

Γραφικά Υπολογιστών

Ιόνιο Πανεπιστήμιο
Τμήμα Πληροφορικής

Οι διαφάνειες βασίζονται σε videos διαλέξεων του:

Cem Yuksel

*Assoc. Prof,
School of Computing, University of Utah)*

Στέργιος Παλαμάς, Επίκουρος Καθηγητής



Μάθημα 14:

Triangular Meshes

Triangular Meshes

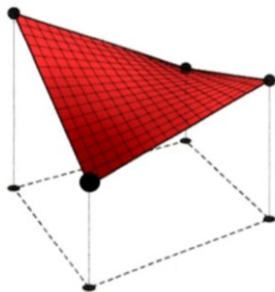
Όποιον τρόπο και να χρησιμοποιεί το πρόγραμμα 3D modeling που θα χρησιμοποιήσουμε για την παραγωγή των 3D μοντέλων (πχ nurb surfaces), στο τέλος θα μετατρέψουμε το μοντέλο σε triangular mesh. Και υπάρχει ένας πολύ καλός λόγος για αυτό.

Τα τρίγωνα μπορούν να οριστούν με 3 σημεία στο χώρο αλλάζοντας θέση στα σημεία που τα ορίζουν (vertices) και κάθε φορά τα 3 αυτά σημεία θα ορίζουν ένα **μοναδικό επίπεδο στο χώρο**.

Triangles



Αυτό ΔΕΝ ισχύει για περισσότερα σημεία. Αν είχαμε πχ 4 σημεία αυτά ΔΕΝ θα είναι υποχρεωτικά στο ίδιο επίπεδο.



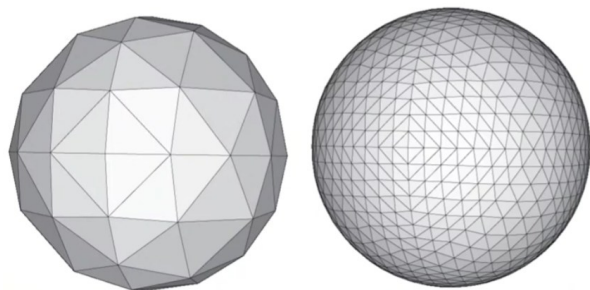
Triangular Meshes

Τί γίνεται όμως όταν θέλουμε καμπύλες επιφάνειες; Σε αυτή την περίπτωση **χρησιμοποιούμε πολλά και μικρότερα τρίγωνα συνδεδεμένα σε πλέγμα (triangular mesh)**. Όσα περισσότερα χρησιμοποιούμε τόσο καλύτερη προσέγγιση μπορούμε να πετύχουμε.

Με triangular mesh ποτέ δε θα επιτύχουμε μια τέλεια σφαίρα, αλλά με την κατάλληλη ποσότητα και μέγεθος τριγώνων, η διαφορά μπορεί να μην είναι πλέον ορατή, πχ αν τα τρίγωνα έχουν μέγεθος ανάλογο ενός pixel.

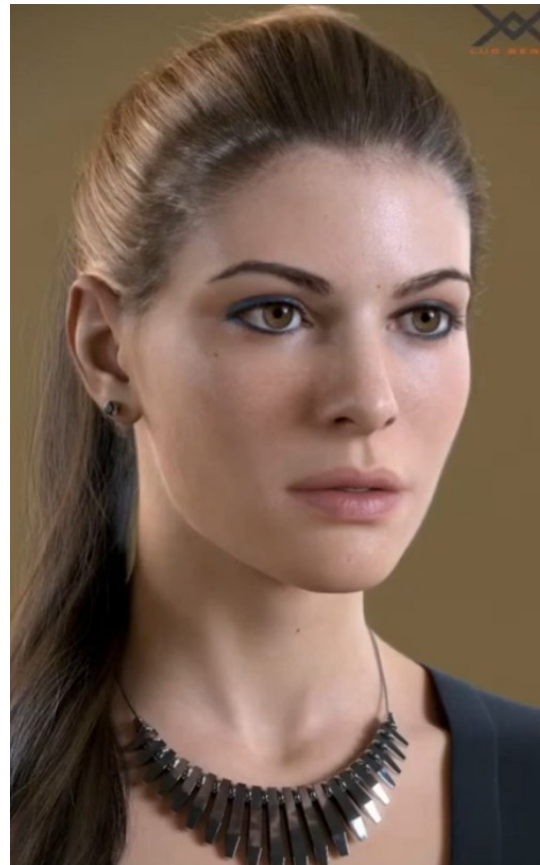
Έτσι μπορούμε να επιτύχουμε την αναπαράσταση πραγματικά πολύπλοκων μοντέλων, χωρία να χάνουμε την απλότητα που μας δίνουν τα τρίγωνα.

