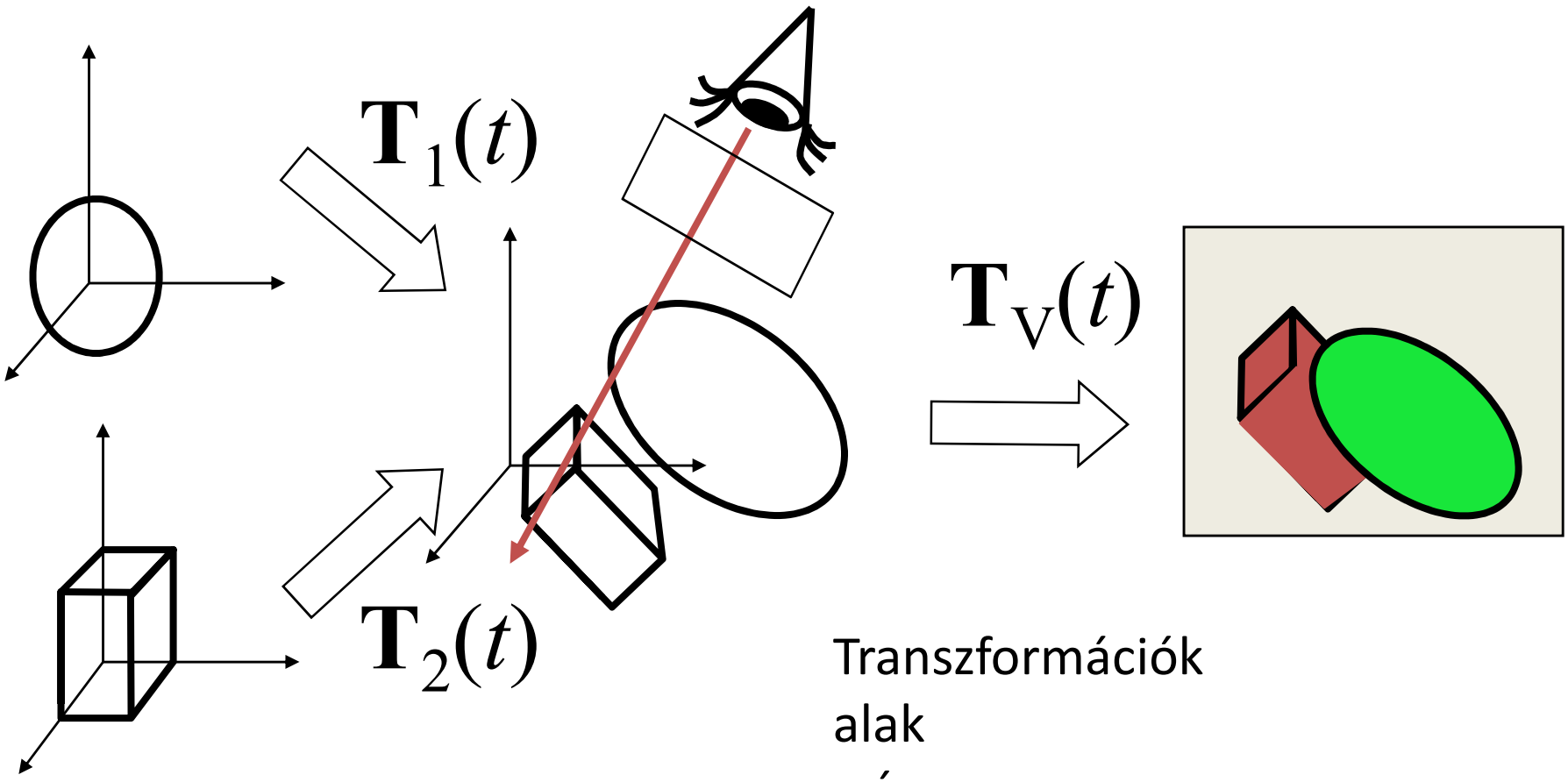
Animáció

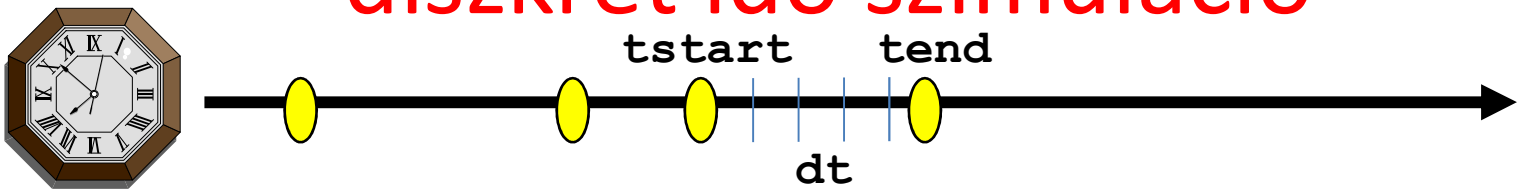
Szirmay-Kalos László

Animáció = időfüggés



Transzformációk
alak
szín
megjelenítési attribútumok, stb

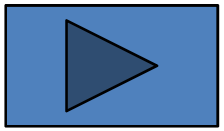
Valós idejű animáció és diszkrét idő szimuláció



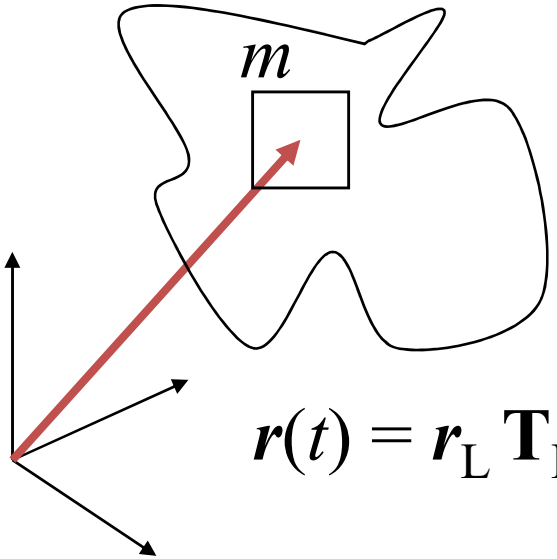

```
void IdleFunc( ) {          // idle call back
    static float tend = 0;
    const float dt = 0.01; // dt is ”infinitesimal”
    float tstart = tend;
    tend = glutGet(GLUT_ELAPSED_TIME)/1000.0f;

    for(float t = tstart; t < tend; t += dt) {
        float Dt = min(dt, tend - t);
        for (Object * obj : objects) obj->Control(Dt);
        for (Object * obj : objects) obj->Animate(Dt);
    }
    glutPostRedisplay();
}
```

Valószerű mozgás

- Fizikai törvények:
  - Newton törvény
    - Impulzus deriváltja az erő; Perdület deriváltja a forgatónyomaték
  - ütközés detektálás és válasz:
    - impulzus és perdület megmaradás
    - energia részleges megmaradás
- Fiziológiai törvények
  - csontváz nem szakad szét
  - meghatározott szabadságfokú ízületek
  - bőr rugalmasan követi a csontokat
- Energiafelhasználás minimuma



Newton törvény



$$\boldsymbol{r}(t) = \boldsymbol{r}_{\mathrm{L}}\,\mathbf{T}_{\mathrm{M}}(t)$$

$$\mathbf{F}/m = \frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{r}$$

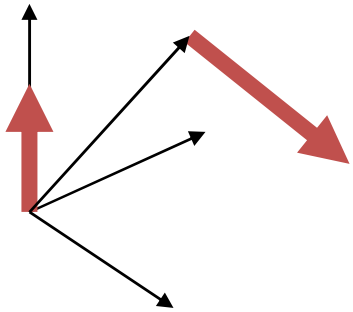
$$= \boldsymbol{r}_{\mathrm{L}}\frac{\mathrm{d}^2}{\mathrm{d}t^2}\mathbf{T}_{\mathrm{M}}(t)$$

Az erő rugalmas mechanizmuson keresztül hat, azaz folytonosan változik

$\mathbf{T}_{\mathrm{M}}(t)$ C^2 folytonos

$T_M(t)$: Mozgástervezés

- Követelmény: ált. C^2 , néha (C^1,C^0) folytonosság
- Mátrixelemek nem függetlenek
  - Tervezés független paraméterek terében



1. skálázás: *sx, sy, sz*
2. orientáció: *wx, wy, wz,* α
3. pozíció: *px, py, pz*

Tengely+szög

$\mathbf{p}(t)$

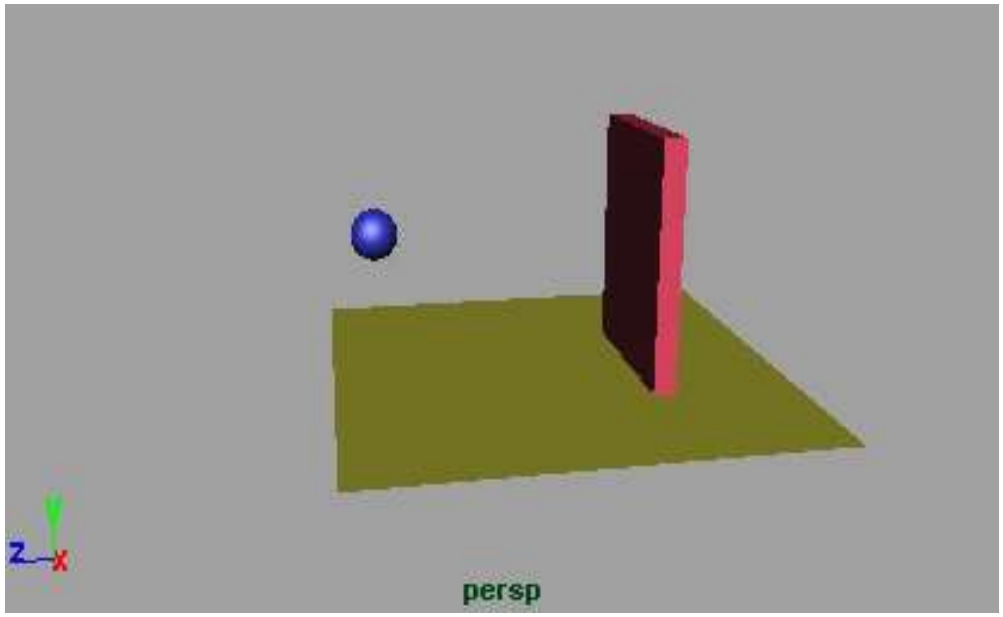
$$\mathbf{T}_M = \begin{bmatrix} sx & & & \\ & sy & & \\ & & sz & \\ & & & 1 \end{bmatrix} \begin{bmatrix} & & & \\ & \mathbf{R} & & \\ & & & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ px & py & pz & 1 \end{bmatrix}$$

