Lab #6 – Mapping the Milky Way

AST326 – April 05, 2019

Ayush Pandhi (ayush.pandhi@mail.utoronto.ca)

Group C: Ayush Pandhi and Hansen Jiang

1 Abstract

The 21-cm hydrogen emission line plays a crucial part in the study of galaxies and in this experiment radio data from a Small Radio Telescope (SRT) at McLennan Physical Laboratories for various longitudes along the galactic plane are analyzed. Systematic corrections are applied corresponding to the baseline noise from background emission, the AirSpy noise and broadband noise and power spectra are generated and discussed for each longitude interval observed. A map of flux in terms of radial velocity and longitude is created by combining all of the data sets at each longitude interval and furthermore a rotation curve of the Milky Way is then generated using the tangent-point method on the observed data. Additionally, the mass enclosed in computed based on the estimated radius and rotational velocities. The rotation curve and estimated mass enclosed is then analyzed to help describe the dynamics of the Milky Way and the presence of dark matter is discussed. Overall, the goal of this experiment was to better understand radio astronomy through 21-cm detection and Doppler shift and analyze dynamical properties of the Milky Way.

2 Introduction

The 21-centimeter line is an electromagnetic spectral line that is caused by neutral hydrogen atoms changing their energy state. Specifically, this radiation is a result of the transition between the two levels of the hydrogen ground state which have a hyperfine energy splitting caused by the spin of the electron and proton being aligned in one state and counter-aligned in the other. Thus, the transition between these states releases a photon with a characteristic wavelength, $\lambda = 21.1cm$, frequency, $\upsilon = 1420.4MHz$, and approximate energy, $E = 5.9\mu eV$ [2]. In most cases this transition would not be useful for physics because it is quantum mechanically forbidden as the excited state has a have-life on the scale of tens of millions of years. However, in astronomy there are key uses for this transition because a lot of astronomical objects, such as galaxies specifically, have extremely large quantities of hydrogen gas, allowing the usually faint signal to be easily detectable.

Since the 21-cm spectral line appears in the radio spectrum, it easy able to easily pass through dust clouds and the Earth's atmosphere, allowing for very precise measurements from ground based radio astronomy. This is particularly useful for studying galaxies, which as stated earlier are large reservoirs of hydrogen gas, because this spectral line is very narrow (since it is limited by the Uncertainty Principle) and allows for accurate measurements of Doppler shift [2]. Specifically for the Milky Way this makes it possible to study the relative velocities of the spiral arms and as a result get a better understanding of the motion of our galaxy, its rotation curve and its mass. This can also be done with other galaxies; however since they are very far away, determining the structure and previously stated parameters is harder and less accurate than for that of the Milky Way.

Apart from studying galaxies, this spectral line is also useful in cosmology; it is one of the only known ways to probe the universe in the time between recombination and reionization. Theoretically one can map the intensity of redshifted 21-cm emission from this age to understand the mater-power spectrum better and perhaps learn more about how the universe was re-ionized [2]. However the issue is that the redshift puts this emission into the 10-200 MHz range on Earth which is dominated by terrestrial radio noise.

The following experiment uses data gathered from the Small Radio Telescope (SRT) at McLennan Physical Laboratories for various longitudes in the galactic plane. Through a data correction process using off-plane observations as reference, the 21-cm emission signal at each observed longitude is plotted as a function of frequency and radial velocity. Using this information, a radial velocity map of the Milky Way is created and analyzed as well as the rotation curve of the galaxy within the solar circle (up to roughly 8 kpc). As a result, the mass enclosed as a function of distance from the center of the galaxy is also analyzed and discussed in relation to the expected outcome and the presence of dark matter.

3 Observation and Data

The data sets used for this experiment were taken by the authors using the SRT as mentioned in the previous section. The telescope is located in Toronto, Ontario, Canada at (43.6532° N, 79.3832° W) in terms of latitude and longitude and is at an altitude of roughly 116 m. The SRT is a radio satellite that provides continuum and spectral line observations in the L-band (1420 MHz). The dish has a 2.3 m diameter and uses an azimuth-elevation mount for pointing and tracking purposes. The focal length is roughly 0.857 m, a beam-width of 7^{o} and the focal length vs. diameter ratio is approximately 0.375 [4]. The telescope receiver uses an AirSpy and is located at the feed of the antenna. The AirSpy device has a tunable heterodyne receiver between 24-1800 MHz, which can record up to 10^{7} samples per second. Additionally, it also contains: a tunable Local Oscillator, a mixer, an Intermediate Frequency filter, a Rafael Tuner chip with three gain adjustments, a Low Noise Amplifier for increasing the strength of the input signal, a mixer amplifier to boost the mixed signal, and the Intermediate Frequency amplifier which boosts the resulting signal in the desired frequency band (specifications acquired from airspy.com). The set of data that will be used for this experiment specifically was taken on March 21, 2019 between 09:21 am and 11:02 am (EST) with a total collection time of approximately 101 minutes. The following table provides a log of the observations:

Date	Time [EST]	Observation Target	Longitude [°]	RA, DEC
2019/21/03	09:21 am	Baseline (Off Plane)	0.0	18.9, -35.3
2019/21/03	09:23 am	Galactic Plane	0.0	17.8, -28.9
2019/21/03	09:25 am	Baseline (Off Plane)	10.0	19.1, -26.1
2019/21/03	09:27 am	Galactic Plane	10.0	18.2, -20.3
2019/21/03	09:28 am	Baseline (Off Plane)	15.0	19.2, -21.5
2019/21/03	09:30 am	Galactic Plane	15.0	18.3, -15.9
2019/21/03	09:32 am	Baseline (Off Plane)	20.0	19.3, -16.9
2019/21/03	09:33 am	Galactic Plane	20.0	18.5, -11.5
2019/21/03	09:35 am	Baseline (Off Plane)	25.0	19.4, -12.2
2019/21/03	09:36 am	Galactic Plane	25.0	18.6, -7.0
2019/21/03	09:38 am	Baseline (Off Plane)	30.0	19.5, -7.6
2019/21/03	09:39 am	Galactic Plane	30.0	18.8, -2.6
2019/21/03	09:41 am	Baseline (Off Plane)	35.0	19.6, -2.9
2019/21/03	09:43 am	Galactic Plane	35.0	18.9, 1.9
2019/21/03	09:44 am	Baseline (Off Plane)	40.0	19.7, 1.8
2019/21/03	09:46 am	Galactic Plane	40.0	19.1, 6.3
2019/21/03	09:48 am	Baseline (Off Plane)	45.0	19.8, 6.4
2019/21/03	09:50 am	Galactic Plane	45.0	19.2, 10.7
2019/21/03	09:51 am	Baseline (Off Plane)	50.0	19.9, 11.0
2019/21/03	09:52 am	Galactic Plane	50.0	19.4, 15.2
2019/21/03	09:54 am	Baseline (Off Plane)	55.0	20.0, 15.7
2019/21/03	09:55 am	Galactic Plane	55.0	19.6, 19.6
2019/21/03	09:56 am	Baseline (Off Plane)	60.0	20.1, 20.3
2019/21/03	09:57 am	Galactic Plane	60.0	19.8, 23.9
2019/21/03	09:29 am	Baseline (Off Plane)	65.0	20.2, 21.0
2019/21/03	10:00 am	Galactic Plane	65.0	19.9, 28.3
2019/21/03	10:02 am	Baseline (Off Plane)	70.0	20.2, 22.7
2019/21/03	10:03 am	Galactic Plane	70.0	20.1, 32.5
2019/21/03	10:05 am	Baseline (Off Plane)	75.0	20.3, 23.4
2019/21/03	10:07 am	Galactic Plane	75.0	20.4, 36.7
2019/21/03	10:09 am	Baseline (Off Plane)	80.0	20.2, 26.5
2019/21/03	10:11 am	Galactic Plane	80.0	20.6, 40.8
2019/21/03	10:14 am	Baseline (Off Plane)	85.0	20.9, 45.0
2019/21/03	10:15 am	Galactic Plane	85.0	20.9, 44.6
2019/21/03	10:18 am	Baseline (Off Plane)	90.0	21.4, 53.5
2019/21/03	10:20 am	Galactic Plane	90.0	21.2, 48.5
2019/21/03	10:22 am	Baseline (Off Plane)	100.0	21.8, 60.2
2019/21/03	10:24 am	Galactic Plane	100.0	22.0, 55.2
2019/21/03	10:26 am	Baseline (Off Plane)	110.0	22.6, 66.0
2019/21/03	10:28 am	Galactic Plane	110.0	23.1, 60.3
2019/21/03	10:30 am	Baseline (Off Plane)	120.0	0.2, 71.3
2019/21/03	10:31 am	Galactic Plane	120.0	0.5, 62.8
2019/21/03	10:33 am	Baseline (Off Plane)	130.0	2.3, 72.5

2019/21/03	10:35 am	Galactic Plane	130.0	1.9, 62.1
2019/21/03	10:36 am	Baseline (Off Plane)	140.0	4.0, 69.2
2019/21/03	10:37 am	Galactic Plane	140.0	3.2, 58.3
2019/21/03	10:39 am	Baseline (Off Plane)	150.0	5.2, 62.6
2019/21/03	10:40 am	Galactic Plane	150.0	4.1, 52.4
2019/21/03	10:42 am	Baseline (Off Plane)	160.0	6.0, 54.6
2019/21/03	10:43 am	Galactic Plane	160.0	4.8, 45.2
2019/21/03	11:00 am	Baseline (Off Plane)	170.0	6.4, 46.3
2019/21/03	11:01 am	Galactic Plane	170.0	5.3, 37.4

Table 1: *McLennan SRT observation summary of 21-cm emission from the galactic plane.*

The data files from these observations are .pkl files (Python pickle files) that contain a power spectra of the L-band over a few seconds of time in a two dimensional format. The primary source of systemic error is from intrinsic noise of the AirSpy and feed as well as the background emission noise that are not from the primary observation target.

4 Data Reduction and Methods

It is important to note that the analysis in this experiment is done in galactic coordinates (Figure 1). This coordinate frame is characterized by the galactic latitude (b) above or below the galactic plane, the galactic longitude (ℓ) with the center of the galaxy being $\ell = 0^o$ [3]. In these coordinates, the Sun is at the center of the reference frame.