Triangular Mesh



Triangular Meshes

Υπάρχουν όμως και λίγες περιπτώσεις που δε μπορούν να χρησιμοποιηθούν τρίγωνα , όπως η περίπτωση των μαλλιών όπου χρησιμοποιούνται πχ splines (καμπύλες). Υπάρχουν ωστόσο ακόμη και για αυτές τις περιπτώσεις τεχνικές για την μετατροπή τους σε triangular meshes.

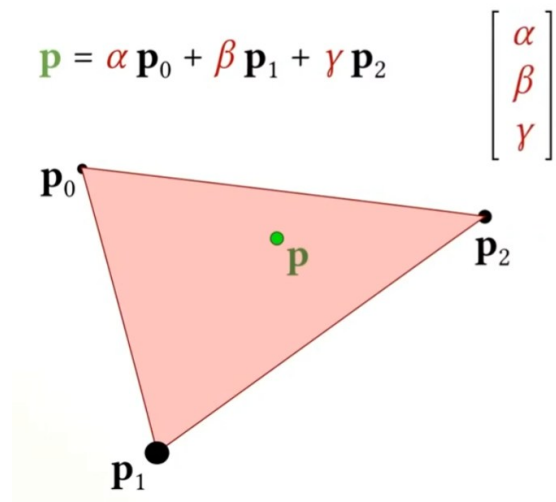


Triangular Meshes

Επιστρέφουμε λοιπόν στα τρίγωνα. Έχουμε ένα τρίγωνο και ένα σημείο στο εσωτερικό του (p).

Πρέπει για κάθε σημείο (p) στο επίπεδο του τριγώνου να μπορούμε να προσδιορίσουμε τη θέση του στο χώρο.

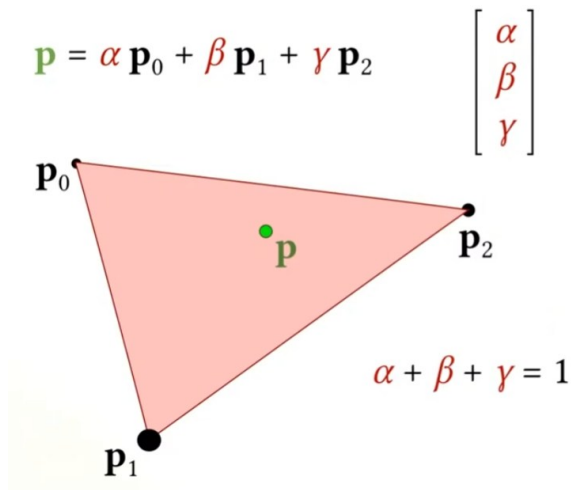
Το πλεονέκτημα είναι ότι **οποιοδήποτε σημείο που βρίσκεται στο επίπεδο που ορίζει το τρίγωνο μπορεί να προσδιοριστεί σαν ένας γραμμικός συνδυασμός των τριών σημείων p_0 , p_1 και p_2 (vertices) που ορίζουν το τρίγωνο.**



Τα α , β και γ τα ονομάζουμε βαρικεντρικές συντεταγμένες (barycentric coordinates) του σημείου p για το συγκεκριμένο τρίγωνο.

Ιδιότητες των βαρυκεντρικών συντεταγμένων:

