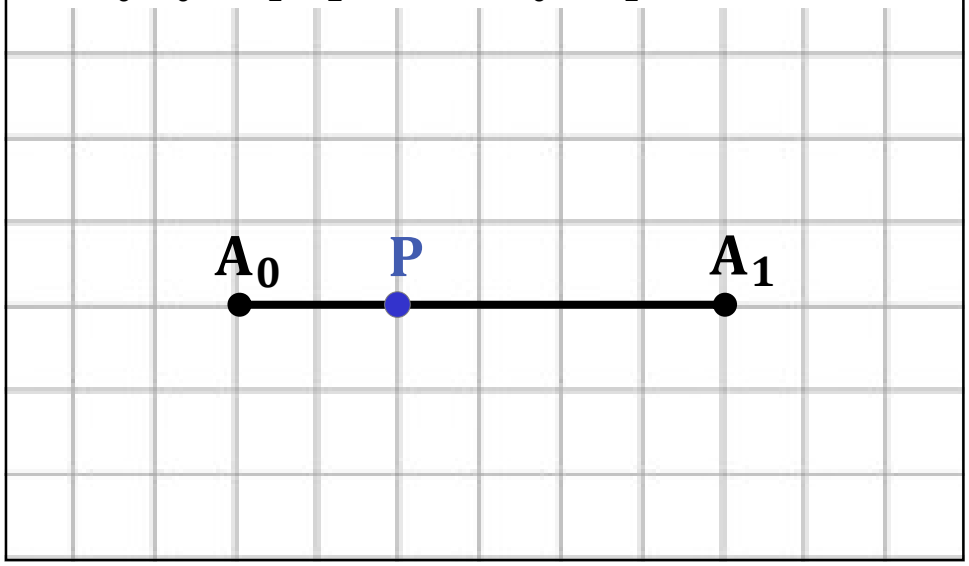
Coordinate baricentriche del punto P nel segmento?

Cioè: la coppia di valori (b_0, b_1) tali che…

$\mathbf{P} = b_0\,\mathbf{A_0} + b_1\,\mathbf{A_1}$, con $b_0 + b_1 = 1$

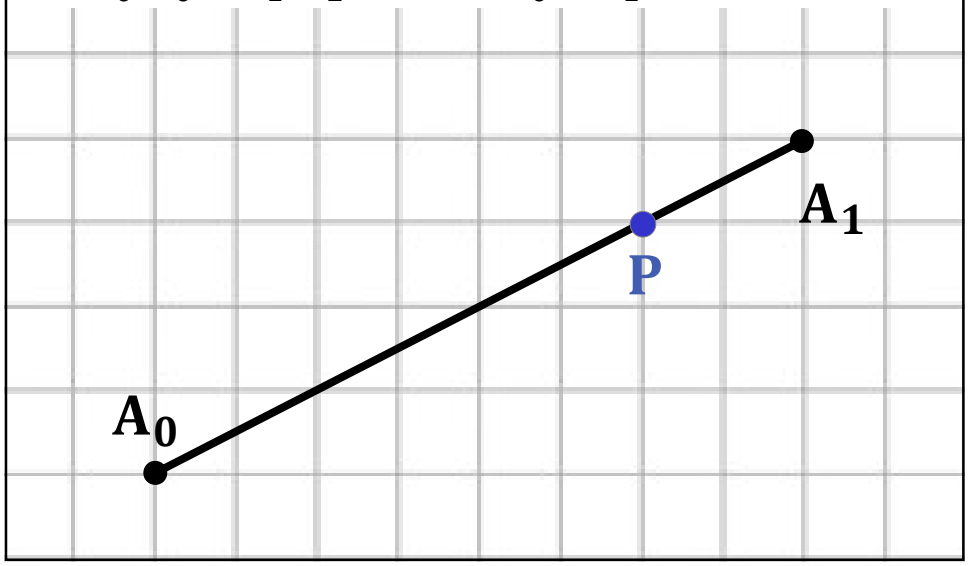


44

Coordinate baricentriche del punto P nel segmento?