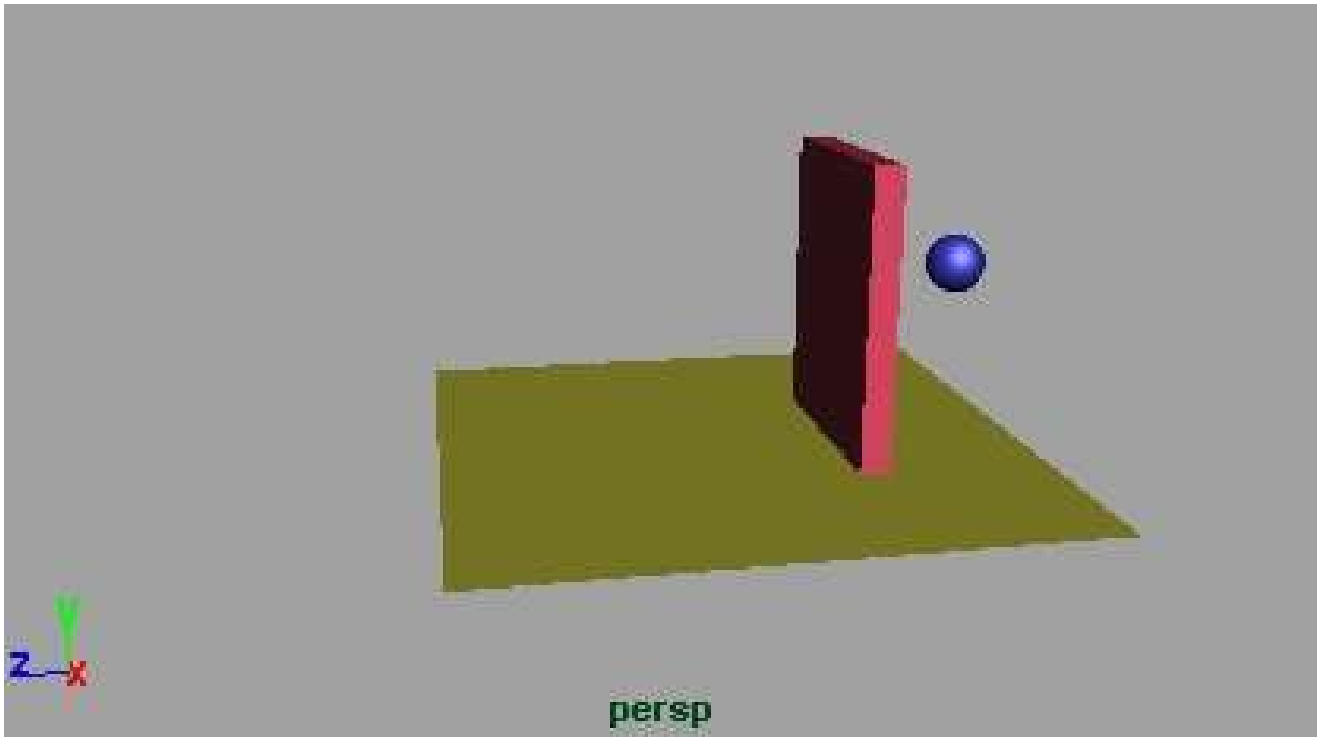
$$\boldsymbol{r}' = \boldsymbol{r}\cos(\alpha) + \boldsymbol{w}^0(\boldsymbol{r}\cdot\boldsymbol{w}^0)(1-\cos(\alpha)) + \boldsymbol{w}^0\times\boldsymbol{r}\sin(\alpha)$$

Mozgástervezés a paramétertérben

- **p**(t) elemei ált. C^2 , néha (C^1,C^0) folytonosak
- **p**(t) elemeinek a definíciója:
  - görbével direkt módon (**spline**)
  - képlettel: **script animation**
  - kulcsokból interpolációval**: keyframe animation**
  - görbével indirekt módon: **path animation**
  - mechanikai modellből az erők alapján: **physical animation**
  - mérésekből: **motion capture animation**

