Jetmir Berisha
Wednesday 03/13/2019

Homework 2 notes:
Software. After building a model and loading it into opengl, rotating the forearm will result into a separation of the 'skin' at the elbow. Linear Blend Skinning(LBS) is a solution to this and it works by considering part of the upper arm and part of the lower arm as skin that is blended on movement. Also this is not a requirement to Homework 2. General Steps to follow for Hw. 2: Display 3D model, Transform Skeleton, Apply transform to mesh/skin.
Hardware. The professor explained the ultrasonic sensor as a sensor that emits ultrasonic waves out of one of its cylinders and reads the bounced back wave into the other cylinder. Ultrasonic: Sound with higher frequency than the human hearing limit. The time elapsed after sending out a signal and the time to read the signal back can be used to determine the distance using x = v/t where v is the known speed of sound.

Rigidbody Dynamics

Rigidbody dynamics implies that each triangle comprised body is not alterable by physics forces in the sense of stretching/elongation or compression and every vertex in the body moves uniformally. Therefore, in order to represent a rigidbody we only need to represent one position, velocity and orientation, and the following demonstrates how.

Instead of position we use the Generalized Position. Which contains information about the rigidbody's position and rotation. We use the center of mass' current location in cartesian coordinates for the position and a unit quaternion for the rotation.

In order to update the position we use two laws of motion:

$$\vec{F} = m\vec{a}$$
$$\vec{P} = m\vec{v}$$

Where $\vec{P}$ is the linear momentum

Say we start with $\vec{x^0}$ and $\vec{v^0}$. Where i as a superscript denotes the i-th delta time.
And we use forward differencing to calculate the next velocity:

$$\vec{P}^0 = m\vec{v}^0$$

$$\frac{d\vec{P}}{dt} = \vec{F} \;\rightarrow\; \frac{\vec{P}^1 - \vec{P}^0}{\Delta t} = \vec{F}^0 \;\rightarrow$$

$$\rightarrow\; \vec{P}^1 = \vec{P}^0 + \Delta t\,\vec{F}^0 \;\rightarrow\; \vec{P}^1 = m\vec{v}^1 \;\rightarrow\; \vec{v}^1 = \frac{\vec{P}^1}{m}$$

And to calculate the next position we do the following:

$$\frac{d\vec{x}}{dt} = \vec{v} \;\rightarrow\; \frac{\vec{x}^1 - \vec{x}^0}{\Delta t} = \vec{v}^1 \;\rightarrow\; \vec{x}^1 = \vec{x}^0 + \Delta t\,\vec{v}^1$$

Note that we use forward differencing to calculate the velocity and we use backward differencing to calculate the position. The forward differencing will introduce noise and the backwards differencing will dampen that noise, which makes the General Position not diverge.

From the formulas we notice that velocity is a 3D vector and mass is a scalar. We try to represent the mass as a 3x3 matrix.

$$\vec{P} = m\vec{v} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

Using this new definition of mass we can define angular momentum: $\vec{L}$

$$\vec{L} = I\vec{\omega}$$

Where I = Inertia tensor and ω = Angular velocity

Computing I is challenging since it involves computing the following for a rigidbody:

$$I = \int_{\Omega} m * \vec{r}^2 dr = \int_{\Omega} m \|\vec{r}\| dr \quad \text{Where } \vec{r} \text{ is the position of the body}$$

It is important to note that by changing the axes' location and rotation the formula will output different results. We always want a diagonal inertia tensor, meaning all entries are in the diagonal and the rest are 0, so that it is easy to compute its inverse.

I_{Body} refers to the inerta tensor of the rigidbody in the material frame and
I_{Frame} refers to the inertia tensor in the world frame.

The relation between the two is such that:

$$I_{World}(t) = R^T(t) I_{Body} R(t)$$

We know that:

$$R R^T = I = R^T R \quad \rightarrow \quad R^{-1} = R^T$$

In the above line I refers to the identity matrix. Where in all other cases it refers to the inertial tensor.

$$I_{World}^{-1}(t) = (R^T(t) I_{Body} R(t))^{-1}$$

From the formula: $(ABC)^{-1} = C^{-1} B^{-1} A^{-1}$, where A, B and C are matrices

$$I_{World}^{-1}(t) = R^{-1}(t) I_{Body}^{-1} (R^T(t))^{-1} \quad \rightarrow \quad I_{World}^{-1}(t) = R^T(t) I_{Body}^{-1}(t) R(t)$$

In order to integrate and update the body's orientation we use the following formulas:

$$\vec{L} = I\vec{\omega}$$

$$\frac{d\vec{L}}{dt} = \vec{\tau} \qquad \text{where } \tau \text{ is torque}$$

The same logic used to update position and velocity can be used here. Forward differencing for the angle:

$$\frac{\vec{L}^1 - \vec{L}^0}{\Delta t} = \vec{\tau}^0 \quad \rightarrow \quad \vec{L}^1 = \vec{L}^0 + \Delta t\, \vec{\tau}^0$$

$$\vec{\omega}^1 = I_{World}^{-1} \vec{L}^1$$

To be continued next class...