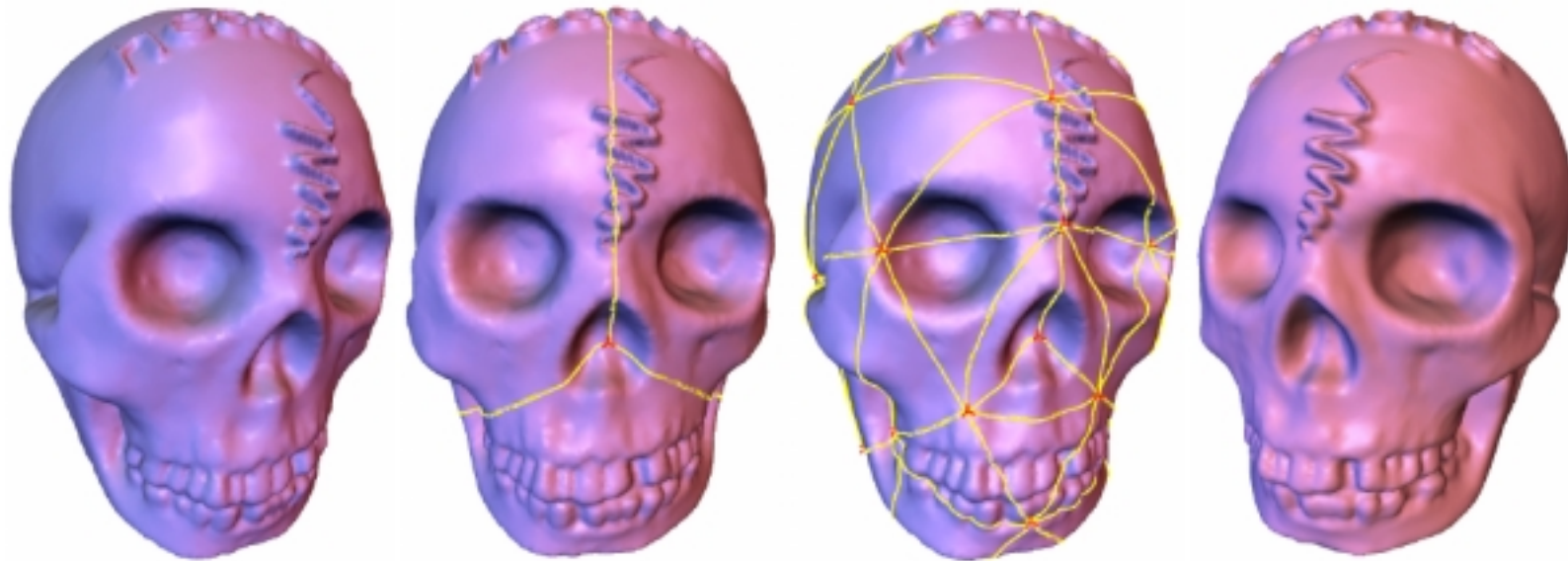


Normal Meshes

Igor Guskov, Kiril Vidimčič, Wim Sweldens, Peter Schröder
Proceedings of SIGGRAPH 2000



Phyrum Tea - Juni 2000
Betreuer: Andreas Hubeli

Vortragsübersicht



- ⌘ Einleitung in Semi-Reguläre Mesh
- ⌘ Motivation
- ⌘ Normal Mesh
- ⌘ Resultate
- ⌘ Persönliche Beurteilung

Einleitung



- ⌘ Approximation von Oberflächen
- ⌘ Rohscan als Input
- ⌘ Bearbeitung der Objekte
- ⌘ Viele Algorithmen benötigen semi-seguläre Mesh als Eingabe

Semi-Reguläre Mesh



⌘ Remeshing nötig

⌘ Andere Verfahren

☒ [8] Multires. Analysis of Arbitrary Meshes (95)

☒ [13] Fitting Smooth Surfaces to Dense Polygon Meshes (96)

☒ [15] MAPS: Multires. Adaptive Parameterization of Surfaces (98)