Cioè: la coppia di valori (b_0, b_1) tali che…

$\mathbf{P} = b_0\,\mathbf{A_0} + b_1\,\mathbf{A_1}$, con $b_0 + b_1 = 1$

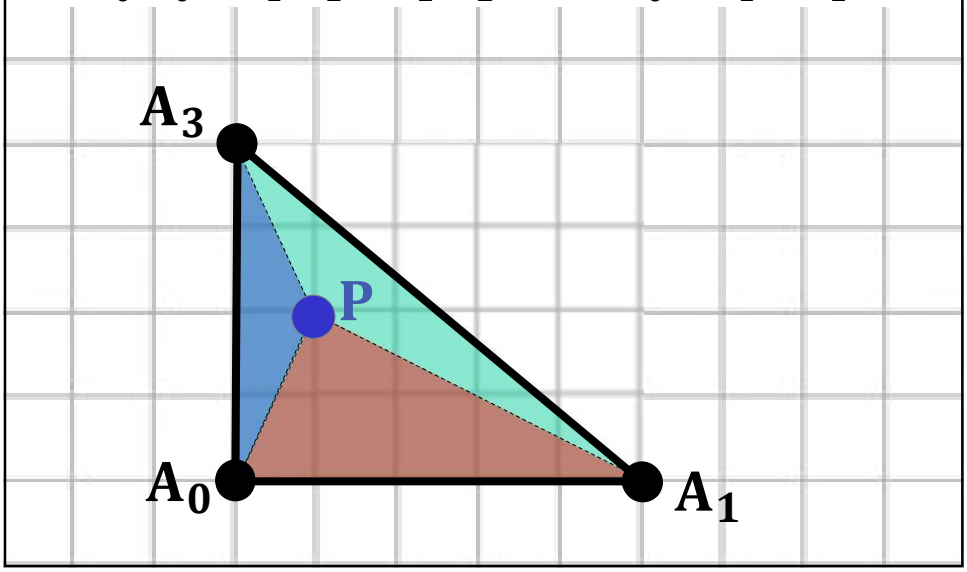


45

Coord. baric. di questo punto P in questo triangolo?

Cioè: la terna di valori (b_0, b_1, b_2) tali che…

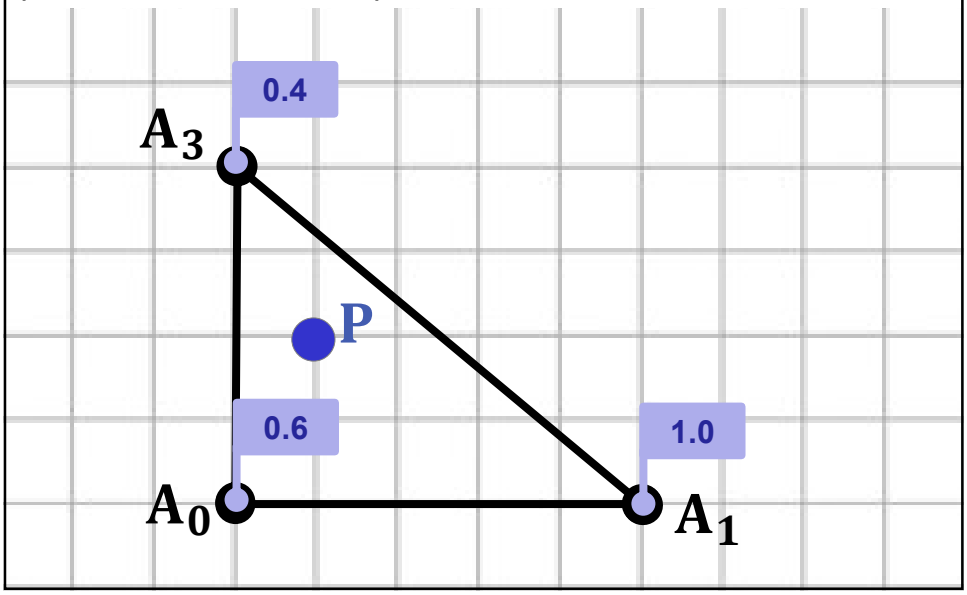
$\mathbf{P} = b_0\,\mathbf{A_0} + b_1\,\mathbf{A_1} + b_2\,\mathbf{A_2}$ con $b_0 + b_1 + b_2 = 1$



46

Coord. baric. di questo punto P in questo triangolo?