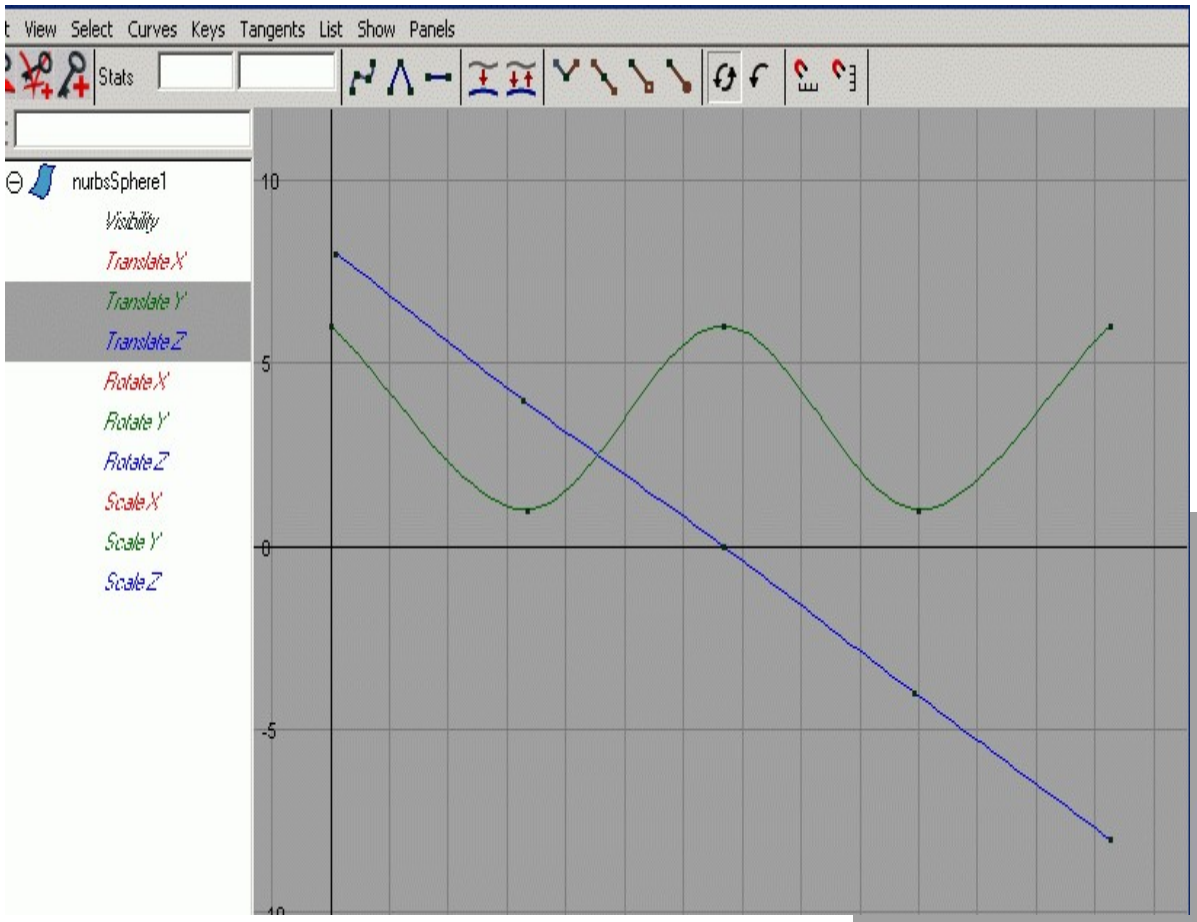
Keyframe animáció

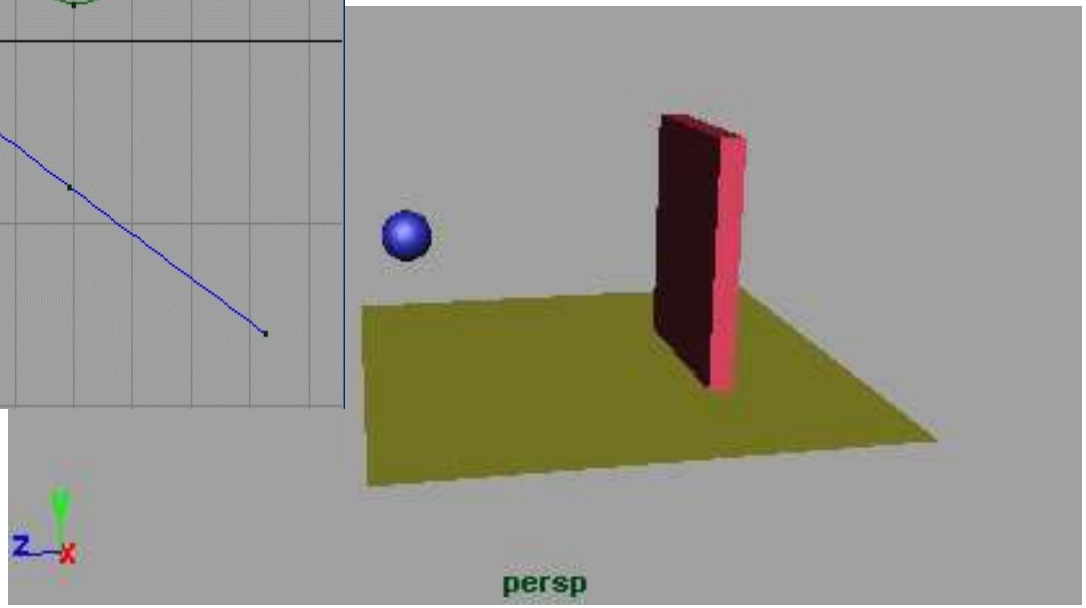




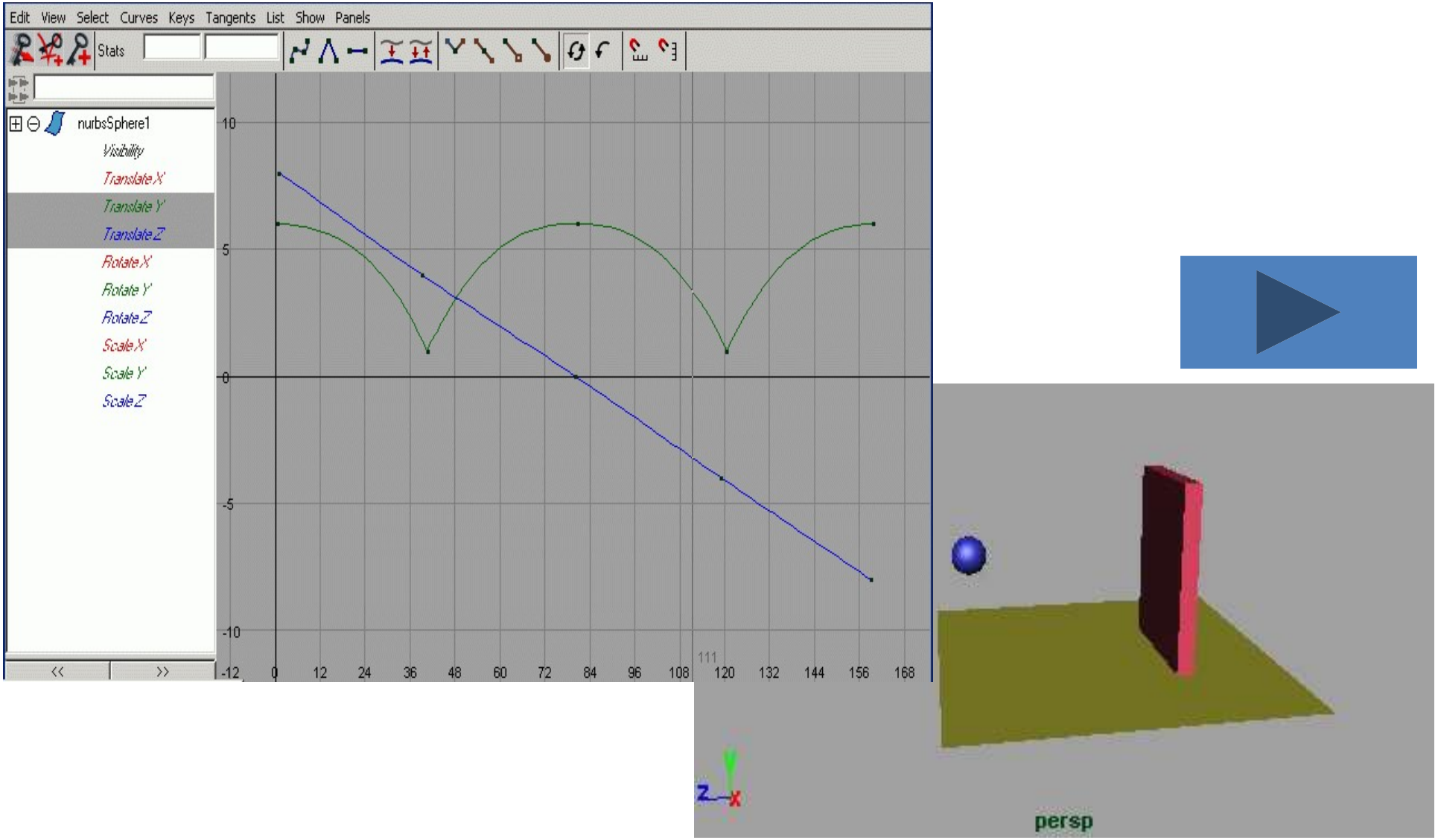
5.key

Keyframe animáció görbéi

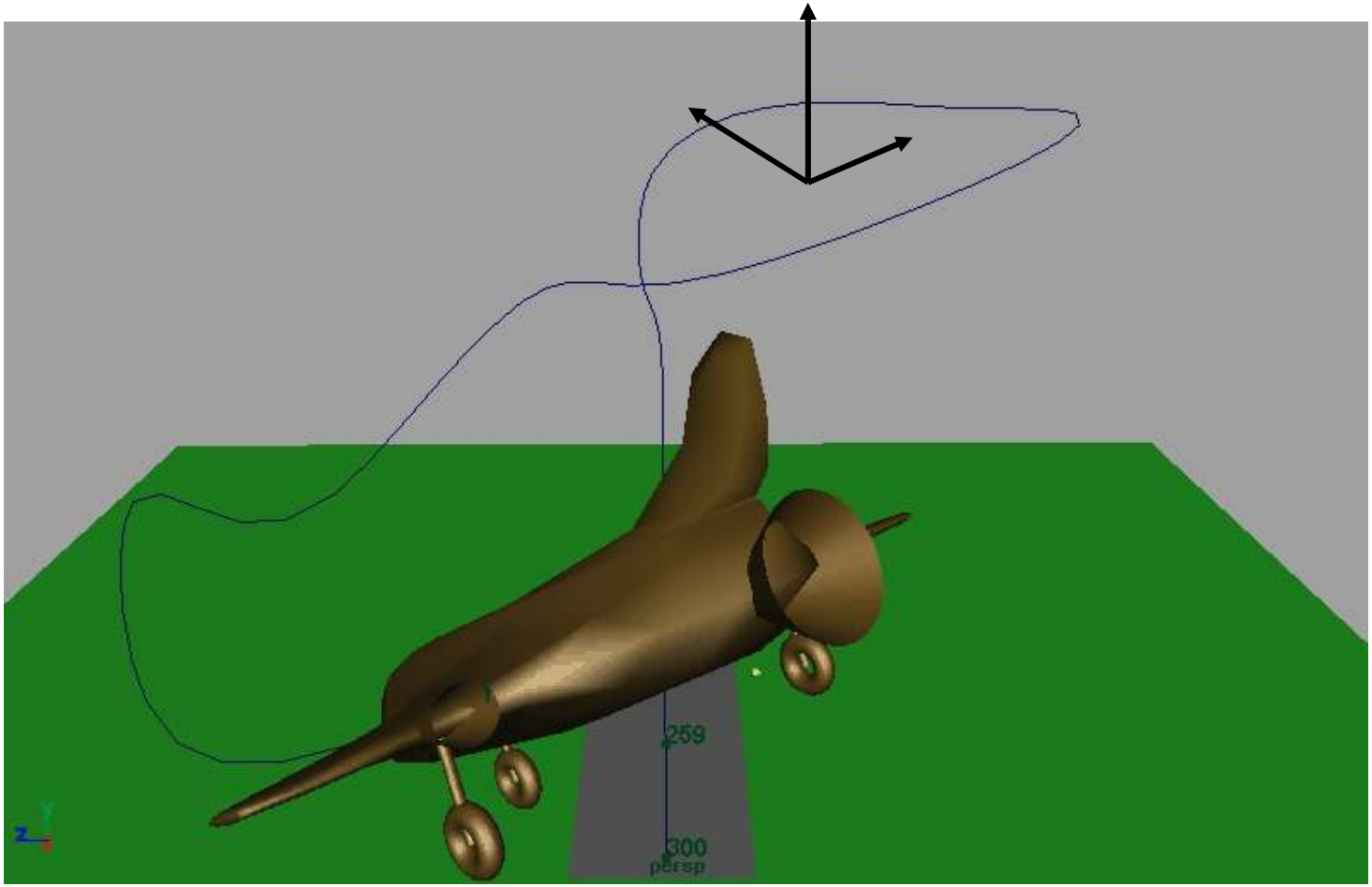




Görbék megváltoztatása

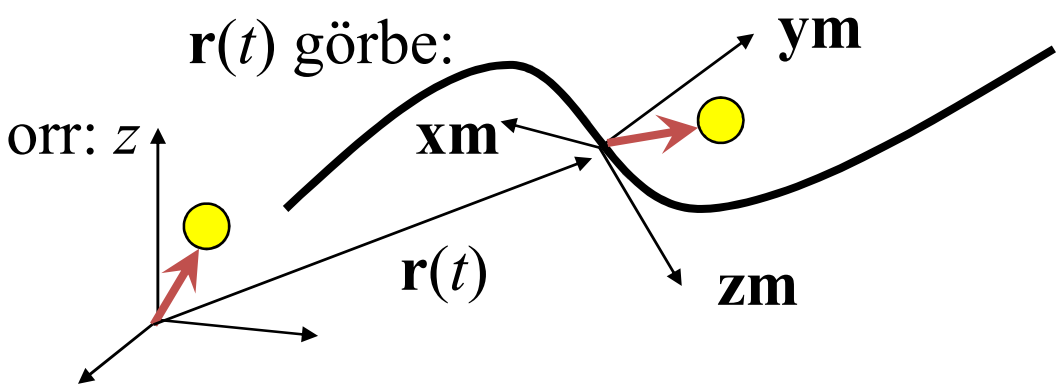


Pálya (path) animáció



t = spline paraméter vagy az ívhossz

Pálya animáció: Transzformáció



$$T_M = \begin{pmatrix} \mathbf{xm}^0(t) & 0 \\ \mathbf{ym}^0(t) & 0 \\ \mathbf{zm}^0(t) & 0 \\ \mathbf{r}(t) & 1 \end{pmatrix}$$

Explicit up vektor

$\mathbf{zm} = \mathbf{r}'(t)$

$\mathbf{xm} = \mathbf{zm} \otimes \text{up}$

$\mathbf{ym} = \mathbf{zm} \otimes \mathbf{xm}$

Frenet keretek:

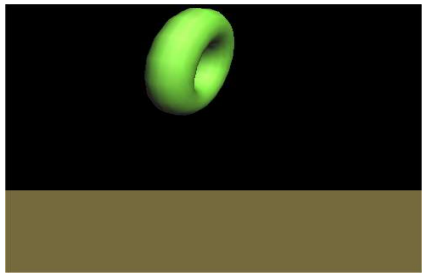
$\mathbf{zm} = \mathbf{r}'(t)$

$\mathbf{xm} = \mathbf{zm} \otimes \mathbf{r}''(t)$

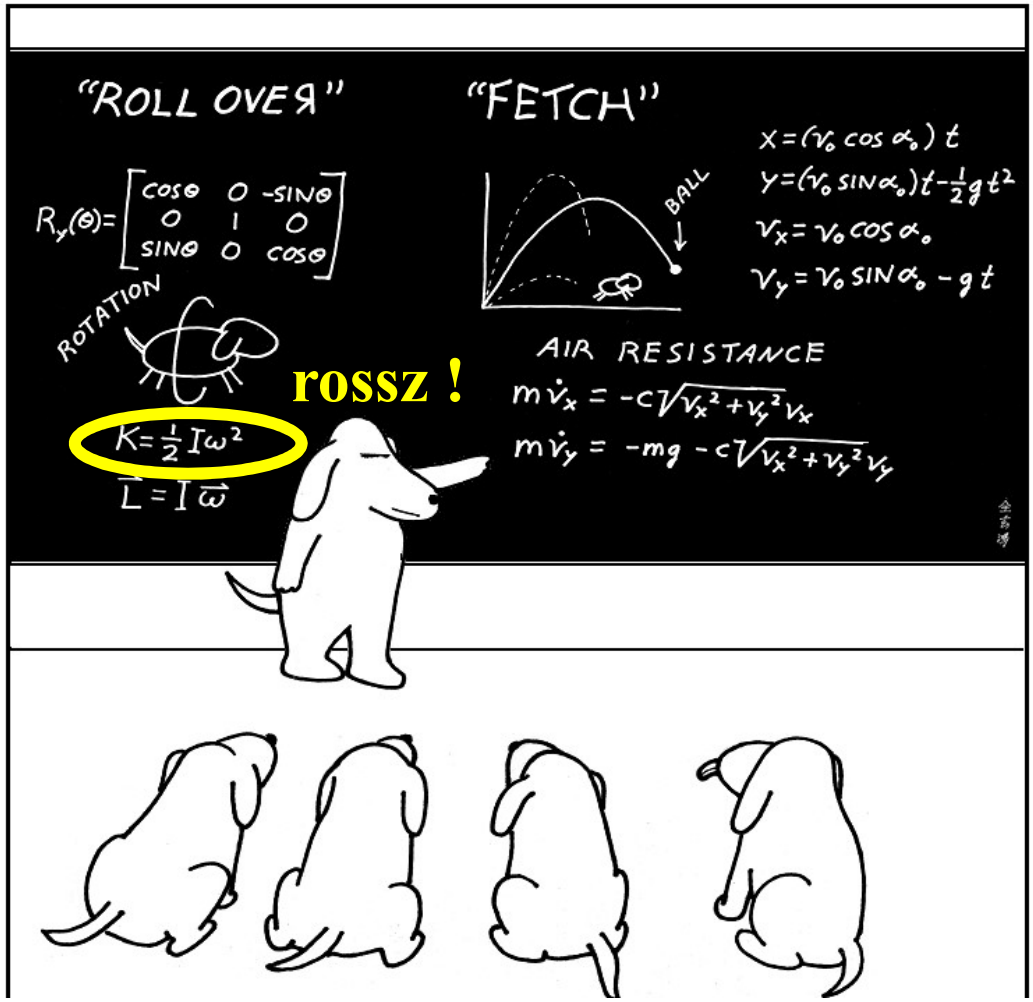
$\mathbf{ym} = \mathbf{zm} \otimes \mathbf{xm}$

