Simulating Simple Physics with Explicit Animation

Overview

There are many occasions when one wants to animate physically accurate motion. This could be a swinging pendulum, colliding billiard balls, or an object tumbling through the air, etc. By explicitly animating groups, and applying some simple physical principals, it's possible to obtain motion that looks very natural and realistic.

The main relevant physical concepts are:

1. Collisions

2. Trajectories

3. Oscillations

In this tutorial, you will animate a bouncing ball that collides with a couple different objects.

Procedure

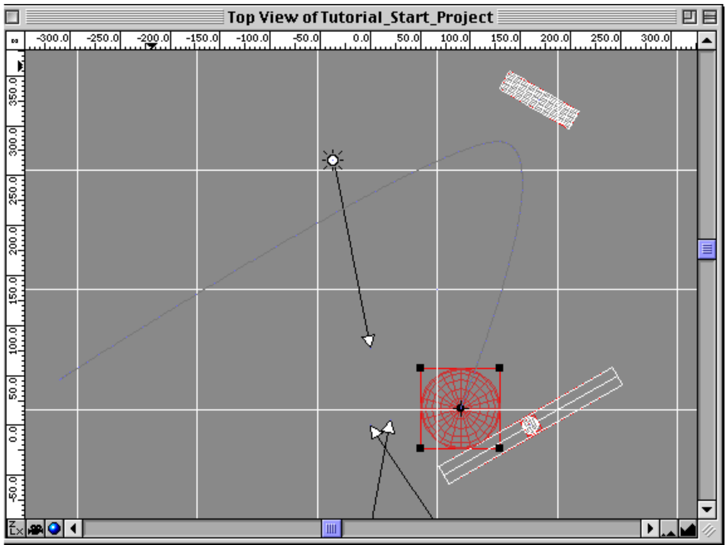
Animating The Ball Horizontally

1. Open the project 'Tutorial_Start_Project'. In it you should find a Ball, a Floor plane, a Pendulum Plank and a Propellor Plank.

2. Start by making sure the Ball object is set to 'Explicit' animation in the Group Info window.

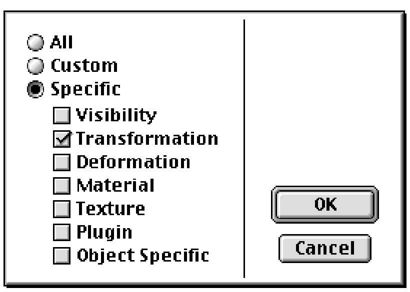
3. Move the time slider in the project window to 1 second. In the top view, drag the Ball until it is touching the Pendulum Plank.

4. Now move the time slider to 2 seconds. Drag the Ball towards the Propellor Plank until it is just touching it. (Note: The angle at which the Ball leaves the Pendulum Plank should approximately mirror the angle at which it hit the plank.) At this point, the motion path will appear curved. We'll fix this later.