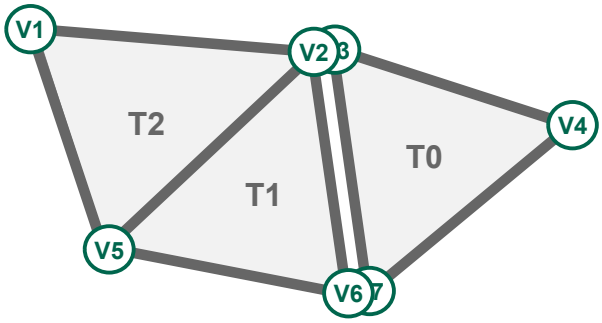
Se assegno questi 3 valori all'attributo per vertice "lucentezza", qual'è la "lucentezza" del punto P?



47

Attributi per vertice con discontinuità

- ✓ Se è necessario, è possibile modellare delle discontinuità (C0) di attributo lungo un edge, duplicando i vertici ai suoi estremi
  - ⇒ La mesh avrà due vertici geometricamente coincidenti (stessa posizione) ma con attributi associati differenti
  - ⇒ Due edge tecnicamente aperti saranno quindi perfettamente sovrapposti, causando una superficie continua (senza «spiragli»)
  - ⇒ Facce adiacenti useranno una oppure l'altra di questa coppia di vertice



- ⇒ Questa configurazione si chiama un «seam» (letteralmente: cucitura) o «vertex seam»

48

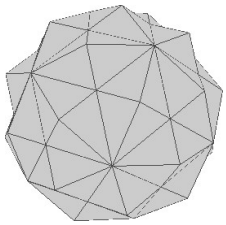
Attributi: il caso delle normali

Nel caso in cui l'attributo è la normale
(cioè il *vettore unitario ortogonale alla superficie*)

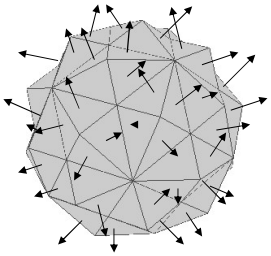
- ✓ se memorizzato per faccia:
  - ⇒ la normale costante su ciascuna faccia
  - ⇒ facce (dall'aspetto) piatto
  - ⇒ discontinuità C0 sugli edge:
    edge sono «spigoli taglienti»,
    detti edge di crease, o «hard» edges
- ✓ se memorizzato per vertice
  - ⇒ normale che varia sulla faccia
  - ⇒ facce (dall'aspetto) curvo, in generale
  - ⇒ continuità C0: → aspetto «smooth» (liscio),
  - ⇒ (ma non continuità C1)

49

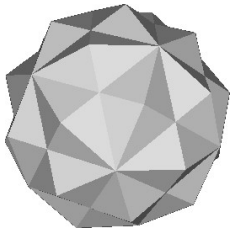
Normale come attributo per faccia



mesh di partenza



per ogni faccia,
calcolo normale

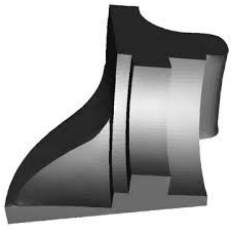


rendering
risultante
(detto "**flat shading**")

50

Normali di una superficie: osservazioni

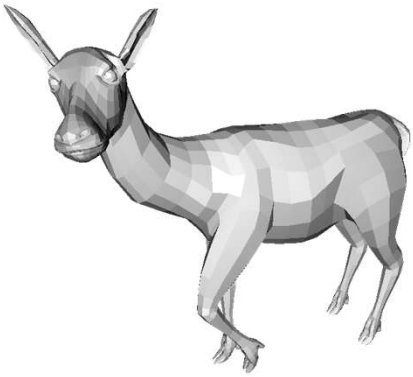
- ✓ Normale costante => superficie *piatta*
- ✓ Normale che varia con continuità => superficie *curva*
- ✓ Normale che varia con discontinuità => un ***crease***
  - ⇒ un *crease* è uno spigolo vivo, una linea dove la normale è discontinua
  - ⇒ anche detto ***hard-edge***, o ***line feature***