A függőleges, amerre az erő hat

Fizikai animáció



- Erők (pl. gravitáció, turbulencia stb.)
- Tömeg, tehetetlenségi nyomaték (F = ma)
- Ütközés detektálás (metszéspontszámítás)
- Ütközés válasz
  - rugók, ha közel vannak
  - impulzus megmaradás

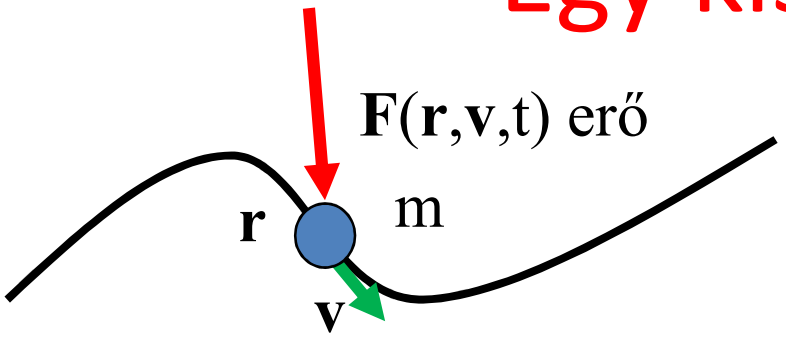


Haladó és forgó mozgás

Pozíció:	$\vec{r}$
Összefűzés:	$\vec{r}_1 + \vec{r}_2$
Sebesség:	$\vec{r}(t + \mathrm{d}t) = \vec{r}(t) + \vec{v}\mathrm{d}t$
Tömeg:	m
Lendület:	$\vec{p} = m\vec{v}$
Erő:	$\vec{F} = \frac{\mathrm{d}\vec{p}}{\mathrm{d}t}$
Energia:	$K = \frac{1}{2}mv^2$

Orientáció:	$\mathbf{R}$
Összefűzés:	$\mathbf{R}_1 \cdot \mathbf{R}_2$
Szögsebesség:	$\mathbf{R}(t + \mathrm{d}t) = \mathbf{R}(t)R(\lvert\vec{\omega}\rvert \mathrm{d}t, \vec{\omega})$
Tehetetlenségi nyom:	$\mathbf{I}$
Perdület:	$\vec{L} = \mathbf{I}\vec{\omega}$
Forgató nyom:	$\vec{M} = \frac{\mathrm{d}\vec{L}}{\mathrm{d}t}$
Energia:	$K = ?$

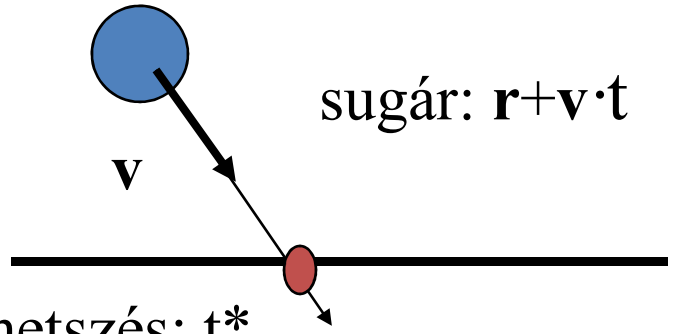
Egy kis mechanika



$$d\mathbf{r}/dt = \mathbf{v}$$
$$d\mathbf{v}/dt = \mathbf{F}(\mathbf{r},\mathbf{v},t)/m$$

```
for( t = 0; t < T; t += dt) {
    F = Eredő erő(r, v)
    a = F/m
    v += a ·dt
    if ( ÜtközésDetektál )
        ÜtközésVálasz
    r += v ·dt
}
```

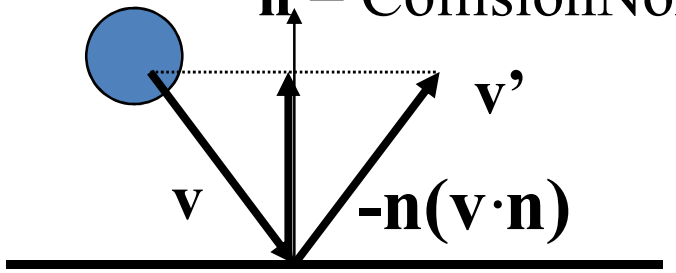
ÜtközésDetektál



metszés: t*

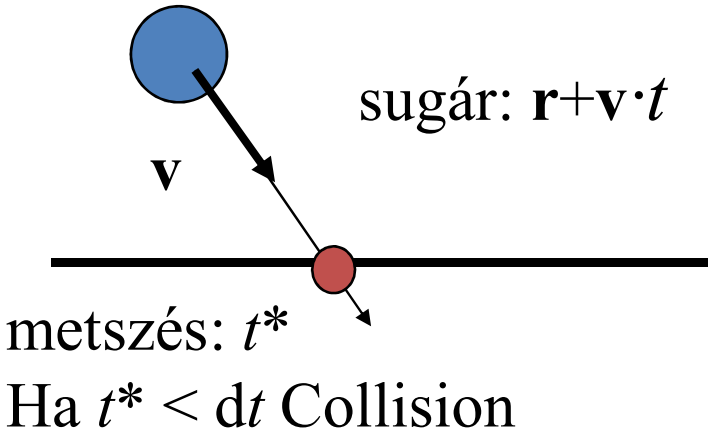
Ha t* < dt Collision

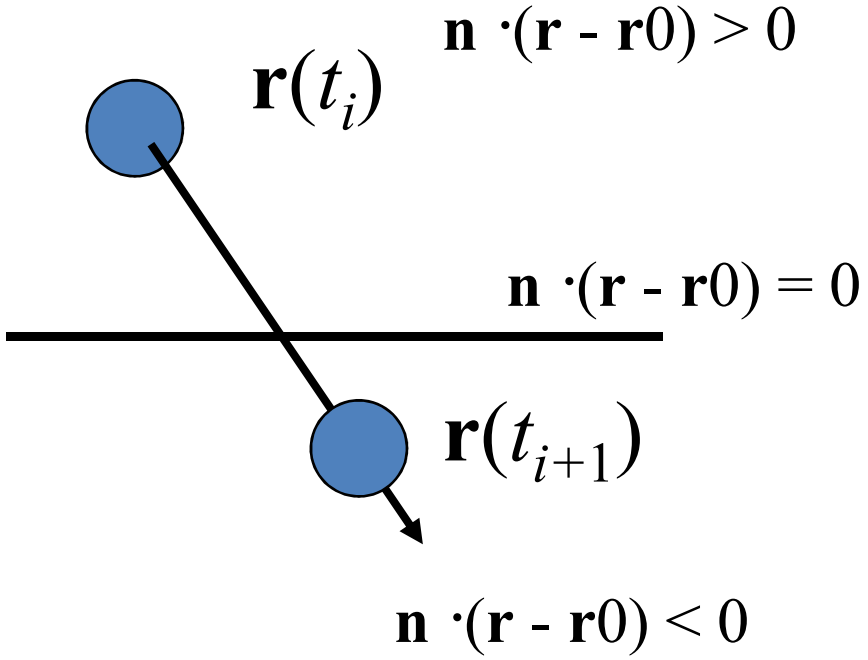
ÜtközésVálasz



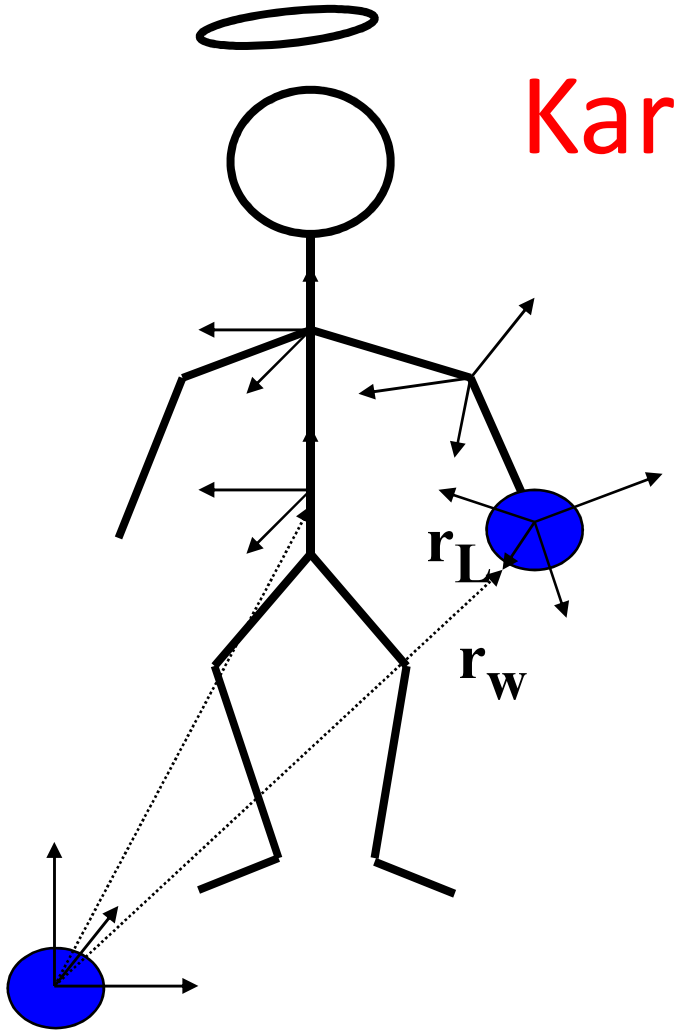
$$\mathbf{v'} = [\mathbf{v} - \mathbf{n}(\mathbf{v}\cdot\mathbf{n})] - [\mathbf{n}(\mathbf{v}\cdot\mathbf{n})\cdot bounce]$$

