Part 1) How Helicopters Work

these rotors counteract all of the undesirable torque created when spinning the main rotor by spinning in the opposite direction. By adjusting the speed of the tail rotor, you can also increase or decrease the net torque on the helicopter, causing it to *yaw*, or spin on the Y axis. This allows the pilot to easily rotate the helicopter to face the desired direction while spinning in place, without having to bank, or make a wide turn like an airplane.

The combination of all of these systems allows a helicopter a full 6 degrees of freedom. It can fly forward, backward, left, right, up, down, and rotate along the X, Y, and Z axes, allowing much more control and maneuverability than any other flying machine in existence. However, if one of these systems fails, the results can be catastrophic.

Part 2) Setting up a Helicopter in Unity3D

Now that we have a basic understanding of how real helicopters work, we have to figure out how to simulate it in Unity3d. While we could technically program all of the dynamics involved into the Unity engine, it would not run nearly fast enough, or be stable enough to control in a game scenario. In this tutorial, we will be constructing a simple rigidbody helicopter that manually applies the forces that the rotors would in real life by means of a simple script. Before we can do that howeveoci7volughge that tle rigidbse tup Tf pe(ne.to fi)Tj E use the menu item *Component/Physics/ … Collider* to attach a collider

These are the actual collision boundaries in Unity, and determine how an object will collide with others. You can look at the colliders in the "Inspector", and edit their properties. You will want to change the sizes, and positions of the colliders (in the box sphere or capsule collider tag theum "Inspec), so thok at y matttars a sample hierarchy for colliders in a helicopter.

a picture of the final helicopter collision boundary, created with 3 box colliders.

Now that the collision boundaries are saup, wthean actually attachat th a snapshot of the properties for the helicopter rigidbody visible in the inspector.

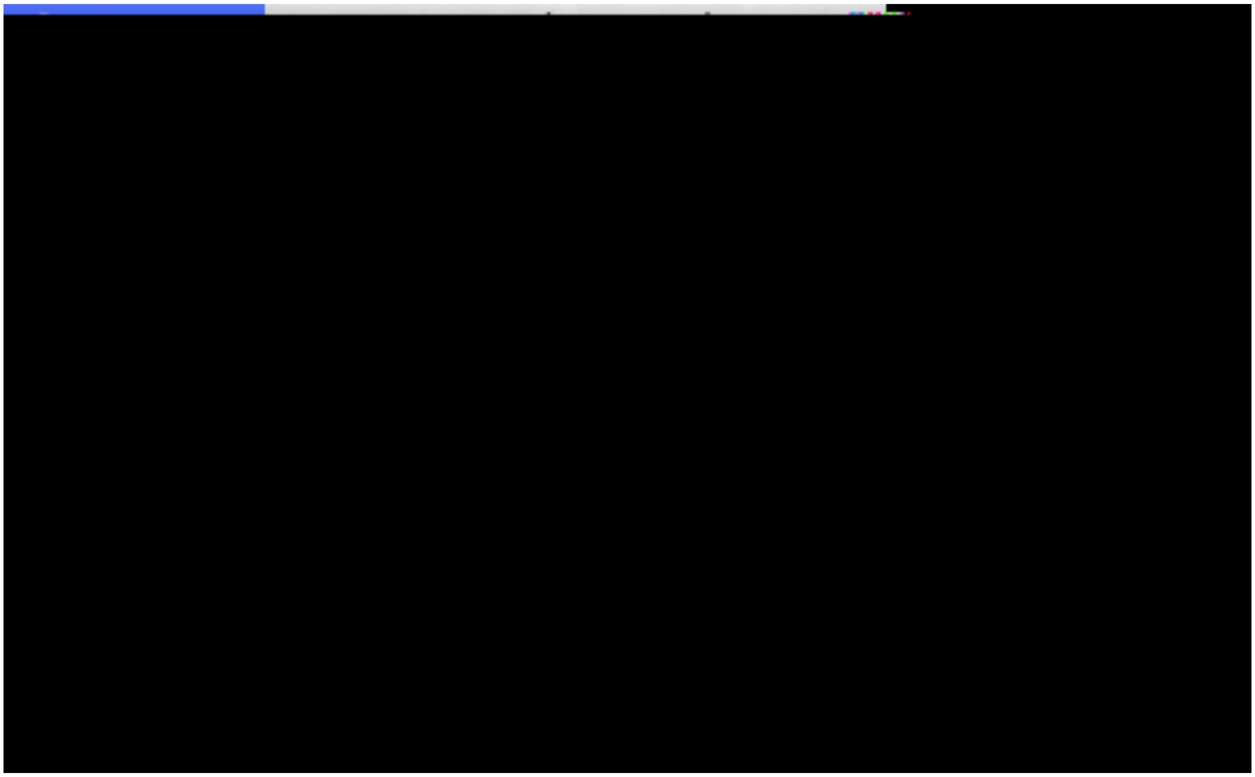
And there you have it. When you press play, your helicopter should plummet to the ground and react accordingly. Next we will write a simple behavior script to allow it to actually fly using player input.

Part 3) Writing the Script

Ok, now that we have a helicopter set up in Unity, we will need to program it to respond to user input. First, we will need to create a new Javascript file. Go to the menu, and select *Assets/Create/Javascript*. This will add a new javascript fi

TorqueValue is just an empty 3 dimensional vector representing the net torque on the object. We will fill this out later on in the script. Now we compute the "controlTorque", or the amount of torque applied to the body of the helicopter based on the player input. This is done to simulate the varying angle of attack on the helicopter blades, without

Lastly, the helicopter should make a sound. This is relatively simple, because we already have a value from 0.0 to 1.0 that corresponds to the throttle, all we need to do is attach an audio source, and adjust its pitch. In unity, you Attach an audio source just like you attach any other component. Go to the component menu, and select *Component/Audio/Audio Source.* This will simply attach a 3D audio source to the object you have selected, in this case our helicopter.



Like all components, the audio source will show up in the inspector view, and you can edit it's properties. Set the audio clip to whatever sound you wish to make your helicopter play, then set play on awake, and loop to true. This will make the audio source play the helicopter engine sound constantly throughout the entire game.

something, such as those included inudeon11 Tm /r project folder. TheincaW nentirely

Now, in the update function of the new gun script, we can write some simple code. Firsthill want to turnctiion ofpartic siemitter,on on check iion offiode.