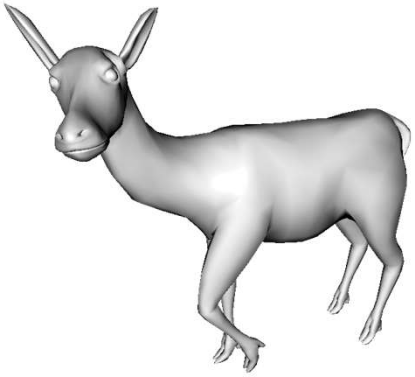
51

Normali come attributi

Normali per faccia



Normali per vertice



Nota: le normali sono rivelate all'occhio dall'*illuminazione*
(interazione con la luce, computata durante del rendering)

52

Normali come attributi per vertice di una mesh

- ✓ Normali definite per faccia:
  - ⇒ Le normali sono costanti sulle facce:
  - ⇒ Le facce appaiono piatte
  - ⇒ coerentemente col modello digitale, ma a differenza (spesso) dell'oggetto 3D che si intendeva rappresentare!
  - ⇒ Appaiono piccoli crease su ogni edge della mesh
- ✓ Normali definite per vertice:
  - ⇒ Le normali variano sulle facce (vengono interpolate dentro le facce, come qualsiasi altro attributo)
  - ⇒ Le facce appaiono in generale curve
  - ⇒ Come ottengo una faccia che appaia piatta? Uso una stessa normale come attributo dei tre vertici di un triangolo
  - ⇒ Come ottengo un crease (una discontinuità di normale)? (vedi next)
- ✓ Nota: le normali risultano visibili all'osservatore attraverso l'illuminazione del modello digitale
  - ⇒ Il calcolo dell'illuninazione è un task del rendering (2nda metà del corso)

53

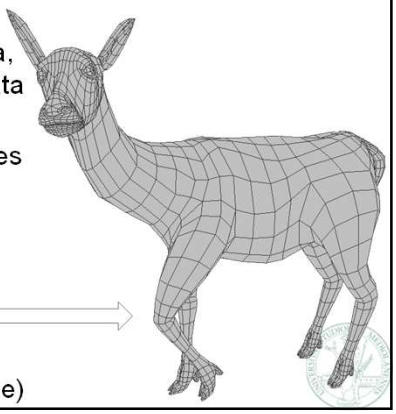
Interpolare l'attributo per vertice "vettore normale"

- ✓ Come abbiamo visto, interpolando vettori unitari si ottengono vettori non unitari
  - ⇒ Che vanno quindi ri-normalizzati
- ✓ Questo significa che gli attributi di tipo *vettore unitario*, come le normali, devono essere rinormalizzati dopo ogni l'interpolazione
- ✓ Esercizio: determinare con una stima se questo comporta un onere di calcolo eccessivo, per una GPU, durante un rendering in real-time
- ✓ Traccia
  - ⇒ stimare quante operazioni in virgola mobile (Floating-point Operation) sono necessarie per normalizzare un vettore (soluzione: circa 10)
  - ⇒ stimare quante volte sarà necessario ri-normalizzare un attributo «normale» in un rendering (stimiamo, a braccio: 1 volta in ogni pixel, quindi circa 1000x1000 = 10^6)
  - ⇒ frame per secondo, in un rendering in tempo reale: stimiamo 60
  - ⇒ totale **FL**oating-point **OP**eration per **S**econdo (FLOPS) necessari a questo passaggio: 10 x 10^6 x 60 = 6x10^8 = 0.6 Giga-FLOPS
  - ⇒ Una GPU in commercio è capace di erogare… Tera-FLOPS = migliaia di Giga-FLOPS

54

Note su rendering delle mesh

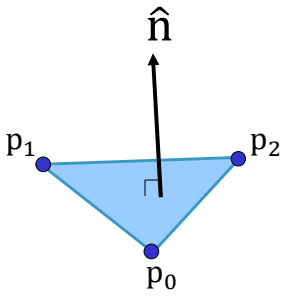
- ✓ L'uso di normali come attributo *per faccia* produce un tipo di rendering detto **flat shading**
  - ⇒ Perché le facce appaiono piatte (flat)
  - ⇒ Questo può essere utile per rivelare la forma poligonale delle mesh (ad esempio per poter giudicare la qualità della triangolazione)
- ✓ L'uso di normali come attributo *per vertice* produce un tipo di rendering detto **smooth shading**
  - ⇒ Perché la superficie appare liscia e curva, nascondendo la sua natura poligonalizzata
  - ⇒ Questo è di gran lunga in metodo più utilizzato, per esempio nei videogames
- ✓ Un tipo di rendering che rivela ancora più chiaramente la poligonalizzazione della mesh è detto **wireframe**
  - ⇒ Nel quale vengono disegnati esplicitamente anche gli edge (come linee)



