Orbital Gravity

Operation Manual

Content

1 Overview

Orbital Gravity is a complete Unity project. The main part of the project is the implementation of the famous Newtonian celestial mechanics. It's a plain gravity driven simulation of celestial body movements. The planetary movements are not based of any geometrical astronomy, which means that the simulation is a classical n-body problem with no Kepler prediction. If you change the mass of a body being part of the system, every celestial body gets affected immediately.

It contains ten celestial models that are 4K PBR textured and have three levels of detail.

2 The Gravity System

The physical system is based on Newton's famous gravity laws. No Kepler, no Einstein. The system is simple in its nature: Every celestial body attracts every other body in the system.

The applied force is defined as:

Force between m_1 and m_2

$$F = G * \frac{m_1 * m_2}{r^2}$$

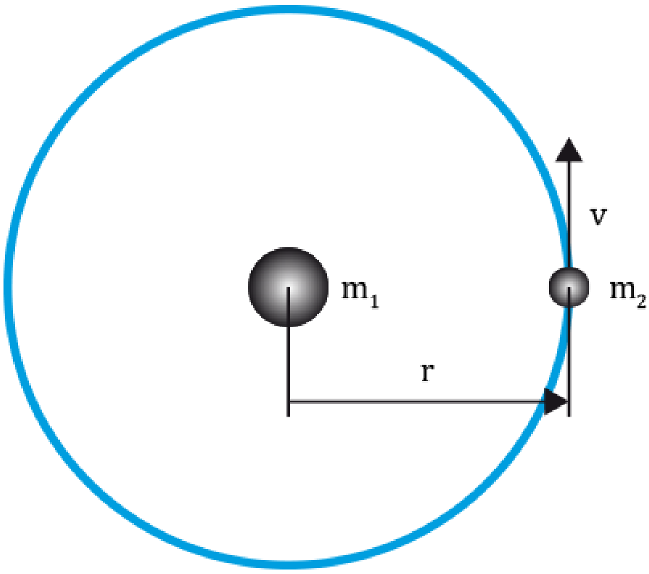
Force between m_k and all m_i

$$\overrightarrow{F_k} = Gm_k * \sum_{i=1|i\neq k}^{n} \frac{\overrightarrow{r_i} - \overrightarrow{r_k}}{|\overrightarrow{r_i} - \overrightarrow{r_k}|^3}$$

Where

F	Resulting Force	
G	Gravitational Constant	$6.673 * 10^{-11}\,\mathrm{N} * \left(\frac{\mathrm{m}}{\mathrm{kg}}\right)^2$
m_1	The mass of body 1	
m_2	The mass of body 2	
r	The distance between the bodies	

The resulting force F gets applied to both directions $\overrightarrow{m_1m_2}$ and $\overrightarrow{m_2m_1}$.



For floating-point-precision reasons all values are scaled by the factor 10^9. That means that e.g. the masses are noted in *Million Tons* instead of *Kilograms*.

3 Usage

3.1 Celestial Body

Every gravitational object needs an `UiModel` and a `GravityObject` component to work properly.

3.1.1 UiModel

The `UiModel` component holds all the information needed to manage the presentation of the celestial body.

Setting Name	Description
Start Zoom	The zoom distance of the camera when the object gets active in million kilometers.
Max Zoom	The maximum zoom distance of the camera in million kilometers.
Min Zoom	The minimum zoom distance of the camera in million kilometers.
Fake Scale	The body gets scaled by this factor. Use this value to improve the display of the celestial bodies in relation to each other.

3.1.2 GravityObject

The `GravityObject` component holds all the information needed to perform the physical calculations.

Setting Name	Description
Is Star	Determines whether the body is a star/sun.
Mass	The mass of the body in million tons.
Diameter	The diameter of the body in million kilometers. This value Is used to scale the body properly at the start of the simulation.
Initial Speed	The body's speed in forward direction at the start of the simulation. Use this value to give the body a starting speed so that it does not fall into the star/sun.
Light Intensity	The light intensity when the body is set active. Use this value to control the visual intensity of the sun light for the corresponding planet in the system.

3.2 Gravity Manager

The `GravityManager` component holds some general information about the simulation.

Setting Name	Description
`Scale Factor`	A uniform scale factor to scale all bodies in the system by a single value. **Note**: Only the model gets scaled, not the underlying physics.
`Time Scale`	Scales the integration time step **dt** by this factor. This makes the simulation run faster. **Note**: larger values can lead to inaccuracies in the physical model.
`Sun Light`	The light object representing the sun.

Note that a high value for the **Time Scale** will reduce the accuracy of the complete simulation.

3.3 UI Manager

The `UiManager` component holds all information about the user interface. It also acts as the view controller.

Setting Name	Description
`Mouse Orbit`	Camera reference holding the mouse orbit component.
`Target UI`	Render texture reference used to render the unprocessed 3D GUI. **Note**: This is not used when you launch this project with **Unity Free**
`Number Format`	The format used to display the numbers. You can insert all C# compatible formats here.

3.4 Mouse Orbit

The `MouseOrbit` component controls the camera and limits its transformations in 3D space.

Setting Name	Description
`Active Object`	The target transform to look at.
`Zoom`	The current zoom distance in million kilometers.
`X Speed`	The speed of the camera that orbits the active body in x direction.
`Y Speed`	The speed of the camera that orbits the active body in y direction.
`Zoom Speed`	The zooming speed. The zoom speed depends on the distance to the object. When the distance to an active object is big, the zoom speed gets bigger too.
`Y Min Limit`	The minimum angle limit around the y-axis relative to the xz-plane.
`Y Min Limit`	The maximum angle limit around the y-axis relative to the xz-plane.
`Min Zoom`	The minimum distance to the active object.
`Max Zoom`	The maximum distance to the active object.

4 Appendix

4.1 Table of Celestial Bodies

	Sun	Mercury	Venus	Earth	Mars	Jupiter	Saturn	Uranus	Neptune
Orbital Speed [in km/s]	0	47,8725	35,0214	29,7859	24,1309	13,0697	9,6724	6,8352	5,4778
Inclination	0	7,00487°	3,39471°	0,00005°	1,85061°	1,30530°	2,48446°	0,76986°	1,76917°
Equatorial Radius [in km]	695.800	2.439,76	6.051,59	6.378,15	3.397	71.492,68	60.267,14	25.559	24.764
Mass [in 10^{24} kg]	$1.988*10^6$	0,33022	4,8685	5,9737	0,64185	1898,7	568,51	86,849	102,44
Gravity @ Equator [in m/s²]	274	3,70	8,87	9,77	3,69	23	8,8	8,6	11
Escape Speed [in km/s]	617,3	4,25	10,36	11,18	5,02	59,54	35,49	21,29	23,71
Equator to Orbit Inclination	7,25°	0,0°	177,3°	23,45°	25,19°	3,12°	26,73°	97,86°	29,58°