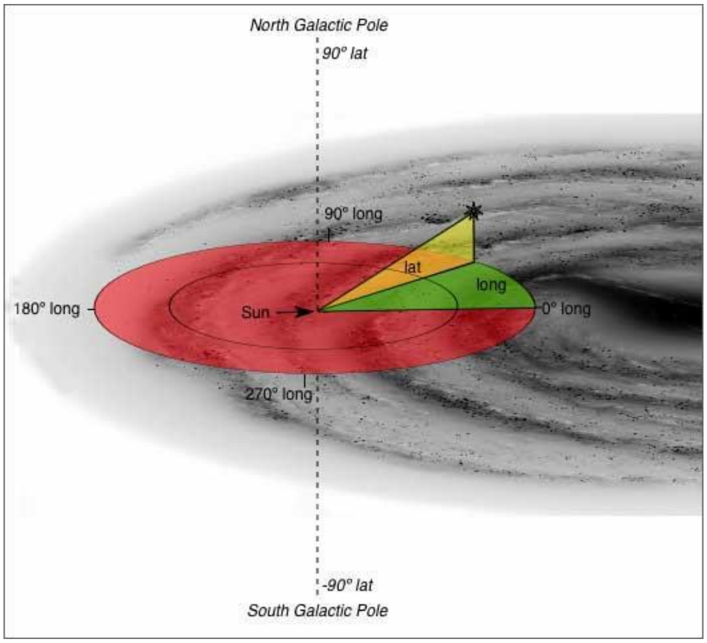

Figure 1: *A visual depicting how longitude and latitude are determined in the galactic coordinate frame* [3].

The first stage of this experiment involves correcting the 21-cm galactic plane data using the baseline that was observed at each longitude. This baseline characterises the noise coming from background sources and the AirSpy device; this is excess noise that must be removed as it is not a consequence of any physical phenomenon that is relevant to the experiment. This is done by first dividing out the baseline from the corresponding galactic plane data set which leaves a residual spectrum with a clear signal from the 21-cm spectral line. However, there is still some broadband noise in the residual spectra that can be removed to increase the signal to noise in the data. This is done by fitting a polynomial to the residuals that do not correspond to the signal (i.e. everything except the frequency range of 1419.9 – 1420.9 MHz) and subtracting the fit from the total residual data. This corrected data provides a clear spectrum with a visible 21-cm signal that can then be used for analysis. The details of how this was done can be found in the code in the Appendix.

$$D_{blr} = \frac{D_{raw}}{D_{bl}} \quad \longrightarrow \quad D_{dt} = D_{blr} - P_{bb} \tag{1}$$

Where D_{blr} is the data with the baseline removed, D_{raw} is the raw data, D_{bl} is the baseline data, D_{dt} is the de-trended data after the broadband noise is removed and P_{bb} is the polynomial fit to the broadband noise. This data reduction process is applied to each data set using their corresponding baseline data as seen in Table 1.

After the data reduction process we are left with a corrected power spectrum in the L-band at various longitudes along the galactic plane. The next step is to convert the frequency axis to radial velocity using Doppler shift analysis. The relationship between radial velocity and frequency is given by [5]:

$$\frac{v_{rad}}{c} = \frac{f_0^2 - f^2}{f_0^2 + f^2} \tag{2}$$

A linear expansion approximation where $f_o \longrightarrow 0$ to the equation gives [5]:

$$\frac{v_{rad}}{c} \cong \frac{f_o - f}{f} = z \tag{3}$$

Where v_{rad} is the radial velocity, f is the observed frequency, f_0 is the expected frequency (in this case 1420.4 MHz), c is the speed of light constant and z is defined as the redshift. Then the power spectra at each observed longitude are plotted against radial velocity instead of against frequency to give a better insight into the physical phenomena causing the 21-cm emission in the Milky Way.

This allows us to plot a two dimensional power spectrum of flux as a function of radial velocity and longitude to create a map of the radial velocities in the Milky Way. Additionally, each spectrum at various longitudes is manually probed to measure the radial velocity at the peaks created by the hydrogen emission. A table with the radial velocities of all observed emission peaks throughout the data can be found in the Appendix (Table 2). This information is then used for further analysis using the tangent-point method to estimate rotation speed as a function of longitude (and thus radial distance from the galactic center) [5]. Note that this is only possible for galactic longitudes between 0° – 90° as the tangent-point method in not applicable beyond this limit. The relationship between longitude and radial distance from the center of the Milky Way is given by [1]:

$$R = R_{sun}\sin(\ell) \tag{4}$$

Where R is the radial distance from the center of the galaxy, R_{sun} is this distance specifically for the sun (roughly 8 kpc) and ℓ is the galactic longitude in radians. Furthermore, the rotational velocity for a given longitude and corresponding radial velocity is given by [1]:

$$v_{rot} = v_{rad} + v_{sun}\sin(\ell) \tag{5}$$

Where v_{rot} is the rotational velocity at a given point and v_{sun} is the average rotational velocity at the solar circle (roughly 220 km/s). This then allows us to generate plots of the rotation curve of the Milky Way. Additionally, analysis on the mass enclosed at certain galactic radii is also done using the relationship [1]:

$$M_{enc} = v_{rot}^2 R/G \tag{6}$$

Where M_{enc} is the mass enclosed within a certain radius around the center of the galaxy and G is the gravitational constant. The results of this analysis are then compared with the expected mass of the galaxy and any discrepancies are discussed in section 6.

The largest source of error throughout the analysis and modeling section arises from the background noise in the data and the noise from the AirSpy device. This error is dealt with by corrections on the baseline and the broadband polynomial fit as mentioned earlier in this section.

