Krittika Summer Project

Astrophysically Accurate Animations

Apurv Keer

Mentors: Aswin Suresh and Namita Agrawal

1 Introduction

My KSP 3.0 animation is about binary black hole merger. This animation contains the simulation of two black holes merging in a spiral orbit around each other, the gravitational waves formed from them and how the final black hole would look if it had matter swirling around it. I divided the animation in four sub parts, the two black holes warping the spacetime, the gravitational waves, graph of strain and the final black hole with accretion disc around it.

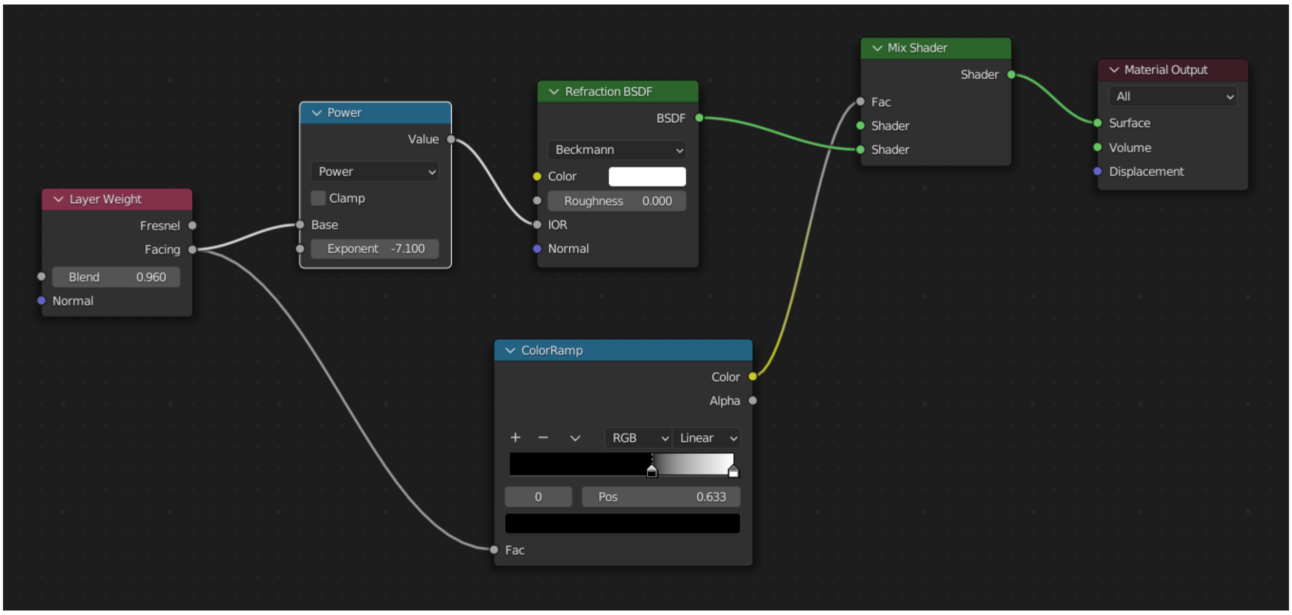
2 Black holes and their orbits

The two black holes have masses of $85M_o$ and $66M_o$ where M_o is the solar mass equal to $1.9885 \times 10^{30} kg$. I started of with two icospheres with their radii equal to their Schwarzschild radii. But I had to scale them because, well the values were quite high. This was the script I wrote to do the same.