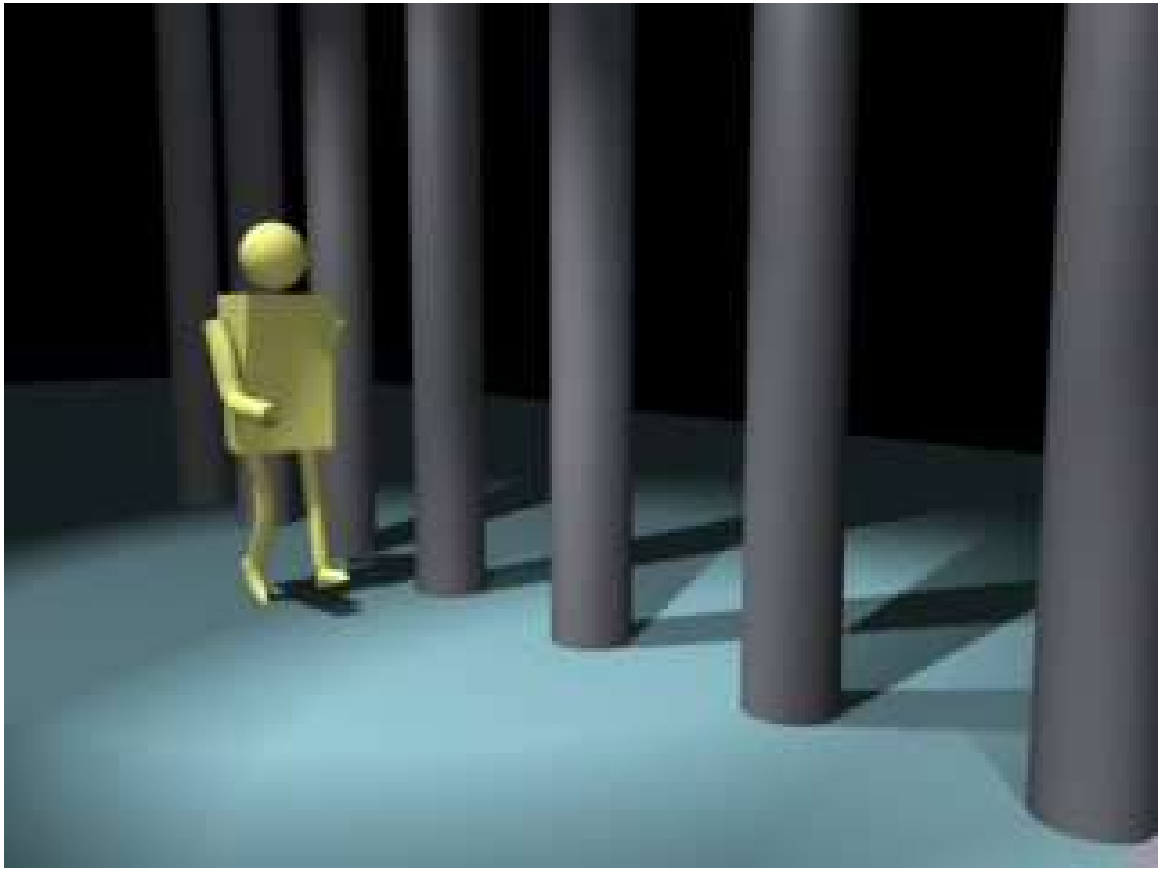
Folytonos-Diszkrét
ütközés detektálás pontra és féltérre





Karakter animáció

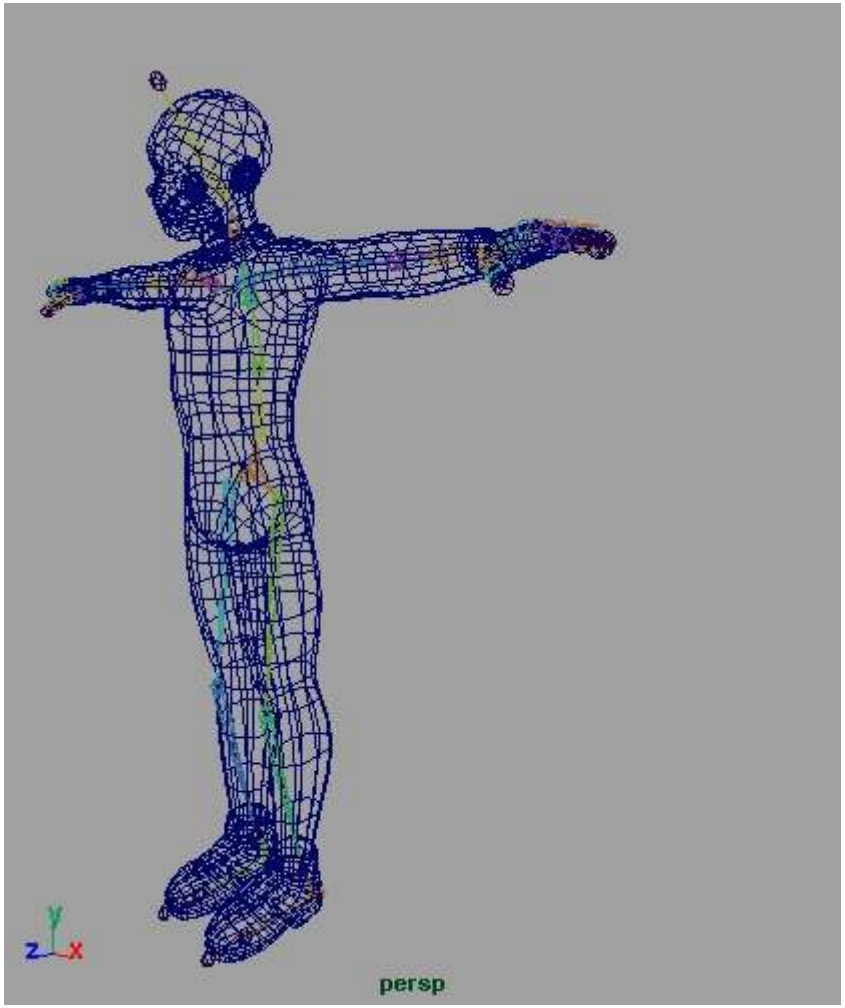


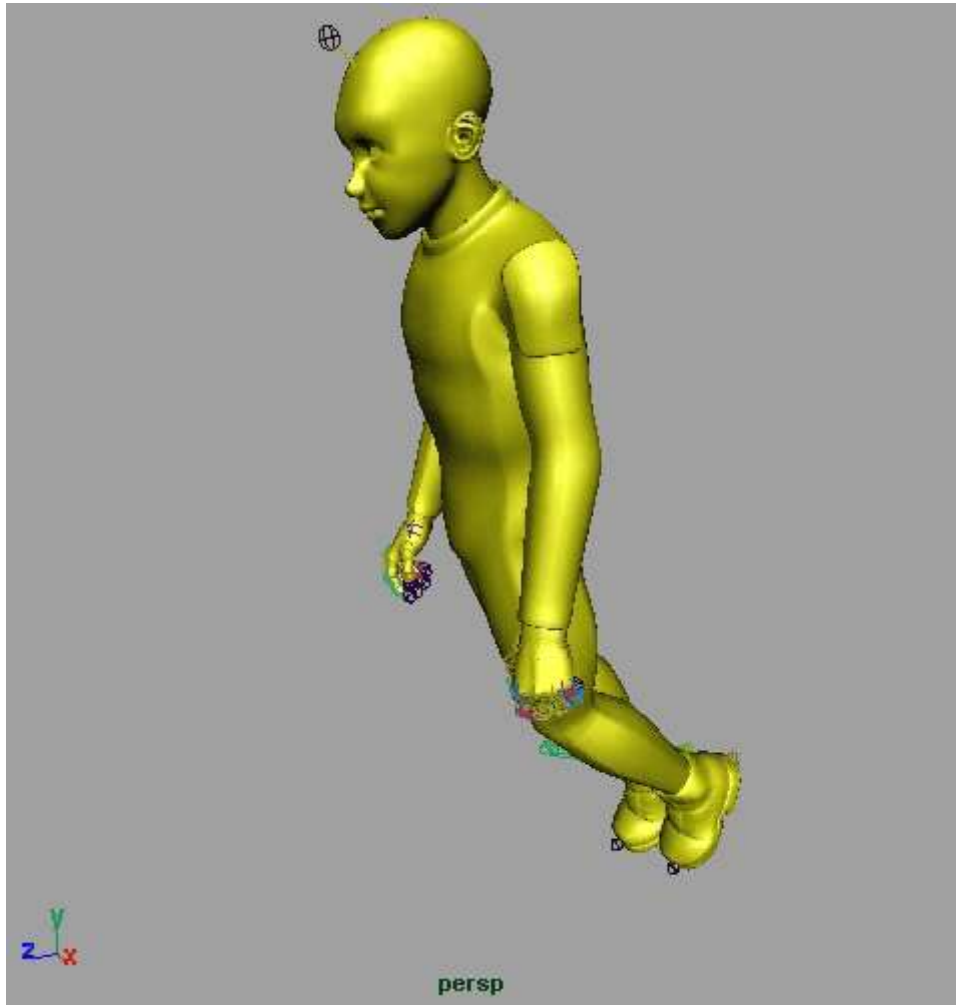


$$\mathbf{r}_w = \mathbf{r}_L \cdot \mathbf{R}_{kéz} \cdot \mathbf{T}_{alkar} \cdot \mathbf{R}_{könyök} \cdot \mathbf{T}_{felkar} \cdot \mathbf{R}_{váll} \cdot \mathbf{T}_{gerinc} \cdot \mathbf{T}_{ember}$$

