

Számítógépes Grafika

2017/2018. II. félév

Animáció

- Állókép helyett képsorozat
- Objektumok/kamera/világ tulajdonságait változtatjuk
- Egy kép \equiv egy idő pillanat
- Képsorozat elég gyors \rightarrow folyamatos mozgást érzékel az emberi szem

Tartalom

- 1 Animáció
- 2 Animálható tulajdonságok
- 3 Animáció típusok
- 4 Hierachikus rendszerek

Mi az, amit változtatunk?

- Lehet: pozíció, orientáció, szín, normál, BRDF, stb.
- "Értelme" van: pozíció, orientáció, kamera
- Animációt alkalmazunk:
 - a modellezési transzformációra
 - a kamera transzformációra

- A set of navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ 🔍 ↺

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ▶ ↺ 🔍 ↻

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ 🔍 ↺

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ▶ ↺ 🔍 ↻

1 Animáció

- Kamera animáció
- Pozíció és orientáció
- Roll, Pitch, Yaw

4 Hierarchikus rendszerek

- Pozíció: *"Hol van az objektum?"*
- Orientáció: *"Hogy áll, merre fele néz az objektum?"*
- Elsősorban ezt a kettőt szerenénk változtatni.
- $M_o(t)$ megadja mindkettőt.
- Normális esetben

$$M_O(t) = \begin{bmatrix} A_{11} & A_{12} & A_{13} & 0 \\ A_{21} & A_{22} & A_{23} & 0 \\ A_{31} & A_{32} & A_{33} & 0 \\ p_x & p_y & p_z & 1 \end{bmatrix},$$

- $\vec{p} = (p_x, p_y, p_z)$ a pozíció.
- Az **A** mátrix **tartalmazza** az orientációt.

- szempozíció (*eye*),
- egy pont amire néz (*center*),
- felfele irányt megadó vektor (*up*),
- a képernyő/ablak oldal aránya (*aspect*),
- nyílásszög (*fovy*).

- A set of small navigation icons typically found in Beamer presentations, including symbols for back, forward, search, and other slide controls.

- Tegyük \vec{p} -t és \mathbf{A} -t is időfüggővé!
- \vec{p} tagjait leírhatjuk külön-külön függvényekkel.
- *Pl.: Valami esik, elég p_y -t változtatni.*
- \mathbf{A} -t tagjai összefüggenek, és csak az orientáció érdekel belőle minket (nem érdekel: méretezés, nyírás).
- Az orientáció megadható három tengely menti forgatással
→ három független függvényekkel.

Tartalom

- 1 Animáció
- 2 Animálható tulajdonságok
- 3 Animáció típusok
 - Képlet animáció
 - Kulcskocka animáció
 - Pálya animáció
- 4 Hierarchikus rendszerek

Kulcskocka (*key frame*) animáció

- Egy bonyolult mozgást nehézkes lenne képlettel megadni.
- Inkább adjuk csak bizonyos időközönként, hogy *akkor* mit szeretnénk látni.
- Ezek a kulcskockák.
- Egy tulajdonságot két kulcskocka között *interpolációval* számolunk ki.

Lineáris interpoláció

- Legyen a két kulcskockánk időpontja t_0 és t_1 .
- Legyen az interpolálandó tulajdonság g .
- Lineáris interpolációval $\forall t \in [t_0, t_1]$ -re kapjuk

$$g(t) = \left(1 - \frac{t - t_0}{t_1 - t_0}\right) g(t_0) + \frac{t - t_0}{t_1 - t_0} g(t_1)$$

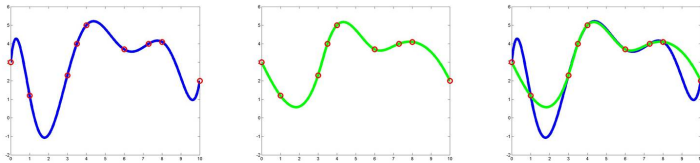
Polinom interpoláció

V: Ritkán néz ki természetesen. Animáció során a sebesség konstans, előtte, utána nulla.

- n kulcspontra fel tudunk írni $n - 1$ -ed fokú polinomot.
- Interpolációs polinom: minden kulcskockában az előírt értéket veszi fel.
- Együtthatók számíthatók Lagrange interpolációval.
- A lineáris interpoláció a Lagrange interpoláció speciális esete $n = 2$ -re.

Spline interpoláció

- A polinom interpolációval kapott fv. magas fokszám esetén a szomszédos pontok között "hullámszik", így elrontja az animációt.
- Spline interpoláció: használjunk több, egymáshoz kapcsolódó, alacsony fokszámú polinomot az interpolációhoz!