5 Data Analysis and Modeling

The detailed python code for generating all figures and data analysis in this section can be found in the Appendix. The plots in this section will show the data analysis applied to the $\ell = 45^o$ data set specifically but in reality this same procedure was applied to all the previously listed data sets. To start, the baseline is removed from the corresponding galactic plane data set by dividing it out according to equation (1). This reduces the systematic error in the data by removing most of the background noise from other sources and the AirSpy. The raw data is plotted alongside the corrected data residuals as waterfall plots (Figure 2).

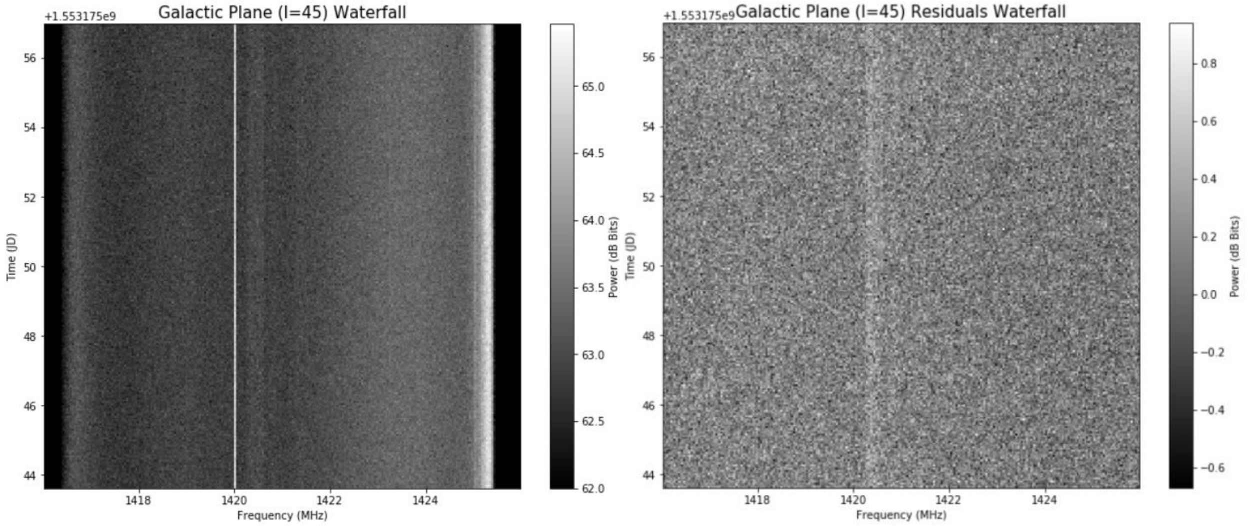

Figure 2: *a) (Left) Waterfall plot of the observed power spectrum in the L-band over time for the Galactic Plane observed at a longitude of 45°, b) (Right) Waterfall plot of the residuals after dividing out the baseline from the galactic plane observation.*

An additional correction is applied by removing the remaining broadband noise in the residuals. As stated in the previous section, following equation (1), this is done by fitting a polynomial to the noise and subtracting it out from the total residuals to obtain an accurate power spectrum for the 21-cm emission line in the data. The residuals power spectrum is obtained by taking the mean over the time axis for each frequency bin and the polynomial fit for the broadband noise is plotted below (Figure 3).

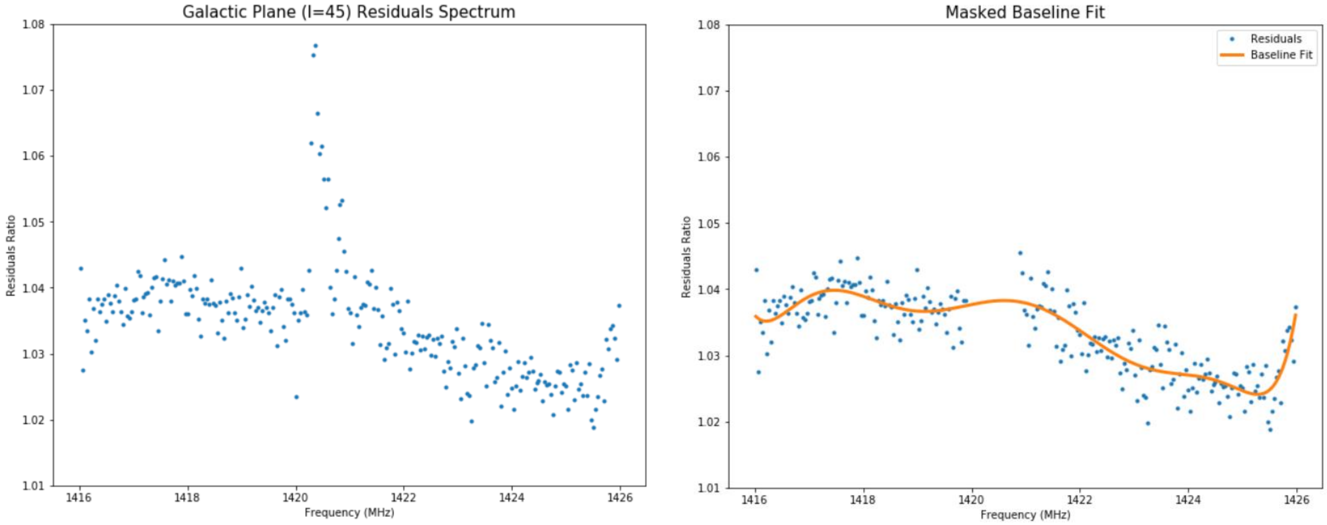

Figure 3: *a) (Left) The mean along each frequency bin of the waterfall plotted as a spectrum of the residuals from the previous plot, b) (Right) A polynomial fit to the broadband noise after removing the 21-cm signal near 1420.4 MHz.*

This corrected data then shows fractional power difference as a function of frequency to more clearly see the 21-cm spectral line. As per equation (2), the frequency axis is then converted into a radial velocity axis using the Doppler shift relationship with $f_o = 1420.4\ MHz$. The fractional power difference is then plotted with dual axes for frequency and radial velocity where it is easy to analyze the velocity of the emission peaks in the data (Figure 4).