⌘ Gemeinsamkeiten

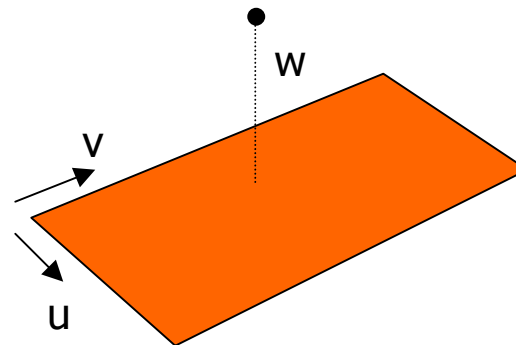
☒ beliebige Mesh als Eingabe

☒ Semi - reguläre Mesh als Ausgabe

Motivation

⌘ Situation

- ☒ Ein Punkt braucht bisher 3 Float-Werte u, v, w



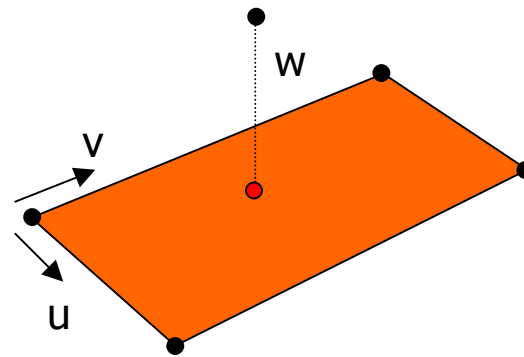
⌘ Ziel

- ☒ Ein Punkt braucht 1 Float-Wert

Parameter

⌘ Wie erreicht man dies?

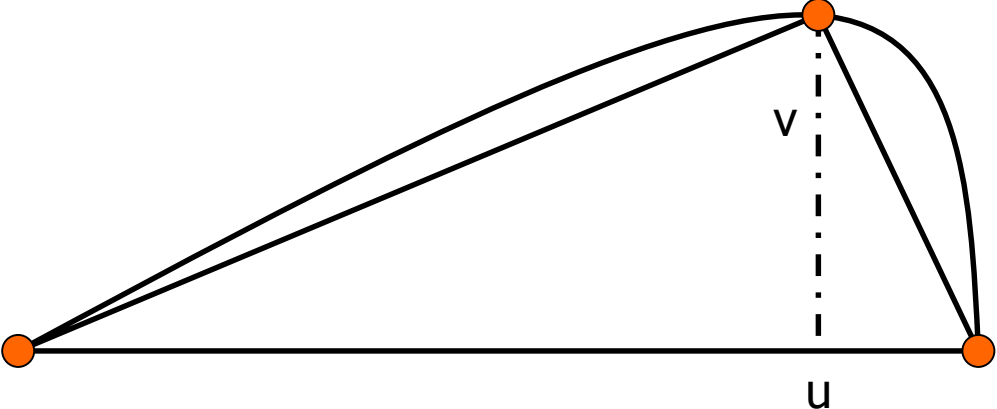
- ☒ Nachbarschaft des Punktes
- ☒ u, v eigentlich Parameter