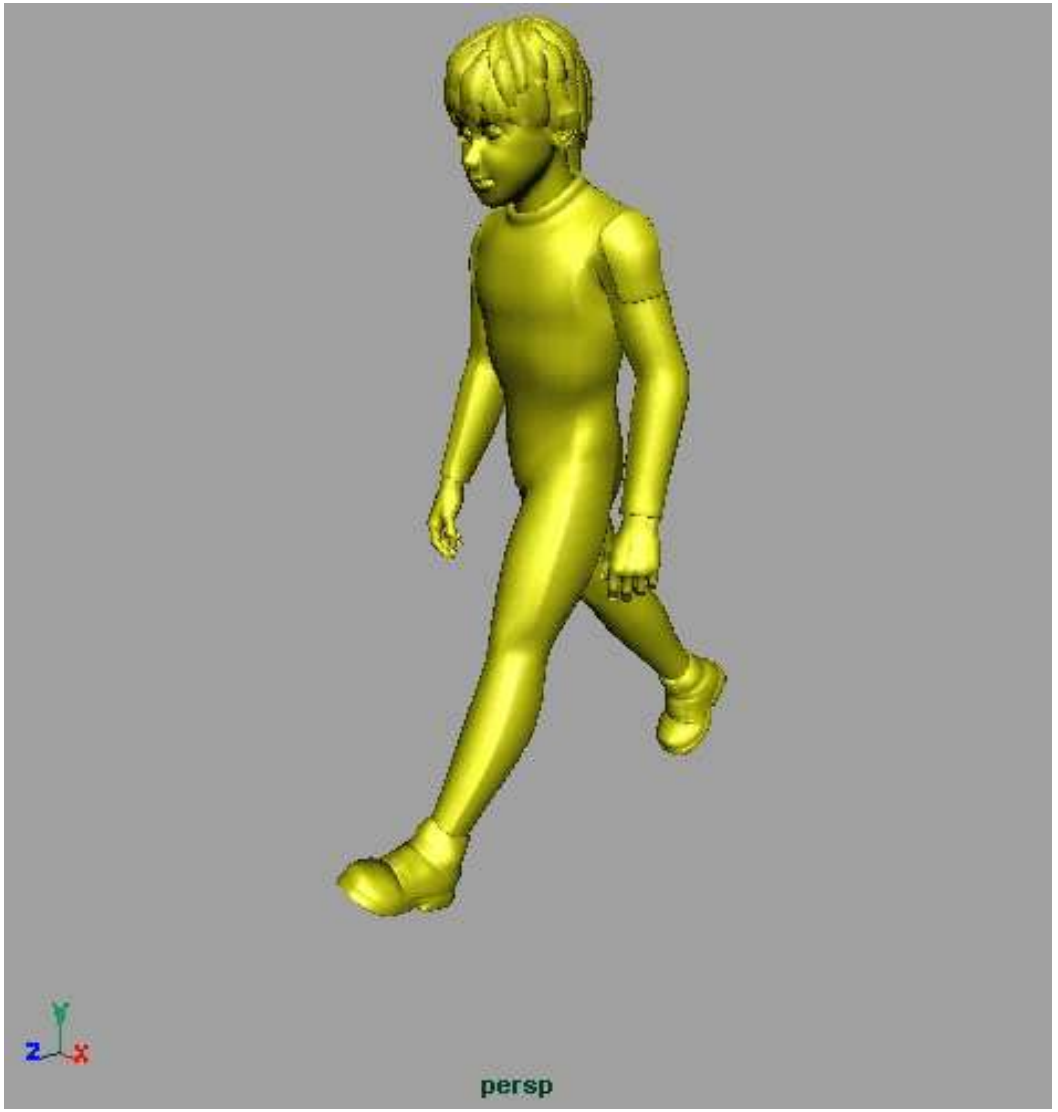
homogén koordináta 4-es

Csontváz

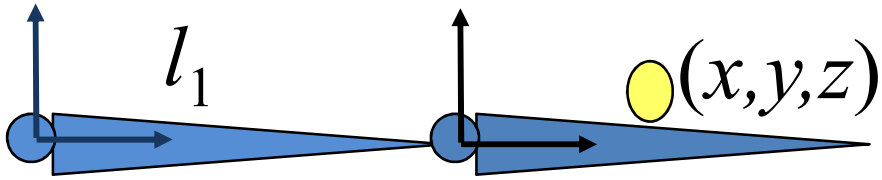




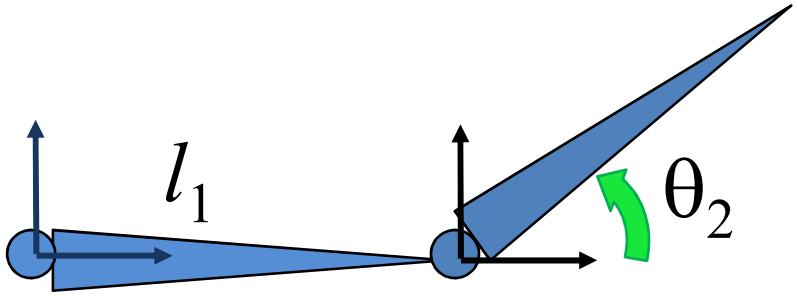
Séta



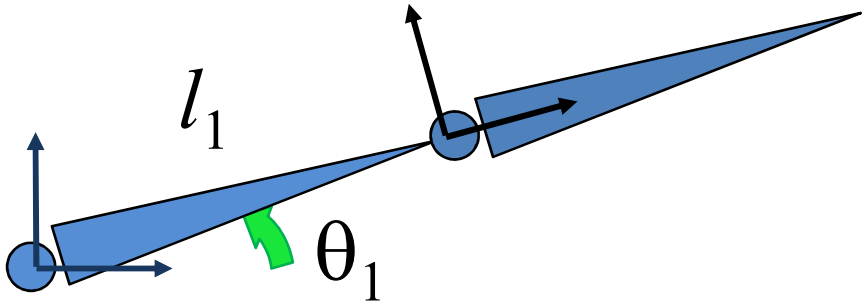
Transzformáció hierarchia



Translate(l_1, 0, 0);

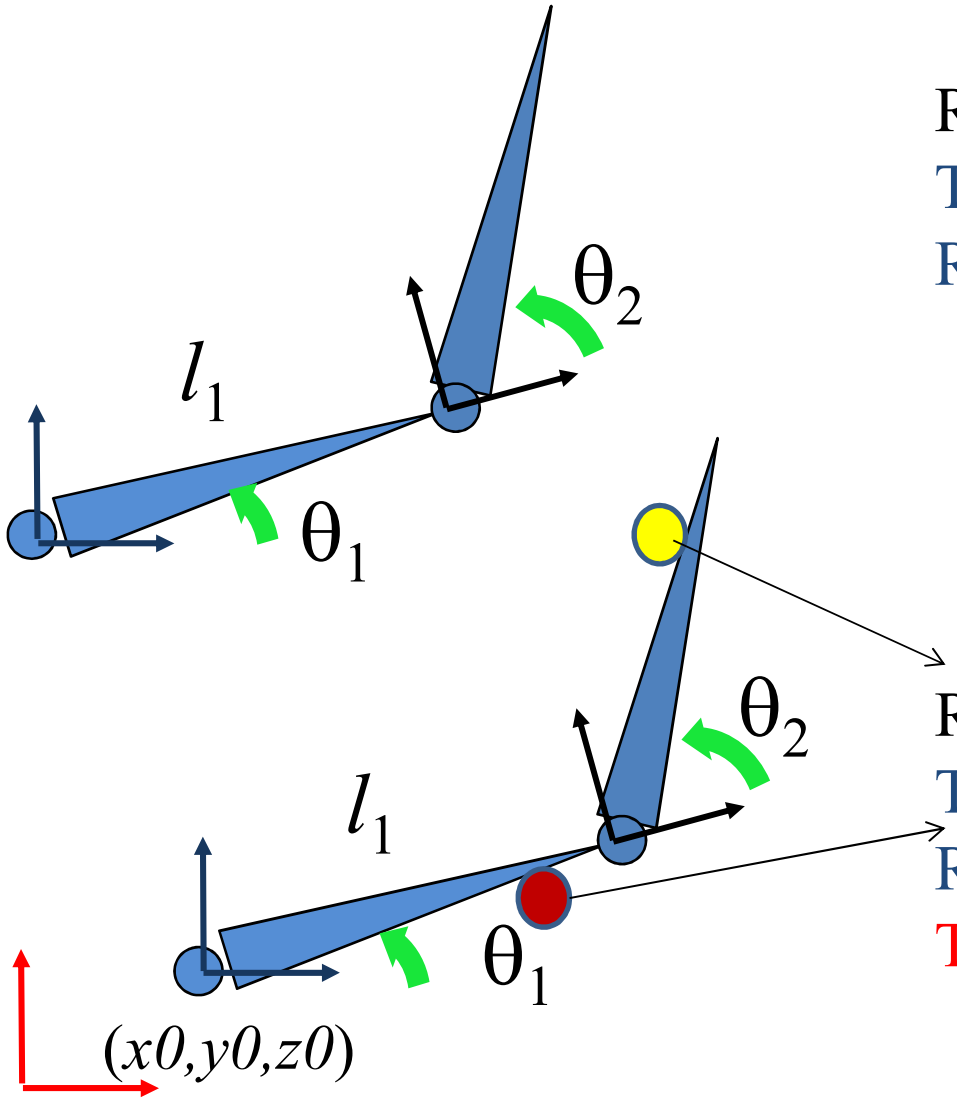


Rotate(theta$_2$, 0, 0, 1) *
Translate(l_1, 0, 0)



Translate(l_1, 0, 0) *
Rotate(theta$_1$, 0, 0, 1)

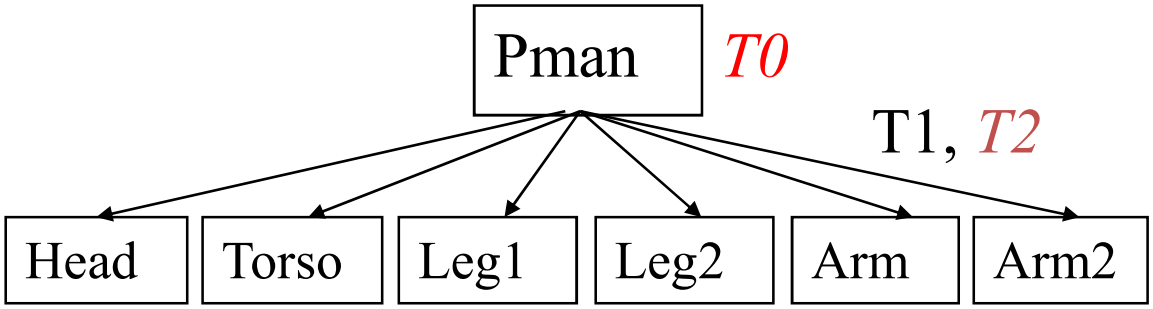
Transzformáció hierarchia



Rotate(theta$_2$, 0, 0, 1) *
Translate(l_1, 0, 0) *
Rotate(theta$_1$, 0, 0, 1);

Rotate(theta$_2$, 0, 0, 1) *
Translate(l_1, 0, 0) *
Rotate(theta$_1$, 0, 0, 1) *
Translate(*x0, y0, z0*);

PMan



T0 = Pman előremozog

Translate(forward, up, 0)

T1= váll pozíció

Translate(leftShoulderPos)

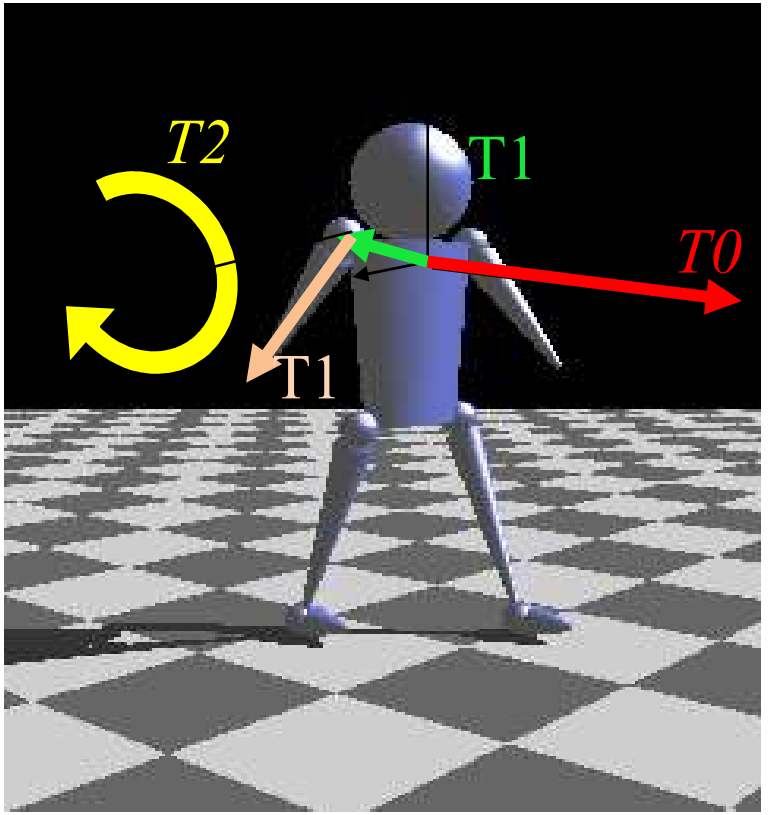
T2 = kar forgatás

Rotate(armAngle, armRotAxis)