1. Οι βαρυκεντρικές συντεταγμένες δεν εξαρτώνται από τη διάσταση των συντεταγμένων των p_0 , p_1 και p_2 . Παράδειγμα το τρίγωνο θα μπορούσε να βρίσκεται στις δύο διαστάσεις και τα σημεία να έχουν μόνο συντεταγμένες x και y . Οι βαρυκεντρικές συντεταγμένες θα ήταν και πάλι 3. Το ίδιο αν τα σημεία είχαν τρεις συντεταγμένες (x, y και z), το ίδιο και με τετραδιάστατα σημεία κ.ο.κ.
2. Παρότι οι βαρυκεντρικές συντεταγμένες ορίζουν ένα σημείο σε ένα επίπεδο (το επίπεδο του τριγώνου) έχω 3 βαρυκεντρικές συντεταγμένες (α , β και γ). Θα έπρεπε να είναι αρκετές δύο από αυτές! Οι βαρυκεντρικές συντεταγμένες όμως έχουν την ιδιότητα **$\alpha + \beta + \gamma = 1$** . **Οπότε μου αρκεί να γνωρίζω ΔΥΟ από αυτές για να βρω την τρίτη.** Οι βαρυκεντρικές συντεταγμένες του p είναι μια γραμμική παρεμβολή (interpolation) των τριών κορυφών του τριγώνου οπότε πρέπει να αθροίζουν σε 1. Αυτός είναι ο μόνος τρόπος το p να ανήκει στο επίπεδο που ορίζει το τρίγωνο.



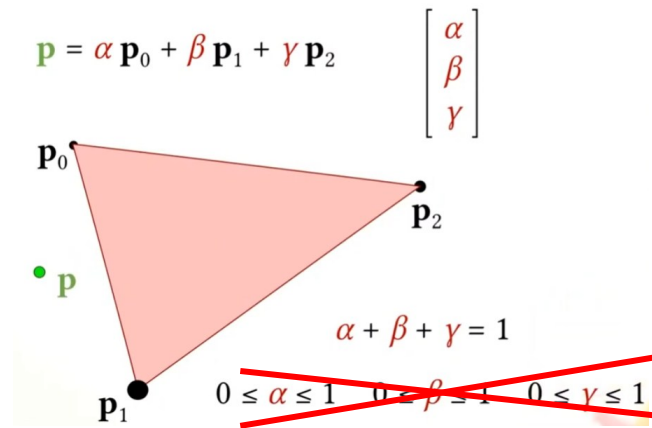
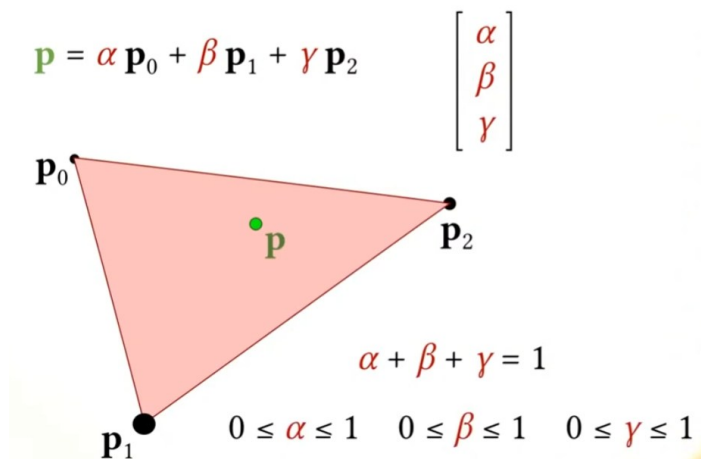
Triangular Meshes

Ιδιότητες των βαρυκεντρικών συντεταγμένων:

3. Μέχρι τώρα θεωρούσαμε ότι το σημείο p θα βρίσκεται ΕΝΤΟΣ του τριγώνου. Στην πράξη όμως ΟΠΟΙΟΔΗΠΟΤΕ σημείο του επιπέδου που ορίζει το τρίγωνο μπορεί να προσδιοριστεί με βαρυκεντρικές συντεταγμένες οι οποίες ΠΑΝΤΑ θα έχουν άθροισμα 1.

Αν όμως το σημείο p είναι ΕΝΤΟΣ του τριγώνου ισχύει επιπλέον ότι και οι 3 βαρυκεντρικές συντεταγμένες είναι θετικές και μεταξύ 0 και 1.