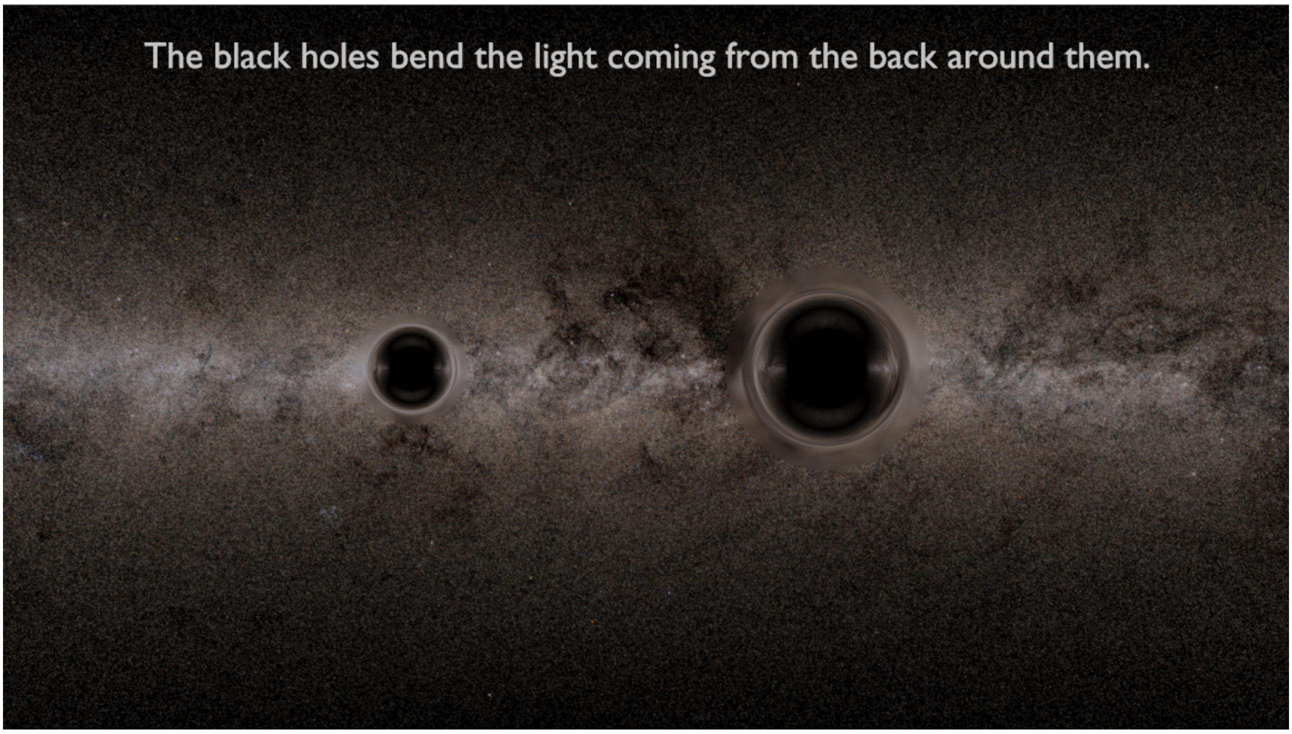
```
import bpy

factor = 0.00002 #scale factor
c = 3 * (10**8) #speed of light
G = 6.67 * (10**(-11)) #Gravitational constant
m_o = 1.9885 * (10**30) #Solar Mass
m1 = 85 * m_o
m2 = 66 * m_o
r_s1 = m1*(2*G/(c*c)) * factor #Schwarzschild radius
r_s2 = m2*(2*G/(c*c)) * factor

bpy.ops.mesh.primitive_ico_sphere_add(subdivisions=10, radius =
    r_s1, enter_editmode=False, align='WORLD', location=(0, 0, 0)
    )
bpy.ops.object.shade_smooth()

bpy.ops.mesh.primitive_ico_sphere_add(subdivisions=10, radius =
    r_s2, enter_editmode=False, align='WORLD', location=(0, 0, 0)
    )
bpy.ops.object.shade_smooth()
```

I used the feature of refraction to make the icospheres refract the image in the background to make it look as the black holes warp spacetime around it. This is the arrangement of nodes I used on a icospheres.[2]

I added an image in the background I got from this source background image. The final render looks like this. Here the icospheres bend the light coming from the background image,



thus giving us our required effect.



Now let's looks at the orbits of the black holes. I found a python module that plots the orbit of a black hole merging [3]. They had a set of reference coordinates stored in a CSV file. I imported the CSV file in Blender. Then I created two curves and assigned them the coordinates from the CSV file by appropriately scaling. This was the script I used.

```python
import numpy as np

data = np.genfromtxt(r"C:\Users\91987\Downloads\reftrajectories.csv"
    , delimiter=',')

