PHYSICS SIMULATION

Introduction

There are three main ways to bring animation into a scene: hand animation, motion capture, and physics simulation. The last of these, physics simulation, allows faster and more realistic animation of physical processes than is possible with either of the other two. Physics simulation includes fire, smoke, fluids, cloth, soft bodies (jelly), and hard bodies (rigid objects like bricks). In this handout we’ll look at the last of these, creating a brick wall that is demolished by a heavy steel ball rolling through it.

Exercise

Make a cube and scale it to resemble a brick (8” long by 4” deep by 3” high). Add a Bevel modifier to it and look at one of the corners. You’ll notice that the facet is not equilateral as expected. That is because the scaling has been carried out in Object mode, and the brick’s scales in each direction are not equal.

Use Control-A and Apply Scale to normalize the scale. The bevel will probably be far greater than needed now; reduce it to a realistic value. Since we’re going to be building a wall with these bricks don’t add more than one segment to the edges or the mesh will be unnecessarily complex.
Now add an Array modifier and make a short row of bricks, leaving a small space between them.

Apply the Array modifier and then in Edit mode separate the loose bricks. Return to Object mode.

Select each brick in Wireframe view. Note they all have the same pivot point, which is that of the original object. For accurate simulation all the pivot points need to be at the geometric centers of the bricks.

This is done by selecting all the bricks and using the pop-up menu at the foot of the 3D window. Use Object - Transform - Origin to Geometry.
Now build a simple wall. Make half bricks to finish the odd courses, but remember to scale those bricks in Edit mode to preserve the uniform scaling and then use Object - Transform - Origin to Geometry again to move the origin to the center of the half bricks.

None of the bricks in the wall touch each other; they seem to be floating in space. Well add a ground plane and settle the bricks on the ground plane using physics simulation, then apply that simulation. This places all the bricks in position before we add a new simulation of a ball knocking the wall down.

Using the Physics tab in the left fly-in, select the ground plane and make it a Passive object. The outline will turn from orange to green, indicating that it’s now part of a group.

Select all the bricks and give them Active physics. Now in the Timeline window, move the playhead to around the one hundred frame mark and press E for end. We’ll need to run the simulation a number of times to get it right, and everything essential will be complete by frame 100.

Hit Alt-A to start the simulation. The bricks will settle on top of each other with a bounce. Some may jump out of position or even fall out of the wall. If this happens, the simulation needs refinement. Go to the Scene tab in Properties (third from the left) and then to the Rigid Body World panel. Increase the Steps per Second to 300 and the solver iterations to 60. This produces a more accurate simulation, though at the cost of more calculation. However, to set up the wall, it’s worth it.
The wall should now drop into an accurate start position. We now have to remove the simulation while keeping the bricks in their present positions. Select all the bricks and hit Control-A. Select Visual Transform.

One problem you may encounter is that at actual scale (using Imperial or Metric units) the simulation may still be twitchy. Scale everything up ten or twenty times and normalize it with Control-A Scale. Then try it again; that will work more reliably. If the bricks are still moving around slightly use Enable Deactivation and Start Deactivated to have them stay still until affected by another element in the simulation.

When the bricks are settled on the ground we can introduce a new element: the ball that’s going to knock them down. Make an Icosphere, give it three degrees of subdivision, and make it smooth. Place it slightly above the ground, give it Active physics, and run the simulation. It will drop to the ground and, if the ground is level, stay there. Find a frame where it is sitting on the ground, use Control-A, and select Visual Transform. Now when you run the simulation with Alt-A the bricks and ball will sit there, though the ball may drift slightly towards the end of the hundred frames as slight errors build up.

Go to the first frame (Shift-LeftArrow). With the ball selected, and using the I key, make a Location keyframe. Go to the last frame (Shift-RightArrow), move the ball past the wall, and make a second keyframe. Run the animation. The ball doesn’t move. The reason for this is that the ball is currently accepting only simulation information, not direct animation through keyframes. Let’s go to the Physics tab in Properties with the ball still selected.

You can see there are two properties at the top of the Rigid Body panel: Dynamic and Animated. The first one refers to the physics simulation and is checked. The second

refers to hand animation - what we just tried to do - and is not checked. Click on the box to place a check mark in it and run the simulation again.

The ball will smash through the wall, sending the bricks flying, and then stop. So far, so good. But
we don’t want the ball to stop. It should roll on until it reaches the edge of the ground and then fall off. But the Animated command causes it to stop at the last keyframe.

The answer is to animate the Animated checkbox. Go to a frame similar to that above, where the ball has smashed through the wall. Place the cursor over the checked Animated box and hit I. The checkbox will turn yellow, indicating it has a keyframe at the current frame. Use the right arrow key to move to the next frame, uncheck the Animated box, and again use I to set a keyframe value. Scrub back and forward in the timeline to see the effect of this.

Many checkbox values can be animated in Blender, including the Visibility, Selectability, and Renderability ones in the Outliner window. Having items disappear in the 3D window at particular points in an animation - but still render as before - is often useful.

So now preview the simulation one more time. The ball will smash through the wall and keep on going until the edge of the ground plane, at which point it will fall under gravity out of the scene. You may need to extend the simulation past one hundred frames to see this.

There are now many ways we can adjust this simulation. If the ground appears too slippery and many bricks are sliding off the edge, increase its friction in the Physics tab in Properties. If you want to make an animation out of this, select the physics simulation objects and use Bake to Keyframes in the left fly-in. Bake to Keyframes removes the physics simulation from the selected objects and bakes their movement to keyframe values. They will then behave as if they’ve been hand animated.
This may take a time, but if you then go to the Dope Sheet to view all the keyframes in the scene you get an idea of why it does. The keyframes can be reduced and smoothed in the Graph editor.

You can then any or all of these values in your animation.

Chris Yonge - 20151016