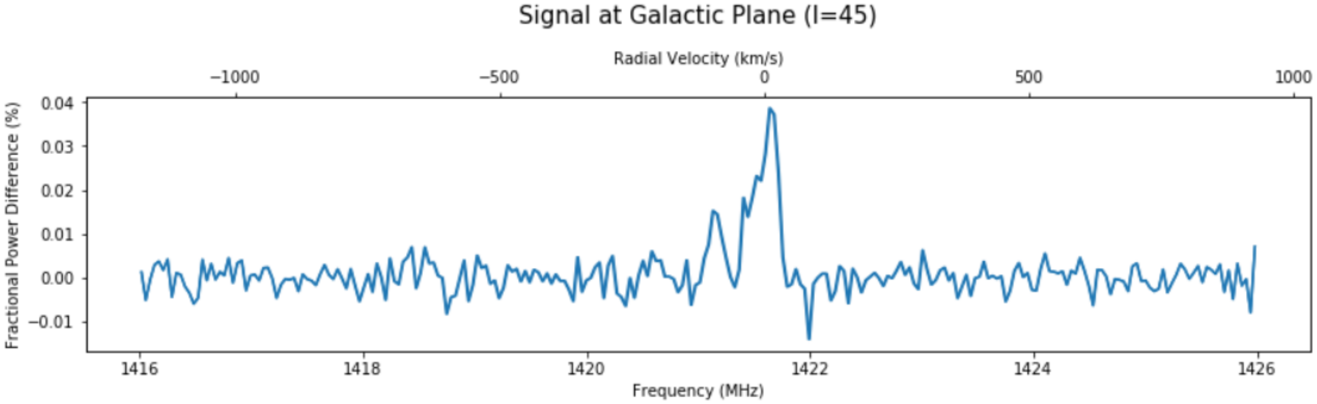

Figure 4: *Fractional power difference in the data plotted along frequency/radial velocity, the peaks in the data are the Doppler shifted 21-cm emission lines from the galactic plane.*

Using the corrected power spectra of all galactic plane observations at various longitudes, a map of observed flux as a function of radial velocity and longitude can be generated for the Milky Way (Figure 5). Since there are some longitude intervals missing (especially beyond 90°), there is no data in those areas. A rough approximation can be inferred in those regions using a smoothing technique (detailed in Appendix) to provide a clearer (though less accurate) map with the limited observing power available.

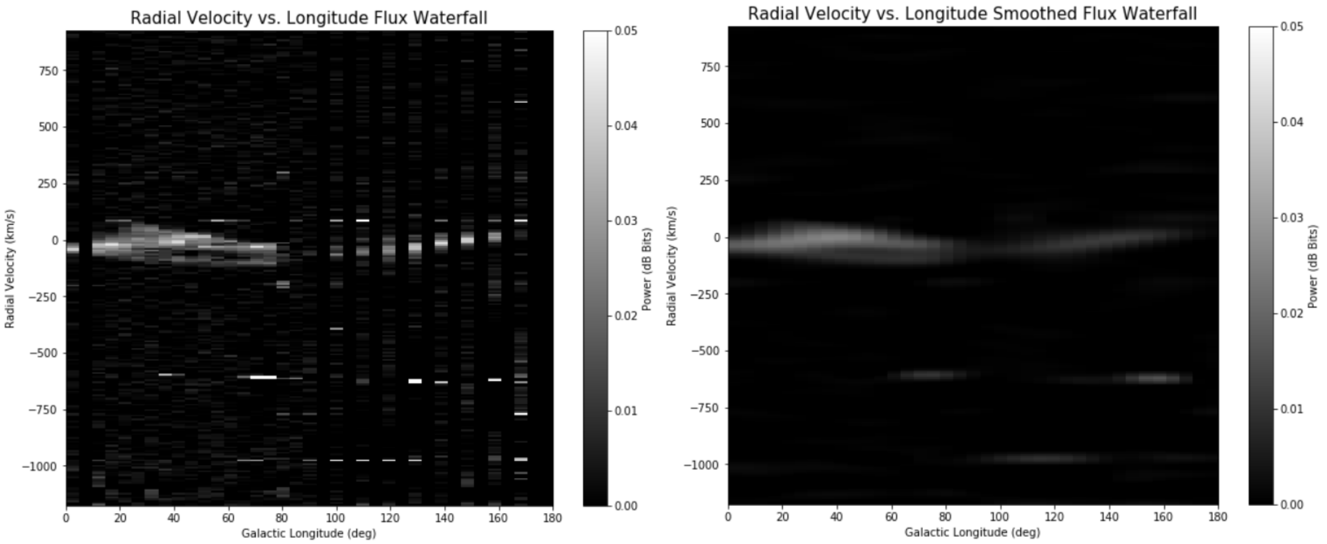

Figure 5: *a) (Left) Waterfall plot of flux in terms of radial velocity and galactic longitude for the Milky Way (after corrections), b) (Right) The same plot as a) but with a smoothing algorithm applied to roughly extrapolate the data in missing regions.*