Αν το σημείο p είναι ΕΚΤΟΣ τριγώνου δεν ισχύει ο περιορισμός να είναι μεταξύ 0 και 1

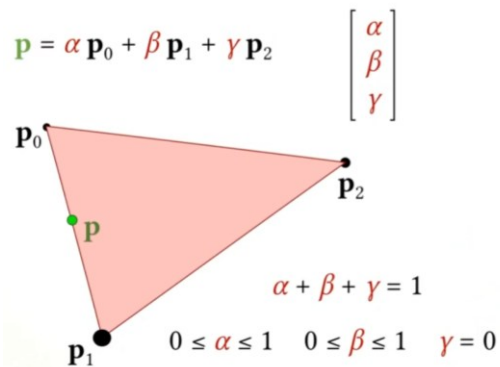


Triangular Meshes

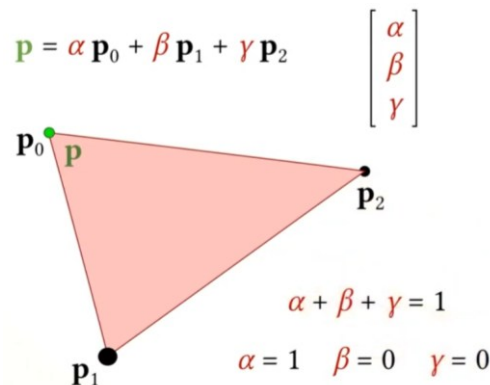
Ιδιότητες των βαρυκεντρικών συντεταγμένων:

4. Αν το σημείο p είναι σε μία από τις πλευρές του τριγώνου, καταλαβαίνουμε ότι πλέον μπορεί να προσδιοριστεί ΜΟΝΟ ΜΕ ΤΙΣ ΔΥΟ από τις τρεις κορυφές (vertices) , οπότε η ΤΡΙΤΗ βαρυκεντρική συντεταγμένη μηδενίζεται.

Πχ αν το p είναι στην πλευρά $p_0 - p_1$, τότε το γ γίνεται 0 αφού το p εξαρτάται πλέον μόνο από το p_0 και p_1 .



5. Αν πάμε το p σε μία από τις τρεις κορυφές, τότε καταλαβαίνουμε ότι η αντίστοιχη βαρυκεντρική συντεταγμένη γίνεται 1 και οι άλλες δύο 0.

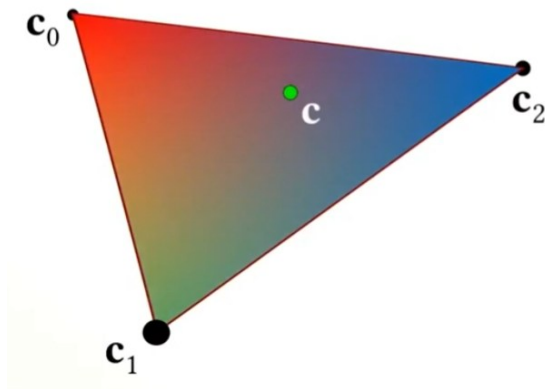


Triangular Meshes

Η ιδέα των βαρυκεντρικών συντεταγμένων μπορεί να επεκταθεί και πέρα από τον προσδιορισμό της θέσης των σημείων ενός τριγώνου.

Στις τρεις κορυφές (vertices) του τριγώνου, εκτός από τη θέση, μπορούμε να αποθηκεύουμε και διάφορες άλλες ιδιότητες, όπως πχ ένα χρώμα για κάθε vertex. Με τον ίδιο ακριβώς τρόπο τότε μπορούμε να προσδιορίσουμε το χρώμα κάθε σημείου του τριγώνου ως γραμμικό συνδυασμό των τριών χρωμάτων των vertices!

$$\mathbf{c} = \alpha \mathbf{c}_0 + \beta \mathbf{c}_1 + \gamma \mathbf{c}_2$$



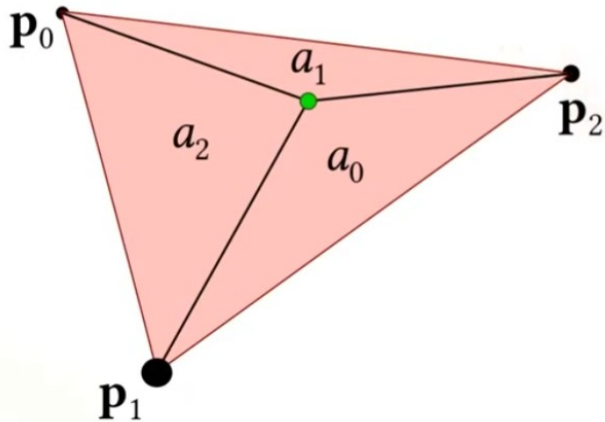
Triangular Meshes

Πώς όμως μπορούμε να υπολογίσουμε τις βαρυκεντρικές συντεταγμένες ενός σημείου;

Υπάρχουν διάφοροι τρόποι για τον υπολογισμό του, ένας από αυτούς είναι μέσω του εμβαδού των τριγώνων που σχηματίζονται εντός του τριγώνου.

