

Projectile Trajectory

Introduction :

"Projectile Trajectory" allows you to draw and visualize the trajectory in-game that a Rigidbody will pass through under the effect of gravity in **2D** and **3D**. The package includes two scripts , **TwoDTrajectory** and **ThreeDTrajectory** and 2 example scenes , one for 2D and one for 3D. The scripts are open and are commented.

Setup guide :

Drag one of the trajectory scripts on a gameobject , which we will name "Launcher" for the rest of this guide.

The parameters are divided into two sections : Launch parameters and Trajectory Preview parameters.

Launch Parameters :

Fixed Launch Angle : Fixes the launch angle and makes it independent of the "Launcher"'s rotation. If this is false , the angle will be set according to the Launcher's Rotation's X or Z rotation in 3D or 2D respectively.

Invert Launch Angle : Inverts the launch angle if your model's pivot is not facing the correct way.

Launch Angle : If the launch angle is fixed , this is the value that will be used , in degrees.

Launch Speed : The launch's initial velocity.

Launch Position : Launcher's transform. If this is not set , the transform in which this script is assigned to will become the Launcher.

Draw : Indicated whether or not the trajectory should be visualized.

Trajectory preview parameters :

Line count : The number of lines to be drawn to represent the trajectory. Keep in mind that a very large number may cause performance issues.

Line Length : The length of a single line , in world units.

Max distance : The distance in which the trajectory will no longer be drawn , in world units.

Preview Material : The material which lines will be rendered with.

Scroll Speed : Lines' movement speed.

Example scenes :

The 2D example scene presents a 2D slingshot example. The projectile is on top of the slingshot and can be dragged by the player. A script attached to the main camera named TestScene2D calculated the launch angle and speed based on the displacement of the projectile from its original position , and will spawn a ball with initial velocity based on the projectile's rotation and the launch speed.

The 3D example is a cannon scene. The player has control of 2 sliders , Angle , and power which affect the launch angle and launch speed respectively. Clicking on the fire button will spawn a ball with initial velocity , in the same direction as the cannon , and with magnitude based on the power.