Pálya animáció

- Egy objektum mozgását megadhatjuk a bejárandó pálya megadásával is.
- A pályát egy 3D görbével adjuk meg.
- A model ezen a görbén halad végig.

Orientáció megadása

- Hogyan adjuk meg az objektumunk orientációját?
- Egy *előre*, és egy *felfele* irány egyértelműen meghatározza ezt.
- *Megjegyzés: v.ö. kamera esetén center-eye ill. up vektorok*
- Ha a pályagörbe differenciálható, akkor az megadja a sebességvektort minden időpillanatban.
- A sebességvektor mindig előre fele mutat.

Orientáció megadása

- A *felfele* irány megadására két lehetőségünk is van.
- Ha van egy természetes *felfele*, akkor használjuk azt. (Mindennél, ami nem ből be a kanyarban.)
- Ha ez az irány is változik, akkor ez megegyezik a gyorsulás irányával, azaz a pályagörbe második deriváltjának irányával.

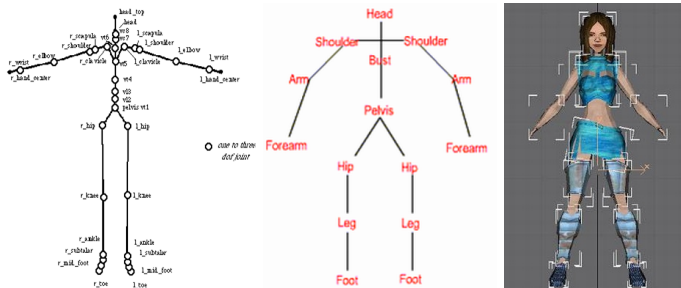
Tartalom

- 1 Animáció
- 2 Animálható tulajdonságok
- 3 Animáció típusok
- 4 Hierarchikus rendszerek
 - Előrehaladó kinematika
 - Inverz kinematika

Hierachikus rendszerek

- Szintér gráfoknál már találkoztunk ilyenekkel.
- Egy gyerek objektum mozgását a szülőhöz viszonyítva adjuk meg.
- Gyerekeknek lehetnek újabb gyerekei, s.i.t.
- Hierachikus rendszert – fát – kapunk.

Példa: Emberi test



Kényszerek (*constraints*)

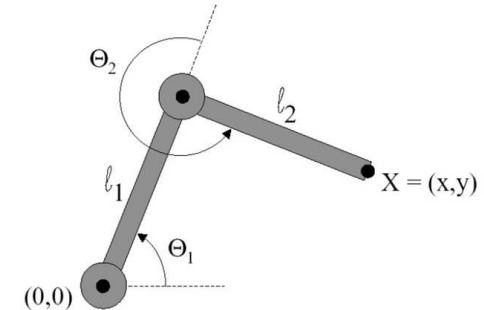
- Nem minden mozgást szeretnénk megengedni a szülőhöz képest.
- Ezeket a megszorításokat írhatjuk le *kényszerekkel*.
- Korlátozhatjuk a szabadságfokokat: pl. könyök csak egy tengely mentén tud forogni, de a csukló kettő
- Vagy a tartományokat: kevesen bírják, ha a fejük 90° -nél többet fordul.

Előrehaladó kinematika

- Végállapotot határozunk meg az állapotváltozók függvényében.
- Szimulációkhoz jól használható.
- Minden elemre megadjuk, a hozzá tartozó transzformációt.
- Ezeket a hierarchiában felülről lefele haladva értékeljük ki.
- Az adott elemhez tartozó transzformáció az összes ő és a saját transzformáció szorzata.

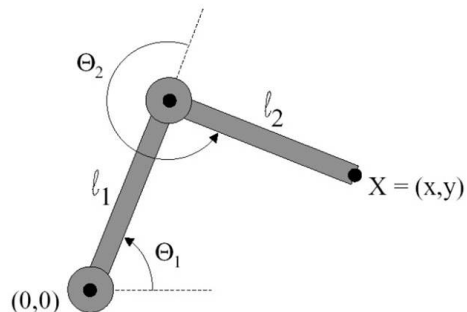
Példa

- Kétszabadságfokú, rotációs csuklókat tartalmazó rendszer.
- A csuklók csak a Z tengely körül fordulnak.