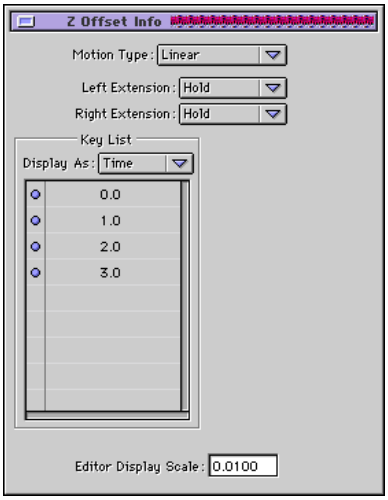


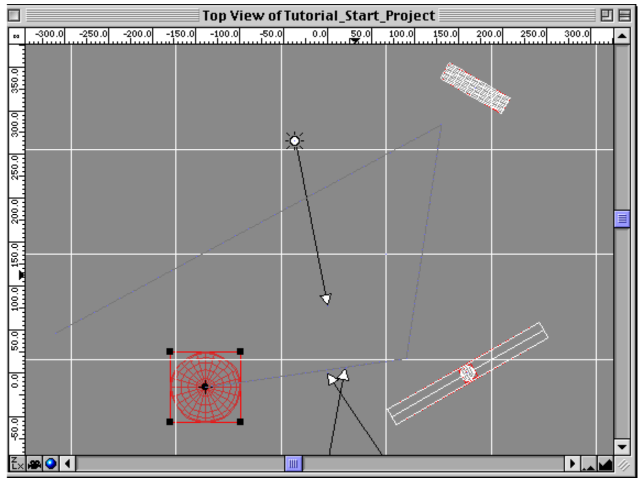
5. Finally, move the time slider to 3 seconds and drag the ball away from the Propellor Plank. As in step 4, the angle of incidence should be roughly equal to the angle of reflection.

6. The ball is now animated horizontally, but it doesn't act as if it's actually colliding with the objects. Open the Key Path Editor (⌘ – `) from the Window menu. With the Ball group selected, click on the second mini-icon from the left on the Key Path Editor window. In the pop-up window, select Specific and the Transformations check box. Un-check all other options and click OK.



7. Select 'Ball_X-Form_X Offset' by clicking on the text. Now click on the seventh button from the left on the Key Path Editor Window. This brings up the X Offset Info window. Notice that the curve type defaults to 'F-curve'. Change it to linear. Do the same for the 'Z Offset' curve. In the Top View Window, it should now look like the ball is ricocheting off the two objects.





Note: Physically what is happening, is that the ball moves at a constant velocity until it hits something. At that point its velocity changes (discontinuously) and it travels again at a different constant velocity until it hits something else. The Key Path Editor Window contains position, velocity and acceleration curves. The position (White) is now linear in between collisions. The velocity (Orange) is constant in between collisions (and thus looks like a stair step). The acceleration (Blue) is zero between collisions because the ball in neither speeding up nor slowing down. In this simple physical model we are neglecting friction or air resistance.

Animating the Ball Vertically

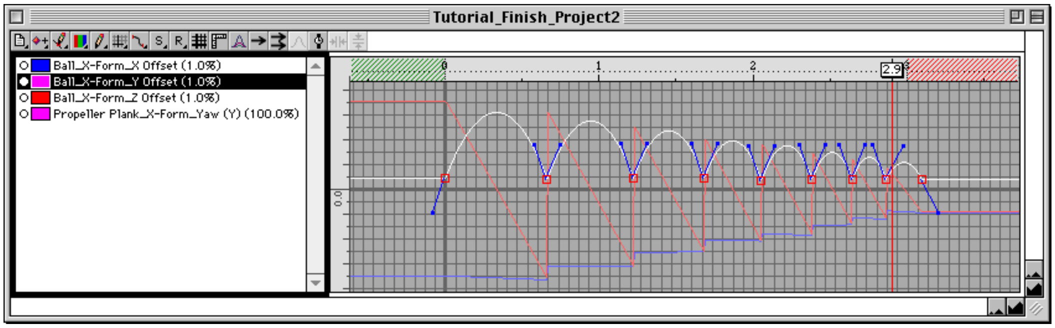
8. With the Key Path Editor still open, select the 'Ball_X-Form_Y Offset'. You can turn off the other curves by clicking the black ball next to curve name. Select the key frames on the curve and choose Delete Keyframe from the Keyframe Menu.

9. Holding down the option key, create another keyframe by clicking on the curve somewhere between time 0.0 and 1.0 seconds. Now adjust the handles on the f-curve so that between the two keyframes the ball goes up and then comes down. To create an accuate trajectory adjust the handles until the acceleration curve is approximately a straight, horizontal line. (The velocity curve should now look like a straight diagonal line.)