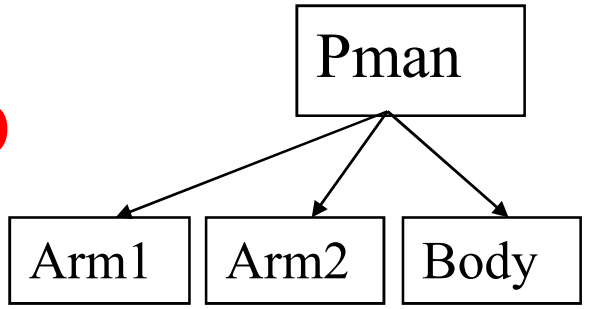
…

T3 = kéz pozíció

Translate(armLength, 0, 0)



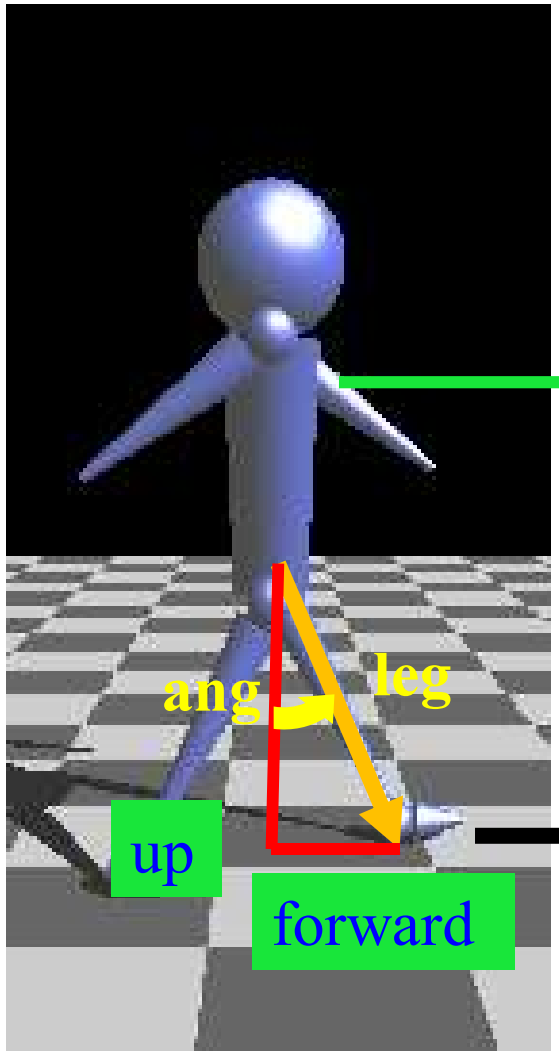
Pman rajzolás és animáció




```
class Pman {
    float armAngle, dArmAngle, forward, up;
    const … armLength, armRotAxis, rightArmJoint, …;
public:
  void DrawArm(float dt, mat4 M) {
     M = Rotate(armAngle, armRotAxis) * M; // T2
     DrawRefArm(M);
  }
  void DrawPman( ) { // set matrices from animation parameters
     mat4 M = Translate(forward, up, 0); // T0
     DrawRefBody(M);
     DrawArm(dt, Translate(rightArmJoint) * M);  // T1
     DrawLeg(dt, Translate(rightLegJoint) * M);
     …
  }
  void Animate(float dt) { // calculate animation parameters
     armAngle += dArmAngle * dt;
     if (armAngle > 0.5 || armAngle < -0.5) dArmAngle *= -1;
     forward += 5 * dt;
  }
};
```

Push/Pop

Inverz kinematika



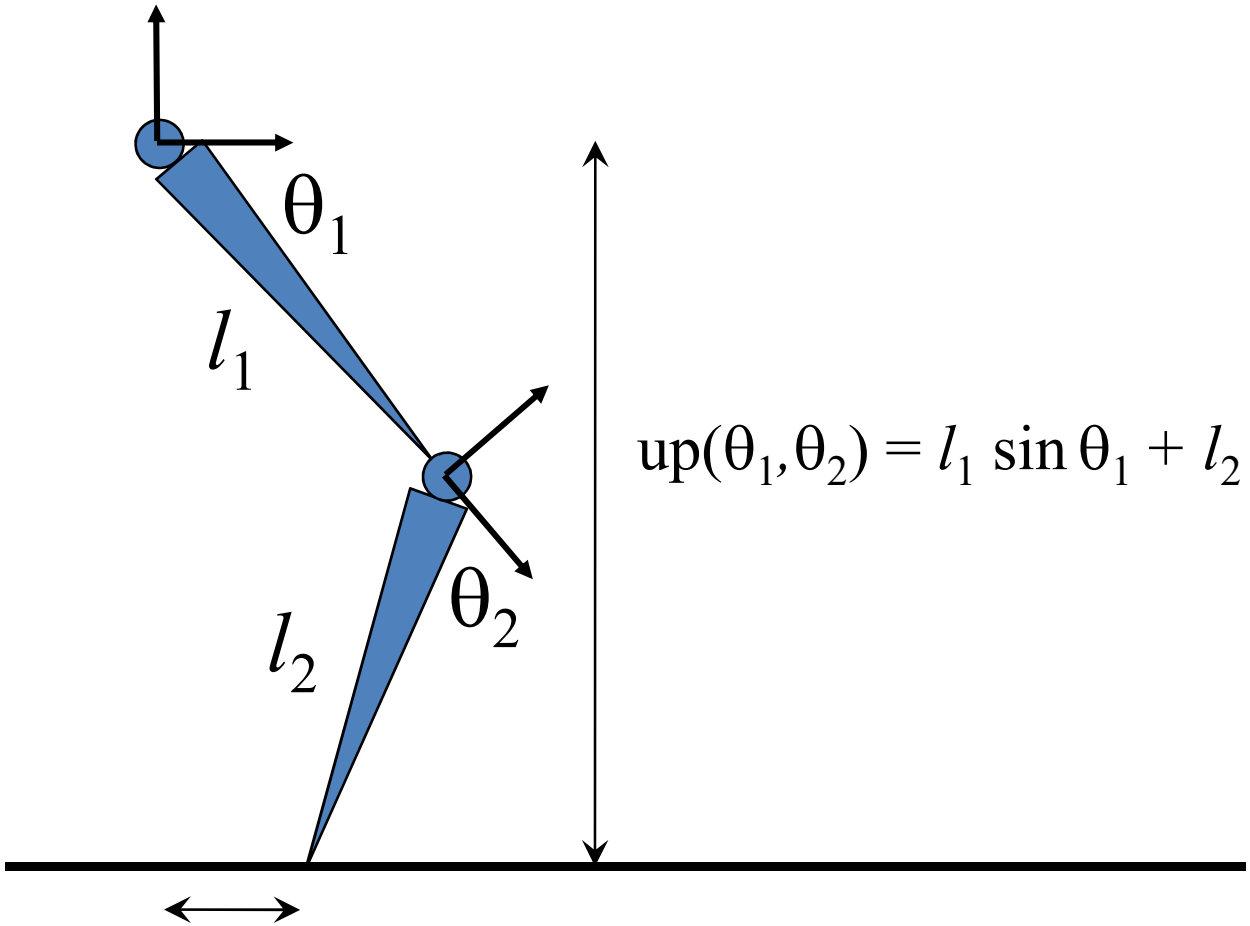
T0 = előremozgás (forward, up) ???

T2 = láb forgatás(ang)

Támaszkodó nem csúszkálhat

```
forward += leg * fabs(sin(angNew) - sin(angOld));
up       = leg * cos(angNew);
```

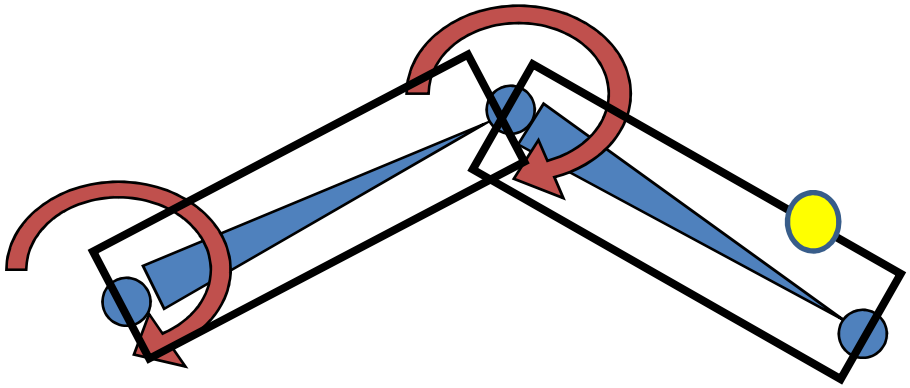
Inverz kinematika

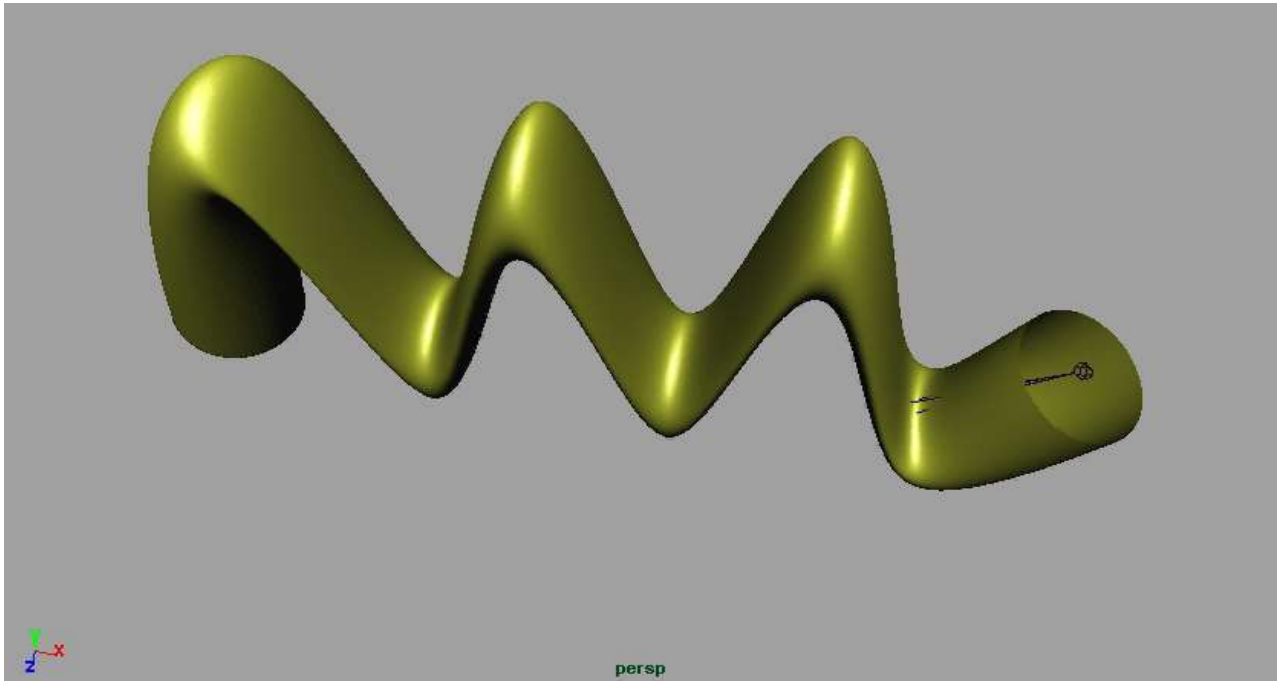


$$\text{up}(\theta_1,\theta_2) = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2)$$

$$\text{forward}(\theta_1,\theta_2) = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2)$$