55

Computo della normale di un triangolo

(Cioè del suo orientamento nello spazio)
(Vedi lezione nella marte matematica sul cross product)



due cosiddetti "edge vectors"

$$\vec{n} = (p_1 - p_0) \times (p_2 - p_0)$$

"area vector" definitp come
un vettore ortogonale al triangolo,
che è lungo quanto la (doppia) area del triangolo

normalizzazione

$$\hat{n} = \frac{\vec{n}}{\|\vec{n}\|}$$

"normale" del triangolo
(un vett unitario)

E' anche la
doppia area
del triangolo

56

Calcolo delle normali per faccia di una mesh (note)

- ✓ La normale delle facce triangolari è data dal semplice computo visto sopra
  - ⇒ ripetuto su ciascuna faccia
- ✓ E' anche detta normale «geometrica» della faccia
  - ⇒ perché corrisponde effettivamente all'orientamento del piano su cui giace il triangolo (ed è costante)
  - ⇒ esiste sempre (nota: questo non vale per facce quads – a meno che non siano planari)
  - ⇒ un'eccezione: se ho triangoli «degeneri» (quelli con area 0) (con 3 vertici allineati o, 2 vertici coincidenti, etc) cosa succede se tento l'algoritmo qui sopra?
- ✓ La normale per faccia è orientata consistentemente (nello stesso verso) solo se la mesh è ben orientata
  - ⇒ (cosa succede altrimenti?)
  - ⇒ normale verso l'interno oppure l'esterno? Dipende della formula che uso per il suo computo (quali due edge-vector scelgo e in che ordine li moltiplico)

57

Calcolo delle normali per vertice di una mesh (note)

- ✓ La normale per vertice di una mesh non è definita in modo univoco per via geometrica
  - ⇒ quindi dobbiamo «inventarcene» una.
- ✓ La domanda che ci dobbiamo porre è:
  - ⇒ «immaginando che la mia mesh (che è composta da facce *piatte*) modelli invece una superficie reale *curva*, quale sarebbe la normale di questa superficie curva, su questo vertice?»
  - ⇒ Non esiste una risposta univoca!
  - ⇒ La risposta dipende da quale superficie si intende rappresentare.
- ✓ Strategia spesso utilizzata in pratica:
  - ⇒ usare una interpolazione delle facce adiacenti (per es, la media)
- ✓ Cioè: *le normali (per vertice) fanno parte del modello* (sono parte della descrizione della superficie) e spesso vengono costruite insieme al modello.
  - ⇒ Solo nei casi di una mesh sprovvista di normali, sarà necessario computarle dalla geometria (e dalla connettività)

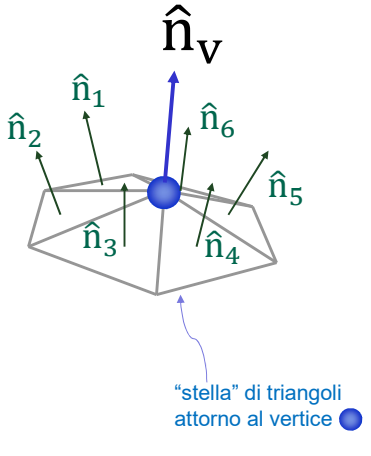
58

Normali come attributo per vertice

Un modo per calcolare di un vertice condiviso da k triangoli: la media delle k normali definite sui triangoli

k è detta la valenza del vertice, (nel disegno, k = 6)

$$\hat{n}_v = \frac{\hat{n}_1 + \cdots + \hat{n}_6}{\|\hat{n}_1 + \cdots + \hat{n}_6\|}$$



Nota: dato che si normalizza comunque il risultato della sommatoria, non c'è necessità di dividere per il numero di elementi sommati, come si fa di solito per la media

59