Σε αυτή την περίπτωση αν a είναι το συνολικό εμβαδόν μπορούμε να υπολογίσουμε τις βαρυκεντρικές συντεταγμένες ως εξής:

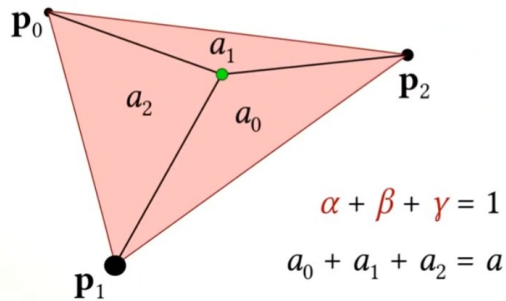
$$\alpha = \frac{a_0}{a} \quad \beta = \frac{a_1}{a} \quad \gamma = \frac{a_2}{a}$$



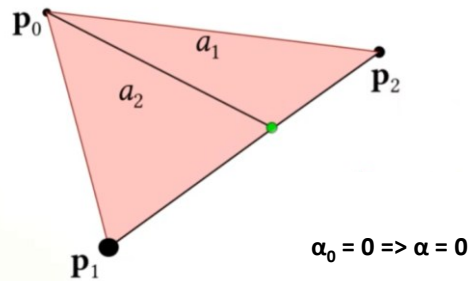
Triangular Meshes

Αυτός ο τρόπος υπολογισμού είναι συμβατός με τις ιδιότητες που είδαμε πριν:

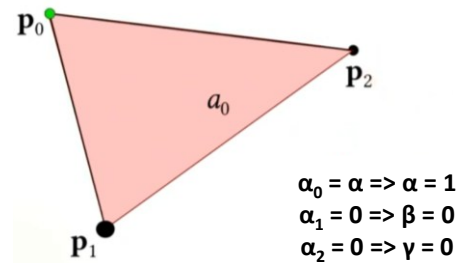
$$\alpha = \frac{a_0}{a} \quad \beta = \frac{a_1}{a} \quad \gamma = \frac{a_2}{a}$$



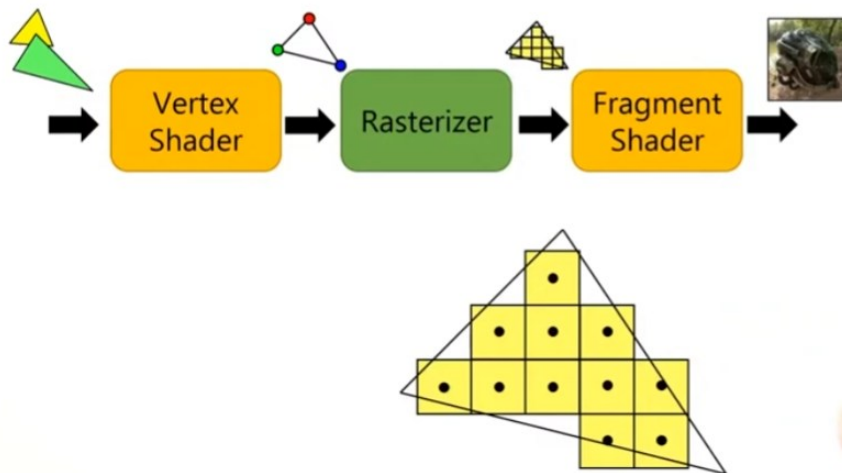
$$\alpha = \frac{a_0}{a} \quad \beta = \frac{a_1}{a} \quad \gamma = \frac{a_2}{a}$$



$$\alpha = \frac{a_0}{a} \quad \beta = \frac{a_1}{a} \quad \gamma = \frac{a_2}{a}$$



GPU Pipeline



Ο rasterizer στη GPU θα υπολογίσει τις βαρυκεντρικές συντεταγμένες για τα κέντρα των pixels (fragments) οπότε αυτές μπορούν να χρησιμοποιηθούν μετά στο fragment shader για το interpolation όλων των ιδιοτήτων που φέρουν τα vertices (όπως πχ το χρώμα, η φωτεινότητα)

Triangular Meshes

Πώς μπορούν να αναπαρασταθούν τα triangular meshes;

- List of vertices

- x, y, z position per vertex

Λίστα με όλα τα vertices η οποία για κάθε vertex θα έχει τις συντεταγμένες του (και πιθανότατα και τιμές για άλλες ιδιότητες του vertex όπως το χρώμα).