```

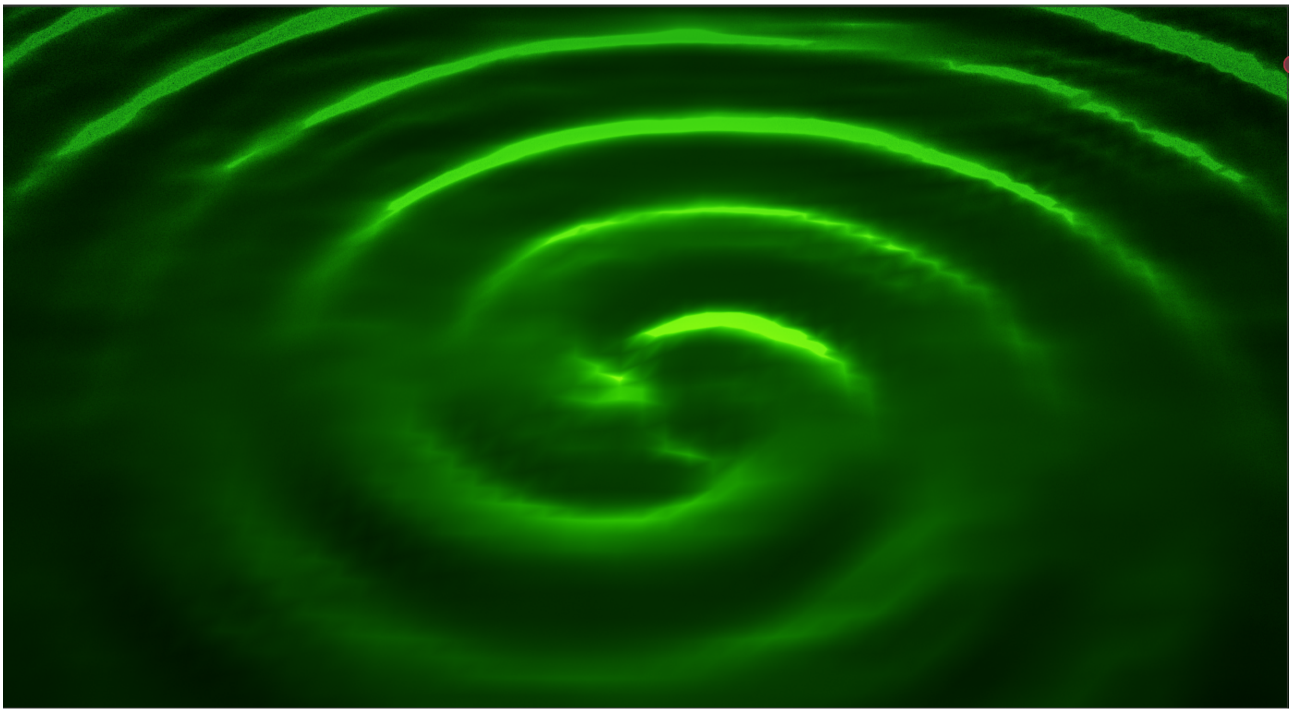
```python
curveData1 = bpy.data.curves.new('m1 trajectory', type='CURVE')
curveData1.dimensions = '2D'
curveData1.resolution_u = 2

curveData2 = bpy.data.curves.new('m2 trajectory', type='CURVE')
curveData2.dimensions = '2D'
curveData2.resolution_u = 2

# map coords to spline
polyline1 = curveData1.splines.new('NURBS')
polyline1.points.add(len(data))
for i, d in enumerate(data):
    x,y = d
    polyline1.points[i].co = ((100*m_o/m1)*x,(100*m_o/m1)*y,0,1)

polyline2 = curveData2.splines.new('NURBS')
polyline2.points.add(len(data))
for i, d in enumerate(data):
    x,y = d
    polyline2.points[i].co = ((-100*m_o/m2)*x,(-100*m_o/m2)*y,0,1)

# create Object
curveOB1 = bpy.data.objects.new('m1 trajectory', curveData1)
curveOB2 = bpy.data.objects.new('m2 trajectory', curveData2)
```

The trajectories didn't have as much rotations as I need So I had to add them manually. After this, I constrained the icospheres to follow the respective curves.

When the black holes come close to each other they form a outer layer around them as they merge. To make this effect, I used metaballs. I parented the metaballs to their respective icospheres. Initially the meta balls are at a very long distance out of the camera frame. Then after a specific frame, the meta balls appear. When they come close, they make as if the black holes are merging into each other. I simply just increased it transmission and changed a few minor things. This was the final result.

