Τελικός αλγόριθμος

- Ξεκινά με το $\mathcal{CH}(P_4)$ and $\mathcal{G}$
- Για κάθε σημείο που παραμένει
  - Καθορίζει τις ορατές έδρες για το p_r ελέγχοντας το $\mathcal{G}$
  - Αφαιρεί την $F_{\text{conflict}}(p_r)$ από το $\mathcal{H}$
  - Βρίσκει τον ορίζοντα και προσθέτει καινούργιες έδρες στο $\mathcal{CH}$ και στο $\mathcal{G}$
  - Ενημερώνει το $\mathcal{G}$ για καινούργιες έδρς δοκιμάζοντας τα σημεία από τις λίστες συγκρούσεων για τις έδρες στο $\mathcal{CH}(P_{\text{r-1}})$ σε σχέση με το e στις καινούργιες έδρες
  - Σβήνει το p_r και τις λίστες $F_{\text{conflict}}(p_r)$ από το $\mathcal{G}$

Χρόνος εκτέλεσης

- Ξεκίνημα αλγορίθμου $\Rightarrow O(n\log n)$
- Δημιουργώντας και σβήνοντας έδρες$\Rightarrow O(n)$
- Ενημερώνοντας $\mathcal{G} \Rightarrow O(n)$
- Βρίσκοντας όλες τις συγκρούσεις$\Rightarrow O(n\log n)$

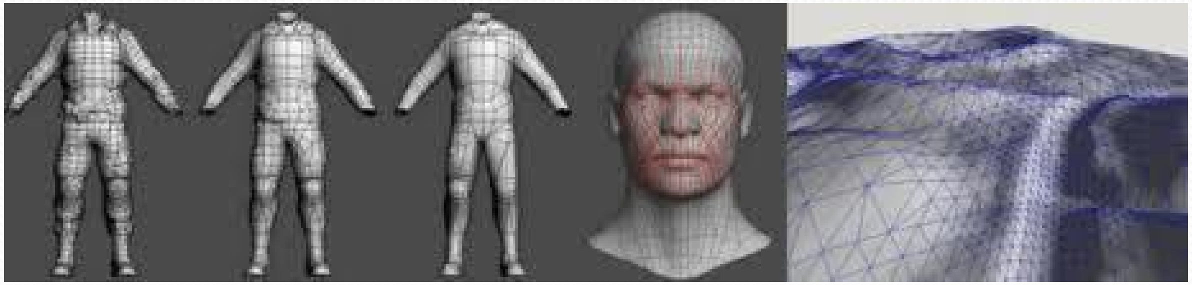
- Συνολικός χρόνος εκτέλεσης $\mathrm{O}(n\log n)$

ΕΦΑΡΜΟΓΕΣ

- Υπολογιστική Γραφιστική
- Η υπολογιστική γραφιστική ασχολείται με τη δημιουργία εικόνων, τυποποιημένων σκηνών, με σκοπό την προβολή τους μέσω οθόνης, εκτυπωτή, ή κάποιας άλλης συσκευής εξόδου. Δεδομένου ότι οι εικόνες αποτελούνται από γεωμετρικά αντικείμενα, οι γεωμετρικοί αλγόριθμοι παίζουν σημαντικό ρόλο στην υπολογιστική γραφιστική.

ΕΦΑΡΜΟΓΕΣ

- Υπολογιστική Γραφιστική
- Τα γεωμετρικά αντικείμενα μπορούν να αποτελούνται από δισδιάστατα σχέδια (ευθείες, πολύγωνα) έως τρισδιάστατες σκηνές που περιέχουν πηγές φωτός, αντικείμενα διαφορετικής υφής κ.ο.κ. Μια σκηνή αυτού του τύπου μπορεί να περιλαμβάνει πάνω από ένα εκατομμύριο πολύγωνα ή τμήματα καμπύλων επιφανειών

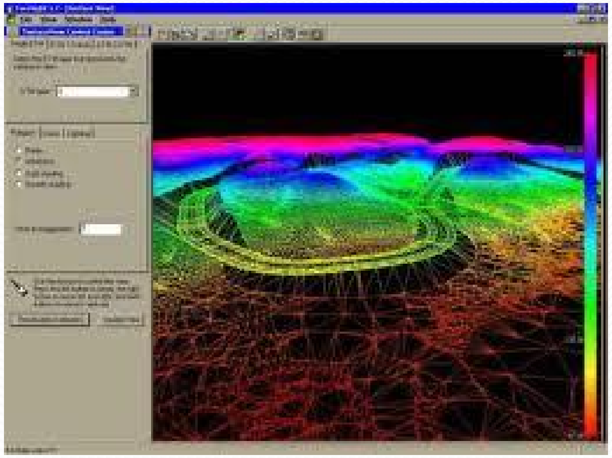


ΕΦΑΡΜΟΓΕΣ

- Ρομποτική
- Τα ρομπότ είναι γεωμετρικά αντικείμενα που λειτουργούν σε τρισδιάστατο περιβάλλον και στη διαχείρισή τους ανακύπτουν πολλά γεωμετρικά προβλήματα. Ο σχεδιασμός της κίνησης είναι ένα πρόβλημα στο σχεδιασμό των εργασιών.

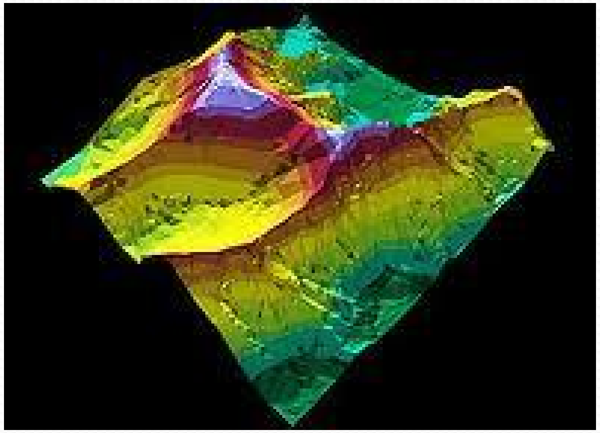
ΕΦΑΡΜΟΓΕΣ

- Γεωγραφικά Συστήματα Πληροφοριών
  - Ένα γεωγραφικό σύστημα πληροφοριών φέρει αποθηκευμένες ανθρωπογενείς δομές π.χ πόλεις, δρόμους, σιδηροδρομικά και ηλεκτρικά δίκτυα, καθώς και γεωγραφικά δεδομένα, όπως το ύψος των βουνών, η ροή των ποταμών, στοιχεία για τον πληθυσμό κ.ο.κ.



ΕΦΑΡΜΟΓΕΣ

- Γεωγραφικά Συστήματα Πληροφοριών
- Το τυπικότερο πρόβλημα είναι η εύρεση όλων των σημείων, τα οποία βρίσκονται σε δεδομένο ορθογώνιο παραλληλόγραμμο στο επίπεδο



ΕΦΑΡΜΟΓΕΣ

- Γεωγραφικά Συστήματα Πληροφοριών
- Η υποδιαίρεση του επιπέδου σε περιοχές που βρίσκονται πιο κοντά σε μια κεραία εκπομπής, ή σε ένα νοσοκομείο, μπορεί να αναχθεί στην κατασκευή του διαγράμματος Voronoi, γνωστό και ως πρόβλημα του ταχυδρόμου, διότι αποτυπώνει τις σχέσεις γειτονίας σε ένα σύνολο σημείων.