Normal Polylines

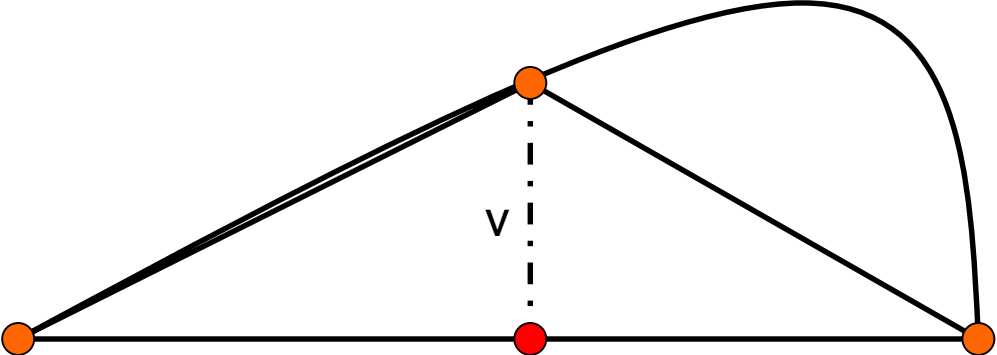
Allgemein

- 1 Parameter u
- 1 Geometrie v



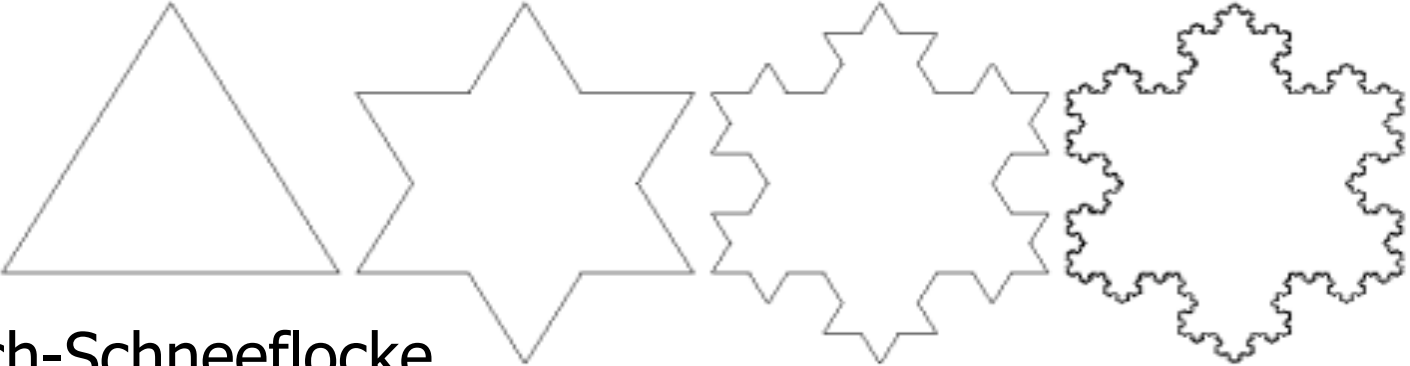
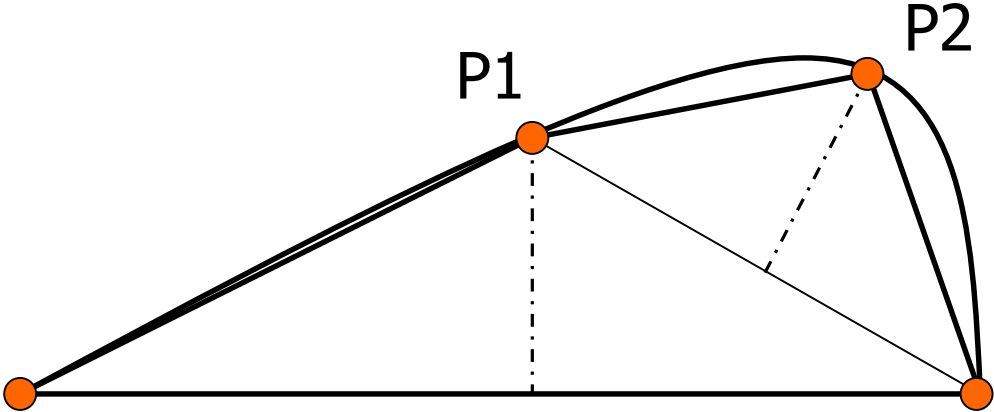
Implizit in der Mitte