10. Set another keyframe on the curve at a time twice the distance from time 0.0 as the last keyframe you set. (For example, if the last keyframe you set was at time 0.35 sec, then set the next one at time 0.7 sec). Break the Bezier handles on the last keyframe by holding down the ⌘ key when you click on the handle. Adjust the handles until the acceleration is horizontal again. It should have the same constant value across the two time intervals.

11. Repeat the previous steps until the ball bounces throughout the animation. Note, you do not have to use a constant time interval between bounces. If you would prefer to have the ball steadily losing energy, you can make the time intervals

between bounces shorter and shorter. The height of the bounces should also be less and less. The acceleration, however, should be the same constant** value throughout the animation!



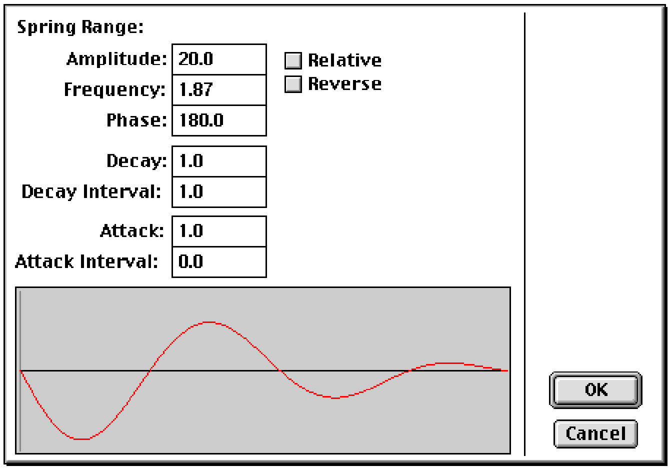
**************************WARNING**************************
The display of the acceleration curves is not quite correct. The magnitude of the acceleration is not displayed.
****

Note: The vertical animation of the ball is completely independent of the horizontal motion. The key concept in simulating the trajectories in a uniform gravitaional field (neglecting air resistance), is that the acceleration is a constant value for all objects, regardless of mass and regardless of velocity. By forcing the acceleration curve to look like a horizontal line you are creating a physically accurate motion. The velocity curve should now look like a sawtooth.

Animate the Swinging Plank

12. If you play your animation now, you should have a nice bouncing ball that appears to collide with a couple objects. At this point you can adjust the vertical position of the pendulum plank if you want. In the project view window move the time slider to the frame at which the ball collides with the Pendulum Plank. Go to keyframe mode and select the field containing the X rotation. Scroll to the end of the animation and Shift click the X rotation field, thus selecting all the X rotation values after the collision.

13. Click on the Fill tool  from the Project View Window and select Spring. Set the attack interval to 0.0 and the decay interval to 1.0. Set the amplitude to be about 20 and the phase to 180. Set the frequency around 2 and click OK.



Note: The spring Fill tool allows you to create sinusoidal oscillations. This kind of curve describes a wide variety of physical motion including springs, swings, pendula and waves. You can also try applying another spring with a slightly different frequency to the Z rotation of the plank. This will cause it to swing as if suspended from point.

Animate the Rotating Plank

14. You may need to adjust the Y offset of the Propeller Plank so that the Ball collides with it. Having done this, go to the frame where the Ball contacts it and set a keyframe for the Y rotation.

15. Now go to the end of the animation and give it a Y rotation of around -120˚. Adjust your function curve so that the acceleration is constant (horizontal).

Note: The Propellor Plank is given an initial rotational velocity at the moment that the ball hits it. After that the only force acting on it is friction, which is a constant force. Constant force equals constant acceleration. Therefore, making the acceleration curve horizontal yields a natural slowing down of the Plank. If you did not want to have friction (ie. an object tumbling through space) you would make the rotational velocity curve horizontal (the acceleration would now be zero) and the object would rotatate constantly until colliding with something else.

---

16. You can play around with different variations of bounce heights, spring frequency, decays and friction until you get the animation to look the way you want. By adhering to some simple physical principles, as a baseline, you can insure that certain aspects of your motion look realistic.