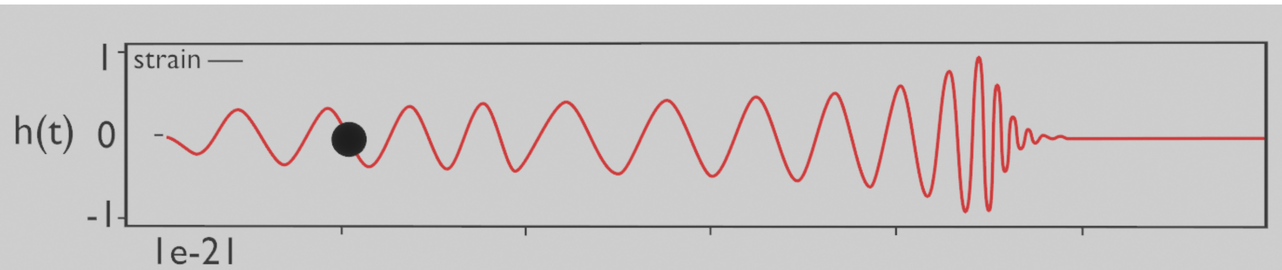
3 Gravitational Waves

Making the gravitational waves was a little tricky part. I used the idea that when a ball is dragged spirally on the surface of a liquid, it produces a similar effect as of gravitational waves. I added a grid and made it a canvas of dynamic paint. I didn't want my icospheres to literally move through the grid as I just needed the effect of grid. So I added two new spheres of same size as of the originals and I hid them in the final render. I made them the brushes of the dynamic paint and dragged them through the grid. For their motion I just constrained them to be the children of the original icospheres, thus giving us the required effect. I changed a few properties of the material. This was the final result.



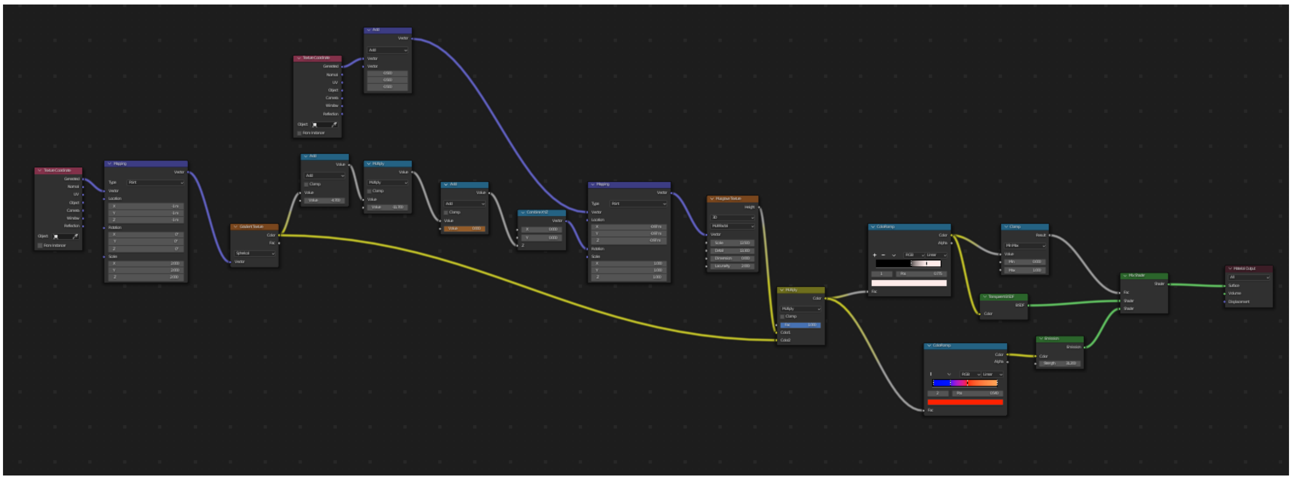
4 Strain Plot

The fractional change in the distance between two measurement points due to the deformation of space-time by a passing gravitational wave is called strain data. I have plotted a strain data plot similar to the one recorded by LIGO. I found the reference plot from here [1]. I manually fitted a curve on it and bevelled it. I added a small sphere, constrained it to the curved and made its position in sync with the gravitational waves simulation. This was the result,