- List of triangles (faces)

- Vertex indices

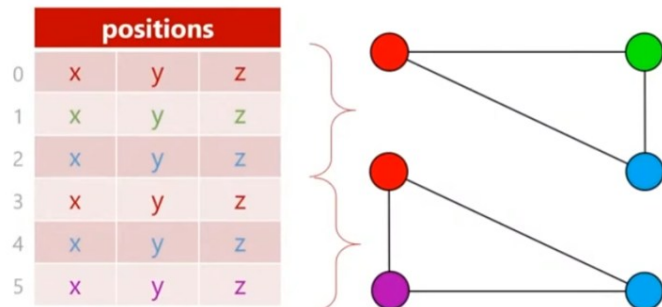
Λίστα με όλα τα τρίγωνα. Για κάθε τρίγωνο θα έχει τα indices των vertices που το απαρτίζουν.

- Vertex Attributes

positions				colors			
0	x	y	z	0	r	g	b a
1	x	y	z	1	r	g	b a
2	x	y	z	2	r	g	b a
3	x	y	z	3	r	g	b a
4	x	y	z	4	r	g	b a
5	x	y	z	5	r	g	b a

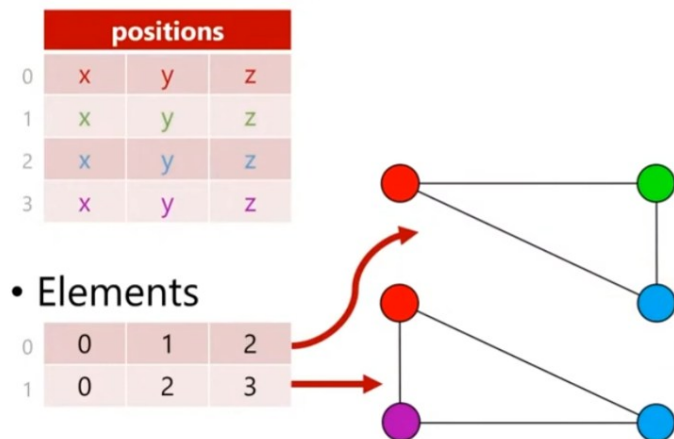
Triangular Meshes

- Vertex Attributes



Ένα πρόβλημα είναι ότι στην περίπτωση διαδοχικών τριγώνων , τα ίδια vertices θα επαναλαμβάνονται πολλές φορές, μία για κάθε τρίγωνο που ανήκουν.

- Vertex Attributes



- Elements

Ένα πρόβλημα είναι ότι στην περίπτωση διαδοχικών τριγώνων , τα ίδια vertices θα επαναλαμβάνονται πολλές φορές, μία για κάθε τρίγωνο που ανήκουν.

Μια πιο αποτελεσματική τεχνική είναι να έχω τη λίστα με τα vertexes και μια ξεχωριστή λίστα με τα elements (τρίγωνα στην περίπτωση μας) και για κάθε element να αποθηκεύω ΜΟΝΟ τα INDEXES των vertices που το απαρτίζουν αντί να επαναλαμβάνω το σύνολο των attributes κάθε vertex.

Triangular Meshes

Τα Triangle Strips μειώνουν πολύ την ανάγκη αποθήκευσης , αλλά εισάγουν μια επιπλέον πολυπλοκότητα: πλέον δε χρειάζεται μόνο να **κάνω την μετατροπή σε triangular mesh, αλλά πρέπει να εντοπίζω και στο mesh τα triangular strips.**

Γειτονικά strips θα έχουν vertices που επαναλαμβάνονται, αλλά τουλάχιστον μέσα στο ίδιο strip γλυτώνω duplications και επιπλέον, κάθε vertex το πολύ να επαναληφθεί κατά μέσο όρο 2 φορές (για το γειτονικό strip) και όχι πολλαπλές όπως είχαμε δει στην αρχή.



Μια εναλλακτική προσέγγιση είναι αυτή του σχηματισμού «triangular fans»