Furthermore, equations (4) and (5) are then used to compute the galactic radius and rotational velocities based on the tangent-method for longitudes ranging between 0° to 90°. This information allows for a rough rotational curve for the Milky Way based on the primary peak (the largest and most prominent peak that is consistent throughout the data) and for secondary/tertiary peaks as well (smaller peaks that only appear for small longitude window) (Figure 6). Note that for the rotation curve only longitudes 10° to 80° were considered as the data around 0° and 90° would not give very accurate line of sight velocities.

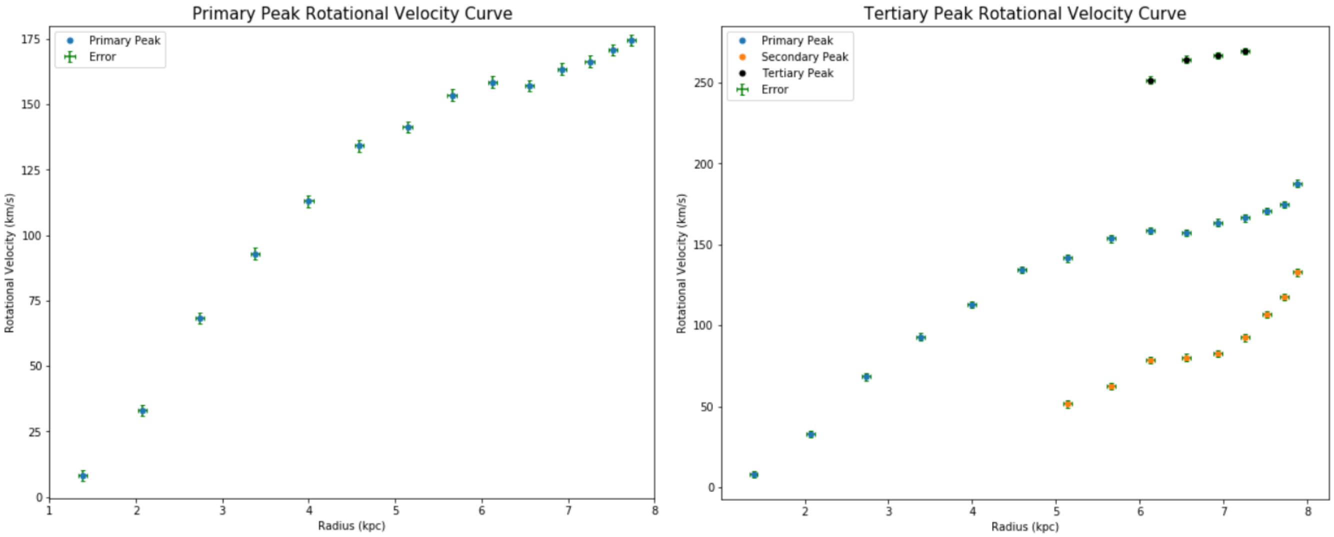

Figure 6: *a) (Left) Rotation curve for the Milky Way based on the primary peak throughout the data, b) (Right) Rotation curves for all three peaks observed throughout the data, note that the secondary and tertiary peaks only appear in small sections of the data and thus the analysis is focused on the primary peak.*

Finally, the enclosed mass was computed as a function of galactic radius and plotted using equation (6) (Figure 7). The estimated mass enclosed in $R \cong 7.9 \pm 0.1\ kpc$ was $M_{enc} \cong 6.5 \pm 0.1\ x\ 10^{10}\ M_{sun}$ whereas the expected value of the mass within the solar circle is $M_{enc} \cong 8.8\ x\ 10^{10}\ M_{sun}$ [1].

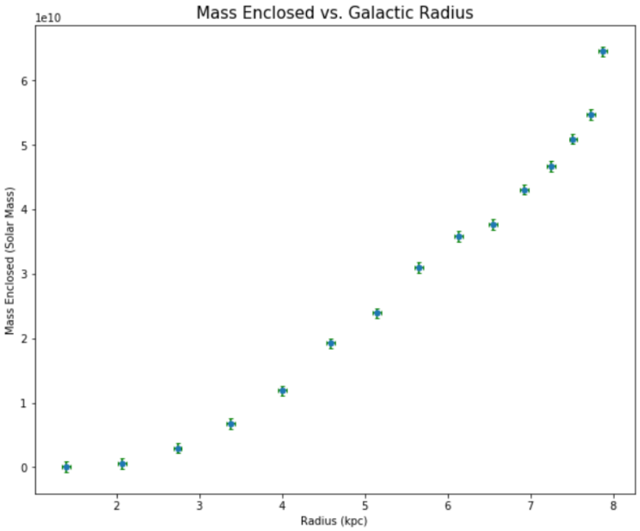

Figure 7: *A plot of the estimated mass enclosed given a certain galactic radius around the center of the galaxy.*

6 Discussion

Firstly, discussing the systemic error correction by dividing corresponding off-plane baseline measurements from each respective galactic plane measurement; it is clear from Figure 2 that this calibration reduces the overall noise in the data. Specifically, the bright section at exactly 1420 MHz created by the AirSpy is removed and the overall background emission in addition to the AirSpy noise is reduced to show a clear 21-cm line. Just from visual observation it is clear that this spectral line has been Doppler shifted to a slightly higher frequency as it does not line up with the expected 1420.4 MHz.