- 0 Parameter
- 1 Geometriewert v



Normal Polylines

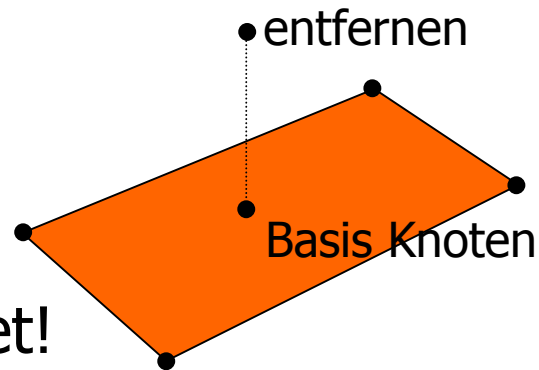
2 Neue Punkte
0 Parameter
2 Geometrie



Koch-Schneeflocke

Definition von Normal Meshes

Definition: „A mesh M is normal in case a sequence of vertex removals exists so that each removed vertex lies on a line defined by a **base point** and **normal direction** which only depends on the remaining vertices.“



Basis Knoten berechnet!

entfernter Knoten rekonstruierbar

Probleme



⌘ Probleme

- ☒ grobe Basis Mesh nötig

- ☒ Die Eingangsdaten erfüllen Bedingung nicht

⌘ Remeshing notwendig

Remeshing



⌘ Einsparung nur durch Neuordnung

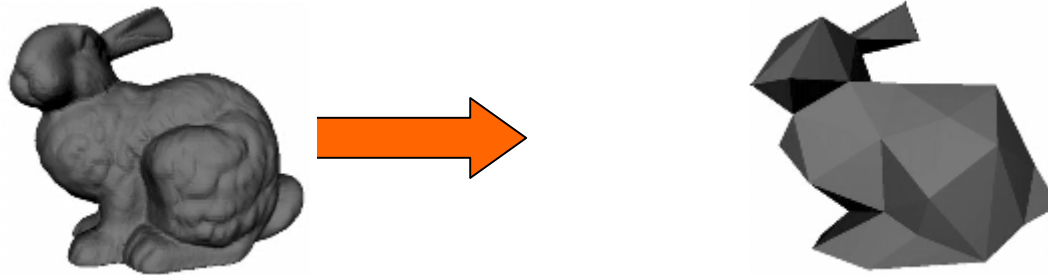
⌘ Diese Arbeit hat 7 stufigen Algorithmus

- ☒ 1. Mesh Vereinfachung
- ☒ 2. Startnetz von Kurven
- ☒ 3. Globale Knoten fixieren
- ☒ 4. Globale Kanten fixieren
- ☒ 5. Initiale Parametrisierung
- ☒ 6. Durchstosspunkte finden
- ☒ 7. Parametrisierung anpassen

