Summary of Visit to the Center for Computational Astrophysics

Tom Abel

December 5 – December 14, 2024

1 Overview

From December 5th to December 14th, 2024, Tom Abel visited the Center for Computational Astrophysics (CCA) in New York City. During the visit, he participated in two workshops:

- Machine learning techniques in galaxy modeling.
- Cosmic dawn science, focusing on early galaxies and primordial star formation.

Tom also presented a talk at the galaxy formation meeting, introducing his new digital twin Universe project. This initiative aims to develop differentiable physical renderers for galaxies and apply them to reconstruct three-dimensional galaxy models from observational data. The project intends to leverage advancements in Gaussian splat modeling and rapid machine learning techniques.

2 Collaborations and Outcomes

During his visit, Tom engaged in discussions with various CCA members and visitors, including:

- Julianne Dalcanton
- Greg Bryan
- Rachel Somerville
- Romain Teyssier
- Jonathan Tan
- Mordecai Mac-Low
- Philip Mocz

- Keaton Burns
- Lehman Garrison

A significant research outcome from these discussions was the development of a prototype code demonstrating substantial speed improvements in computing two-point correlation functions at large radii using Fast Fourier Transforms (FFTs).

3 Python Implementation

The following Python code defines two functions for smoothing images using a Fourier tophat and computing the correlation function of an image.

```
import numpy as np
from scipy import ndimage

def smooth_fourier_tophat(im, radius):
    # Applies a spherical tophat smoothing filter in
       Fourier space
    radii = np.array(radius)
    res = []
    imt = np.fft.fft2(im)
    for i in radii.flatten() * 2:  # Fourier space
       requires diameter
        imc = ndimage.fourier_ellipsoid(imt, size=i)
        imc = np.fft.ifft2(imc).real
        res.append(imc)
    return res[0] if radii.flatten().shape[0] == 1
       else res

def correlation_function_of_image(image, num_bins=30,
   correct=True, method="cic"):
    # Computes the two-point correlation function of
       an image
    Ndim = np.min(image.shape)
    radii = np.logspace(0.3, np.log10(Ndim // 3),
       num_bins)
    m = smooth_fourier_tophat(image, radii)
    nrad = np.concatenate(([0.], radii))
    xi = np.zeros(len(radii))
    sigma = np.zeros(len(radii))
    for i, r in enumerate(radii):
        rhob = (m[i] * nrad[i+1]**2 - m[max([i-1, 0])]
            * nrad[i]**2) / (nrad[i+1]**2 - nrad[i
           ]**2)
        xi[i] = (image * rhob).mean()
        sigma[i] = image.std()
    return nrad, m, xi, sigma
```

4 Functionality Explanation

4.1 smooth_fourier_tophat

This function applies a spherical tophat smoothing operation to an image in Fourier space. The input image is transformed using the Fast Fourier Transform (FFT), filtered using a Fourier-space ellipsoidal kernel, and then inverse transformed to obtain the smoothed image.

4.2 correlation_function_of_image

This function computes the two-point correlation function of an input image by applying a Fourier-based smoothing filter and calculating the density fluctuations. The result provides a statistical measure of the spatial distribution of pixel intensities in the image.