Additionally, to remove the broadband noise, as seen in Figure 3, a 9th degree polynomial is fit to the background residuals and that is subtracted from the data. This leads to Figure 4 in which the spectral line peaks are clearly distinguishable from the background noise. In the specific case presented ($\ell = 45^o$) it can be seen that there are two clear peaks in the data: the primary peak's centroid lines up with an approximate line of sight velocity of -2.0 km/s and the centroid of the secondary peak has an approximate line of sight velocity of -93.0 km/s. These peaks arise from two distinct spiral arms, which have a large amount of hydrogen gas, along the line of sight at $\ell = 45^o$. From these observations it is clear that both of the arms are moving towards us, one slowly at 2.0 km/s and the other quite quickly at 93.0 km/s. This analysis is repeated for each observation set and the results can be found in Table 2 in the Appendix.

Figure 5 shows a flux map in radial velocity and longitude space; it is smoothed to more clearly (although also less accurately) show the approximate 21-cm emission flux in this space. The bright streams represent the hydrogen line emissions from the galactic plane and at some areas it can be seen that there are multiple distinct streams correlating to multiple peaks (and thus spiral arms) at certain longitudes. Specifically, this can be best seen between roughly $\ell = 40^o$ and $\ell = 80^o$, which agrees with the observations as these two streams in this specific longitude range seem to also be around roughly 0.0 km/s and -100 km/s respectively. Unfortunately, with this limited data set it was not possible to create a more detailed map to further analyze the galactic structure using this technique.

From Figure 6, the rotation curve generated from this data is analyzed. The analysis focuses on the primary peak as that has the largest quantity of data and is consistently visible between the observed range of $\ell = 10^o$ and $\ell = 80^o$. The rotation curve seems to be roughly linear, which seems to match the solid body approximation. However, there is a slight levelling off at around 6 kpc which would be consistent with previously measured galactic rotation curves through other surveys. The rotation curve again rises after that point though instead of remaining flat and thus from this data one would conclude the rotation curve of the Milky Way follows a roughly solid body approximation where $v_{rot} \propto R$. Additionally, from Figure 7 one can also see that M_{enc} depends roughly linearly on R. Specifically, this lets us compute that $M_{enc} \cong 6.5 \pm 0.1\, x\, 10^{10}\, M_{sun}$ at $R \cong 7.9 \pm 0.1\, kpc$ which is very close to the expected mass enclosed at the solar radius of $M_{enc} \cong 8.8\, x\, 10^{10}\, M_{sun}$ [1]. This would thus also be in agreement with the presence of dark matter in the galaxy which includes additional non-luminous mass.

7 References

[1] Carroll B. W., Ostlie D. A. 1996, Weber State University, Ogden, Utah. “An Introduction to Modern Astrophysics”

[2] Condon J. 2016, National Radio Astronomy Observatory (NRAO), Charlottesville, Virginia. “Essential Radio Astronomy – The HI 21cm Line”, https://www.cv.nrao.edu/course/astr534/HILine.html

[3] McCollum J. 2009, Houston Astronomical Society (HAS), Houston, Texas. “Celestial Coordinate Systems”, https://www.astronomyhouston.org/sites/default/files/presentations/HAS_Novice_Program_Celestial_Coordinates.pdf

[4] MIT Haystack Observatory, Westford, Massachusetts. “Small Radio Telescope (SRT)”, https://www.haystack.mit.edu/edu/undergrad/srt/oldsrt.html

[5] Ransom S. 2018, National Radio Astronomy Observatory (NRAO), Charlottesville, Virginia. “Essential Radio Astronomy – Spectral Lines”, https://www.cv.nrao.edu/~sransom/web/Ch7.html#S8

8 Appendix

8.1 Radial Velocity of the Centroids of Observed Emission Peaks at Various Longitude

Longitude [^{o}]	Primary Peak v_{rad} [km/s]	Secondary Peak v_{rad} [km/s]	Tertiary Peak v_{rad} [km/s]
0.0	-40.0	-	-
5.0	N/A	N/A	N/A
10.0	-30.0	-	-
15.0	-24.0	-	-
20.0	-7.0	-	-
25.0	0.0	-	-
30.0	3.0	-	-
35.0	8.0	-	-
40.0	0.0	-90.0	-
45.0	-2.0	-93.0	-
50.0	-10.0	-90.0	83.0
55.0	-23.0	-100.0	84.0
60.0	-27.0	-108.0	76.0
65.0	-33.0	-107.0	70.0
70.0	-36.0	-100.0	-
75.0	-38.0	-95.0	-

80.0	-29.0	-84.0	-
85.0	-	-	-
90.0	-	-	-

Table 2: *Observed radial velocities for the centroids of the emission peaks corresponding to longitude. Note that there was no data observed at all for longitude of 5^o and for $85^o/90^o$ there were no visible peaks in the corrected data. In general the primary peaks were the largest in amplitude followed by the secondary peaks and then the tertiary peaks.*