1. Mesh Vereinfachung

[10] Garland-Heckbert surface simplification

- schnell
- liefert gute Approximation



Als Abschätzung nur gröbste Stufe verwendet

2. Startnetz von Kurven

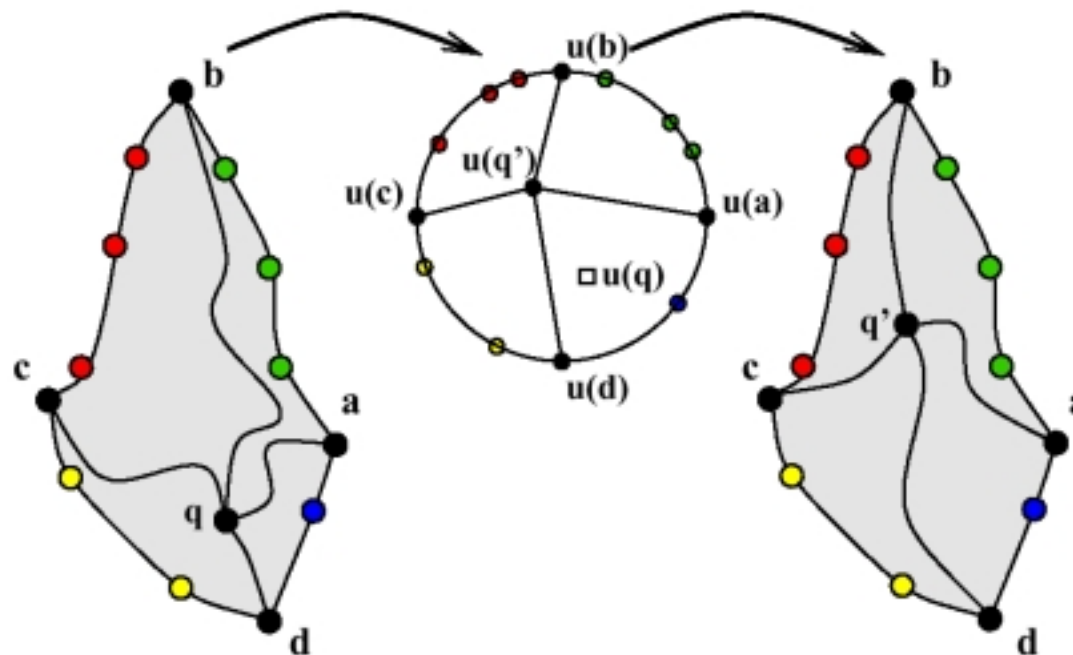


[15] MAPS: Parametrisierung von Meshes

- ⌘ liefert eine bijektives Mapping der Levels
- ⌘ die Kurven starten und enden bei Basis-Knoten
- ⌘ die Kurven bilden ein Dreiecks Patch

3. Globale Knoten fixieren

Basis-Mesh bestimmt qualität der Normal Mesh
Mesh aus Schritt 1 war Annäherung
Neubeurteilung von Basis-Knoten q



4. Globale Kanten fixieren

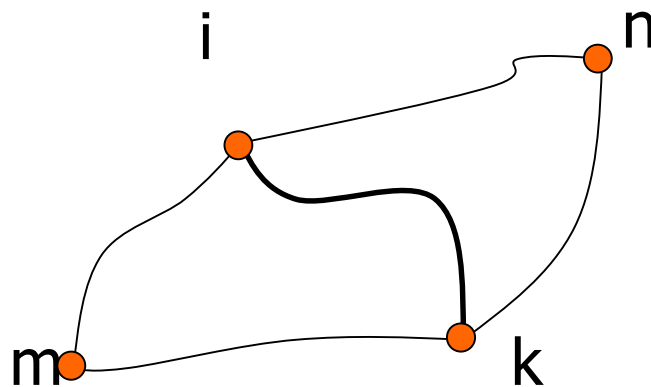
Parameterfunktion p berechnen, so dass diese die Region abdeckt

Bedingung:

$$p(i) = p(k) = 0$$

$$p(m) = 1$$

$$p(n) = -1$$

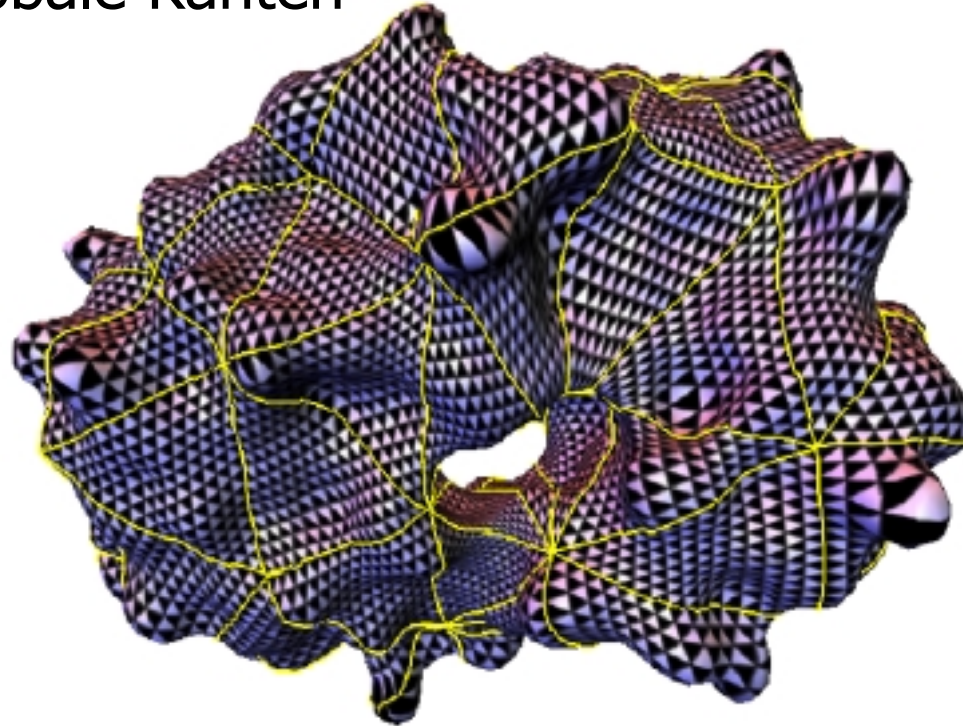


neue Kurve von i nach k bei $p()=0$

5. Initiale Parametrisierung

Parametrisierung der Patches zu Dreiecken

(gelb) Globale Kanten



6. Piercing

⌘ s Butterfly subdivision

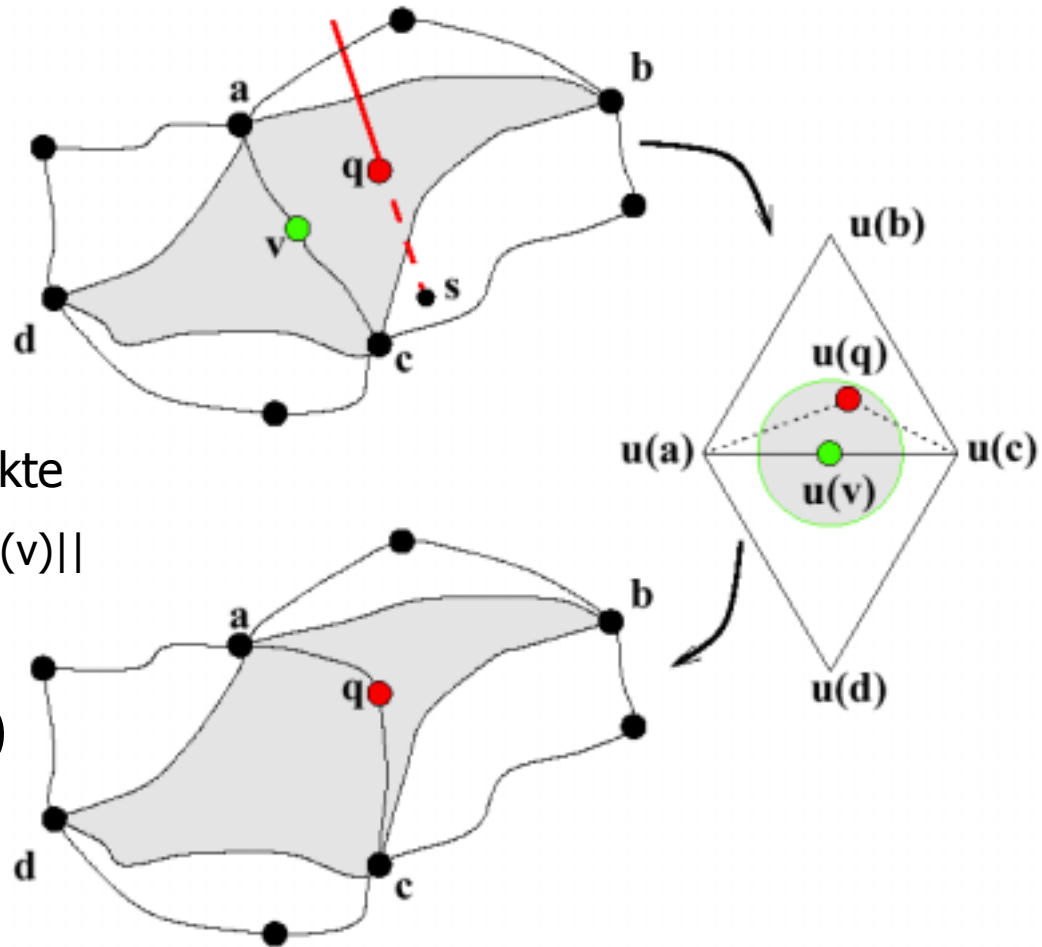
⌘ 2 Fälle

⊠ 0 Schnittpunkt

⊠ 1 und mehrere Schnittpunkte

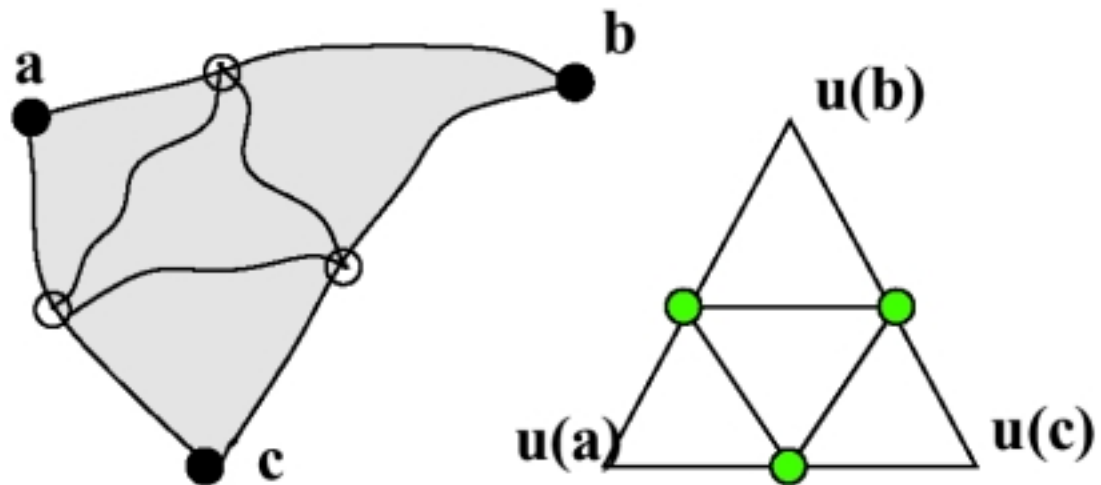
$$\boxtimes ||u(q)-u(v)|| < k||u(a)-u(v)||$$

⌘ Neue Kurve durch $u(q)$



7. Parametrisierung anpassen

- ⌘ Berechnung der Seitenmitten aus Stufe j
- ⌘ Neue Flächen für Stufe $j+1$ durch Quadrisektion mit Kurven



Quadrisektion des Dreieckspatch

Schritte

Zusammenfassung



1. Input Mesh -> Basis Mesh
2. MAPS: Basis Mesh - Input Mesh
3. Basisknoten überprüfen, anpassen
4. Basiskanten überprüfen, anpassen
5. Parametrisierung der Patches zu Dreiecken
6. Piercing
7. Parametrisierung anpassen.