Bőrözés

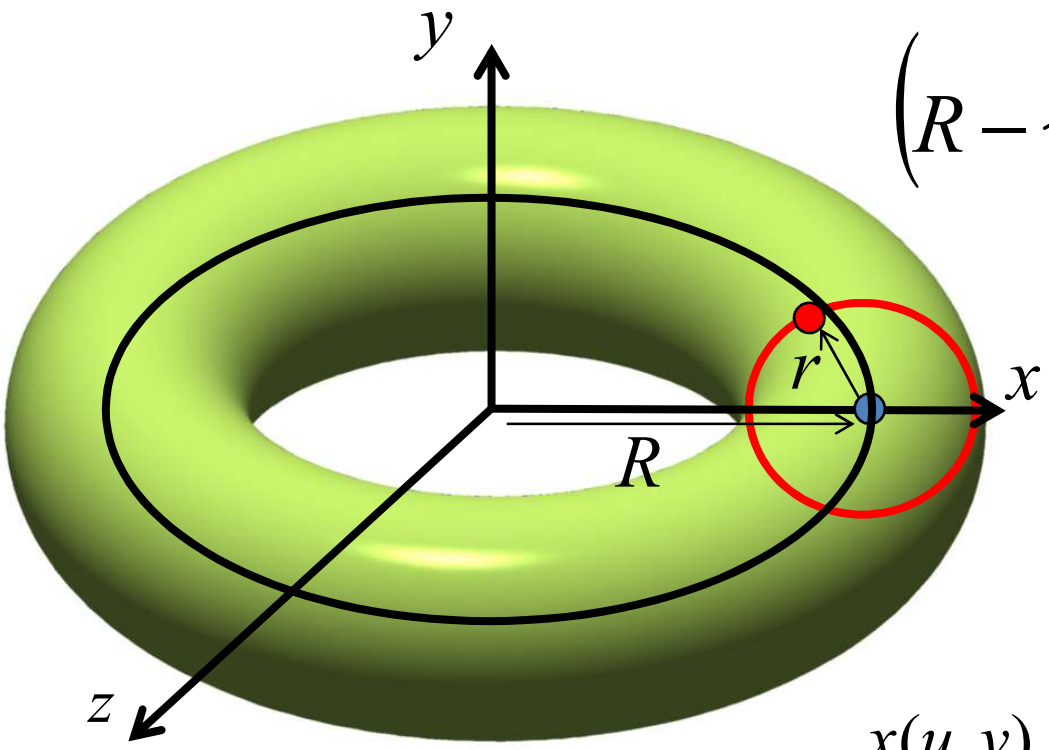




3. házi: Tóruszba zárt Spiderman

Egy procedúrálisan textúrázott, diffúz-spekuláris tórusz belsejében egy ugyancsak procedurálisan textúrázott golyó gördül, egy cián és egy sárga fényű fényforrás labda pattog a tórusszal rugalmasan ütközve és a mechanikai energiáját megtartva, valamint spiderman avatárunk várja a sorsát, akinek a szemszögévől követjük az eseményeket és gyönyörködünk a Phong árnyalt színtérben. A tórusz rögzített, golyó anti-gravitációs készülékkel van ellátva, így nem szakad el a tórusz falától. A többiekre pedig hat a homogén nehézségi erőtér. A golyó pályája a tórusz falán periódikus és nem kör (pályaanimáció). Spiderman mindig a golyó irányába néz, mert szeretnénk elkerülni, hogy a golyó legázolja. Ha a tórusz belső falának egy pontjára mutatunk a bal egérgomb lenyomásával, akkor oda egy nem zérus nyugalmi hosszúságú gumikötelet lő ki, ami megnyúlás esetén a Hooke törvény és a dinamika alaptörvénye szerint magával rántja, így a közeledő golyó elől el tud ugrani. Minden újabb gumilövés a régit oldja.

Tórusz



$$\left(R-\sqrt{x^2+z^2}\right)^2+y^2-r^2=0$$

$$x(u,v)=(R+r\cos(2\pi u))\cos(2\pi v)$$

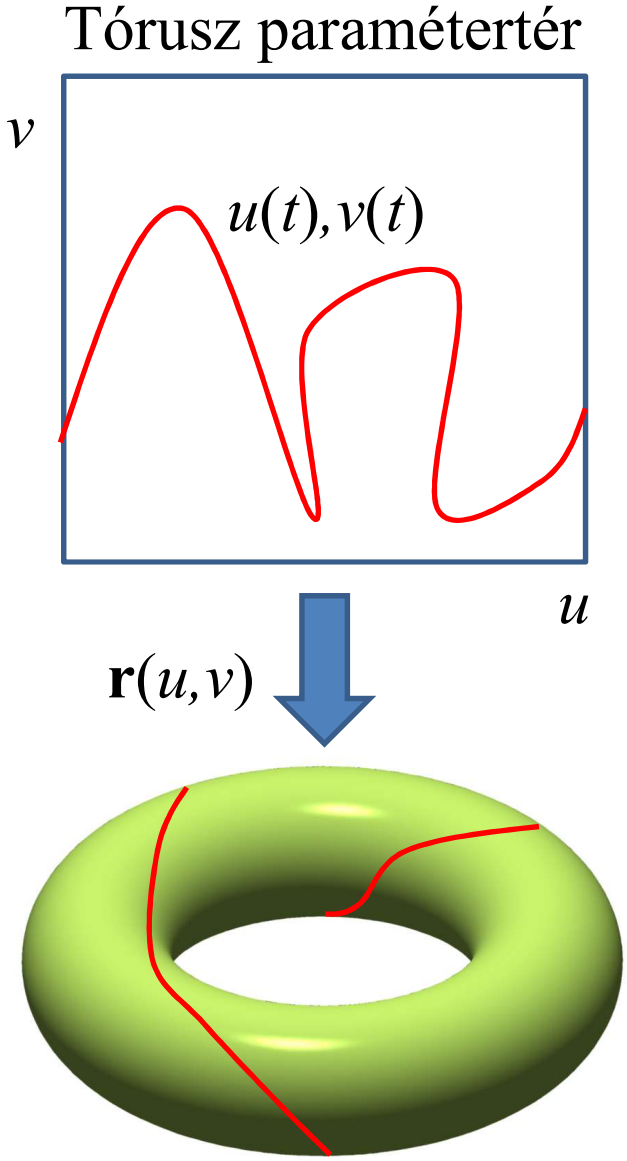
$$y(u,v)=r\sin(2\pi u)$$

$$z(u,v)=(R+r\cos(2\pi u))\sin(2\pi v)$$

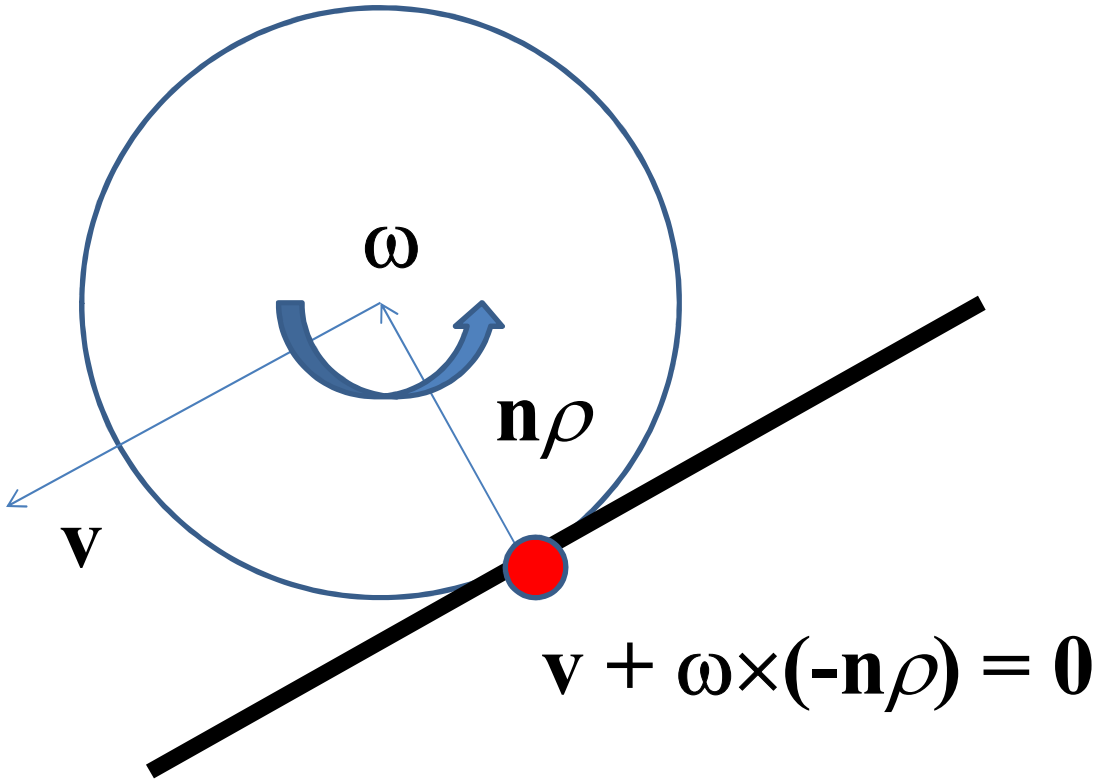
Meglőtt pont előállítása

- Irány visszatranszformálása világkoordinátarendszerbe és ray tracing
- Irány és mélység visszatranszformálása világkoordináta rendszerbe (glReadPixels)
- Képszintézis színek helyett világkoordináták rajzolásával (vigyázat 8 bites rasztertár vagy *render-to-texture*)

Golyó gördül a tóruszon



$$\mathbf{v} = \frac{\mathrm{d}\mathbf{r}(u(t),v(t))}{\mathrm{d}t} = \frac{\partial \mathbf{r}}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}t} + \frac{\partial \mathbf{r}}{\partial v}\frac{\mathrm{d}v}{\mathrm{d}t}$$



$$\mathbf{v} + \omega \times (-\mathbf{n}\rho) = \mathbf{0}$$