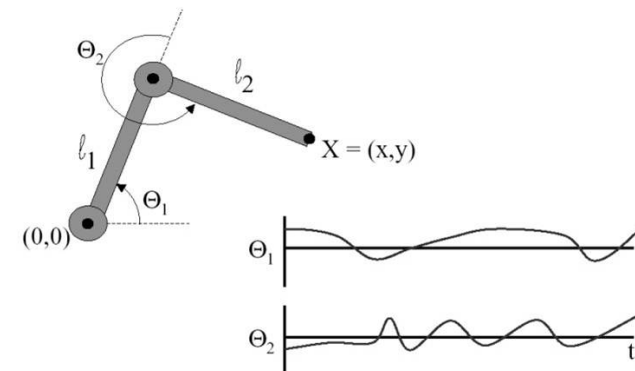
Példa – folyt.

- Állapotváltozók: Θ_1, Θ_2
- A végberendezés (X) pozícióját a gép számolja.
- $X = (l_1 \cos \Theta_1 + l_2 \cos(\Theta_1 + \Theta_2), l_1 \sin \Theta_1 + l_2 \sin(\Theta_1 + \Theta_2))$



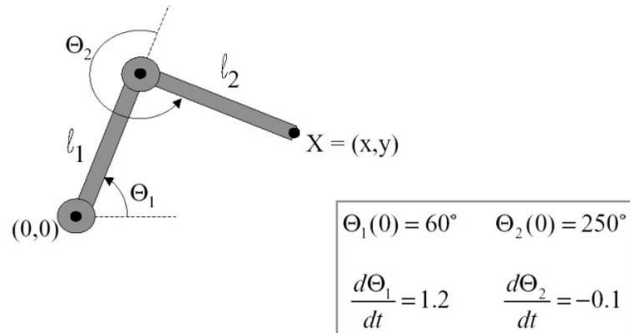
Példa – folyt.

- Az állapotváltozókat megadhatjuk (pl. spline) függvénnel.



Példa – folyt.

- Az állapotváltozókat megadhatjuk kezdeti értékkel és sebességgel.



Inverz kinematika

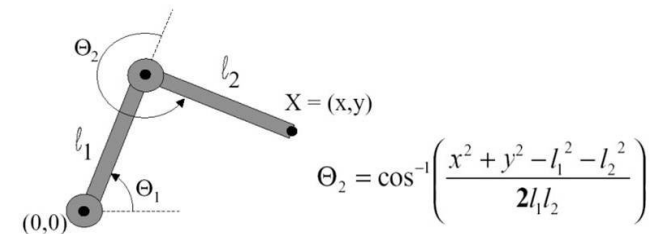
- Az inverz kinematika a kritikus végberendezés helyzetét interpolálja, majd az állapotváltozók értékét végberendezés interpolált helyzetéből számítja vissza.
- Az inverz kinematika másik neve a cél-orientált animáció.
- "Ezt szeretném megfogni, hogyan forgassam az ízületeimet?"

Mit nem tud az előrehaladó kinematika?

- Az előrehaladó kinematika nem használható, ha a strukturális összefüggés erősen nem lineáris
- Hiába interpolálunk egyenletesen az állapottérben, a végberendezés vadul kalimpálhat a kulcspontok között
- Problémás esetek:
 - Láb mozgása a talajon
 - Végállapot jó, de menet közben a berendezés részei átmehetnek egymáson.

Példa

- A végberendezés helyzetéből visszaszámoljuk az állapotváltozók értékét.

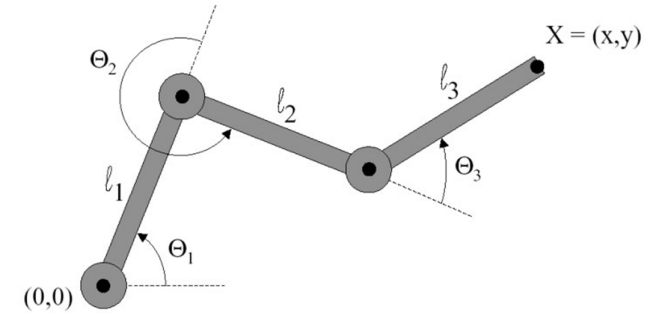


Problémák

- Nehéz "természetesnek látszó" mozgást leírni vele.
- Az inverz függvény kiszámítása nem triviális,
- és nem is egyértelmű (redundancia).

Példa

- Egyenletek száma: 2, ismeretlen változók száma: 3 \Rightarrow Végtelen sok megoldás!
- Rendszer DOF > végberendezés DOF
- Az emberi csontváz kb 70 DOF!



Példa

Nem egyértelmű, ill. nem létező megoldás