Resultate 1

Dataset	Size	Base	Normal mesh size	Not normal (%)	% L^2 error	Time (min)
Feline	49864	156	40346	729 (1.8%)	.015	4
Molecule	10028	37	9521	270 (2.8%)	.075	1.5
Rabbit	16760	33	8235	196 (2.4%)	.037	2
Torus3	5884	98	5294	421 (8.0%)	.03	3
Skull	20002	112	25376	817 (3.2%)	.02	2.5
Horse	48485	234	59319	644 (1.1%)	.004	6.8

Zeiten gemessen auf Intel Pentium III 700MHz

Resultate 2



- ⌘ Basis Mesh darf beliebige Auflösung haben
 - ☒ Beim Schädel war es nur ein Tetraheder
- ⌘ gute Approximation der Rohdaten
- ⌘ effiziente Repräsentation des Objekte

Anwendungsgebiete



- ⌘ Kompression: bei Wavelets
- ⌘ Filter
- ⌘ Texturierung
- ⌘ Rendering mit Normal Maps

Persönliche Beurteilung



- ⌘ Interessanter Ansatz, Einsparung von 2 Floatwerten
- ⌘ Man erhält regelmässige Form von Dreiecken
- ⌘ Mesh simplification mit Garland-Heckbert und nicht mit MAPS?

Fragen beantworten.



Fragen?