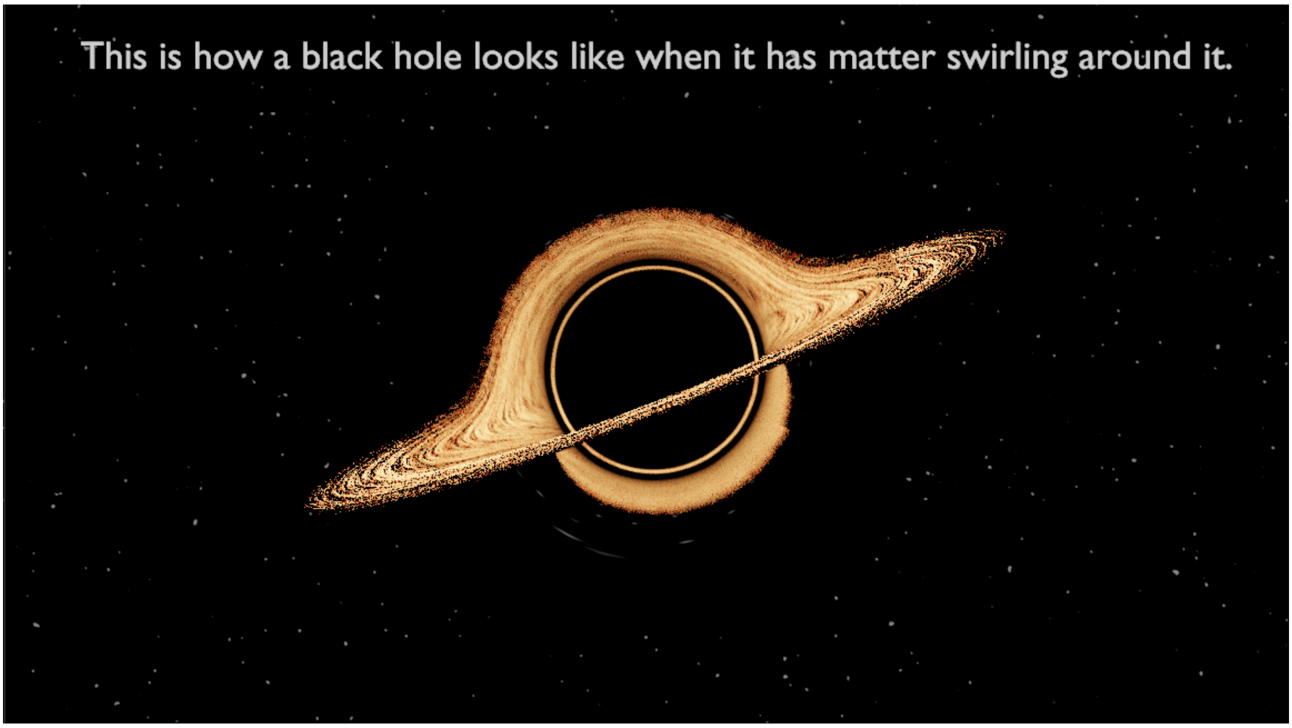


5 Accretion Disc

Finally we come to the accretion disc. I added a plane and used the following nodes arrangement,[2]



This was the result,



6 Lighting, Camera and Text

I used the area light for this. Adding the texts was simple. As all the motions were centered at the origin, I added an empty at the origin and constrained the camera to track it. So as the camera changed locations, it always looked at the empty.

7 Video Editing

Now that I had all the subparts, it was time to compile them in one. I used the video editing software in Blender. With some editing and adding transitions, I made the final render.

Referências

[1] *Guide to GW detections and noise*. Gravitational Wave Open Science Center. URL: https://www.gw-openscience.org/tutorials/.

[2] Kammerbild. *A interstellar-esque black hole in Blender*. Youtube. URL: https://www.youtube.com/watch?v=SnZEUpLE1AY.

[3] Vijay Varma, Leo C. Stein e Davide Gerosa. «The binary black hole explorer: on-the-fly visualizations of precessing binary black holes». Em: *Class. Quant. Grav.* 36.9 (2019). https://vijayvarma392.github.io/binaryBHexp/, p. 095007. DOI: 10.1088/1361-6382/ab0ee9. arXiv: 1811.06552 [astro-ph.HE].