8.2 Python Code for Data Analysis of $\boldsymbol{\ell = 45^o}$ Data Set (Figures 2-4)

Note that this section of code walks through the detailed steps done to analyze the data set for a longitude of 45°. A function was then defined to quickly process this analysis for the remaining data sets which can be seen in section 8.3.

```
#AST326 Lab 6 – Mapping the Milky Way
#Ayush Pandhi (1003227457)
#April 05, 2019


#Importing required modules

import numpy as np

import matplotlib.pyplot as plt

import astropy as ast

from astropy import coordinates

from astropy.time import Time

from scipy.optimize import curve_fit

import struct

import pickle

from scipy.ndimage.filters import gaussian_filter


#Loading l=45 degree data

with open('GroupC_2019_03_21/MP_20190321-095005.pkl', 'rb') as f1:
    gal_45 = pickle.load(f1, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-094822.pkl', 'rb') as f2:
    gal_45_off = pickle.load(f2, encoding='iso-8859-1')

#Break up data into variables
f = gal_45['freqs']
t = gal_45['times']
d = gal_45['data'][:,:,0]
d_off = gal_45_off['data'][:,:,0]

#Waterfall plot for both galactic plane and off plane
plt.figure(figsize=(10,8))
plt.imshow(10*np.log10(d_off), aspect='auto', extent=[np.amin(f), np.amax(f), np.amin(t), np.amax(t)], cmap='gist_gray', vmin=62)
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Time (JD)', fontsize=10)
plt.title('Off-Plane Baseline (l=45) Waterfall', fontsize=15)
plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)
plt.show()

plt.figure(figsize=(10,8))
plt.imshow(10*np.log10(d), aspect='auto', extent=[np.amin(f), np.amax(f), np.amin(t), np.amax(t)], cmap='gist_gray', vmin=62)
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Time (JD)', fontsize=10)
plt.title('Galactic Plane (l=45) Waterfall', fontsize=15)
plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)
plt.show()

#Plotting On and Off Galaxy Spectrums
plt.figure(figsize=(10,8))
plt.plot(f, 10*np.log10(d.mean(axis=0)))
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Power (dB bits)', fontsize=10)
plt.title('Galactic Plane (l=45) Spectrum', fontsize=15)
plt.ylim(60, 65)
plt.show()

plt.figure(figsize=(10,8))
plt.plot(f, 10*np.log10(d_off.mean(axis=0)))
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Power (dB bits)', fontsize=10)
plt.title('Off-Plane Baseline (l=45) Spectrum', fontsize=15)
plt.ylim(60, 65)
plt.show()

#Getting residuals and plotting
dnorm = d/d_off
```

```
plt.figure(figsize=(10,8))
plt.imshow(10*np.log10(dnorm), aspect='auto', extent=[np.amin(f), np.amax(f), np.amin(t), np.amax(t)], cmap='gist_gray')
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Time (JD)', fontsize=10)
plt.title('Galactic Plane (l=45) Residuals Waterfall', fontsize=15)
plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)
plt.show()

plt.figure(figsize=(10,8))
plt.plot(f, dnorm.mean(axis=0), '.')
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Residuals Ratio', fontsize=10)
plt.title('Galactic Plane (l=45) Residuals Spectrum', fontsize=15)
plt.ylim(1.01, 1.08)
plt.show()

#De-trending data
f21cm = 1420.4
q = np.where(np.abs(f-f21cm)>0.5)
deg = 9
dd = dnorm.mean(axis=0)
fit = np.polyfit(f[q]-f21cm, dd[q], deg)
```

```
#Plotting residuals with 21cm removed

plt.figure(figsize=(10,8))

plt.plot(f[q], dd[q], '.')

plt.xlabel('Frequency (MHz)', fontsize=10)

plt.ylabel('Residuals Ratio', fontsize=10)

plt.title('Residuals Spectrum (21cm Removed)', fontsize=15)

plt.ylim(1.01, 1.08)

plt.show()


#Plotting residuals polynomial fit

plt.figure(figsize=(10,8))

plt.plot(f[q], dd[q], '.', label='Residuals')

plt.plot(f, np.polyval(fit, f-f21cm), linewidth=3, label='Baseline Fit')

plt.xlabel('Frequency (MHz)', fontsize=10)

plt.ylabel('Residuals Ratio', fontsize=10)

plt.title('Masked Baseline Fit', fontsize=15)

plt.ylim(1.01, 1.08)

plt.legend()

plt.show()


#Plotting the fractional power difference with removed noise

ddd = dd-np.polyval(fit, f-f21cm)

plt.figure(figsize=(14.1,3))
plt.plot(f, ddd, '-')
plt.xlabel('Frequency (MHz)', fontsize=10)
plt.ylabel('Fractional Power Difference (%)', fontsize=10)
plt.title('Signal at Galactic Plane (l=45)', fontsize=15)
plt.show()

#Getting radial velocities from frequencies
v = (((f21cm**2 - f**2)/(f21cm**2 + f**2))*(3*(10**8)))*0.001

#Re-plotting with an additional radial velocity axis
fig, ax1 = plt.subplots(figsize=(12,4))
ax1.set_title('Signal at Galactic Plane (l=45)', fontsize=15, y=1.25)
ax1.set_xlabel('Frequency (MHz)', fontsize=10)
ax1.set_ylabel('Fractional Power Difference (%)', fontsize=10)
ax1.plot(f, ddd[::-1], '-')
ax1.tick_params(axis='x')
ax2 = ax1.twiny()
ax2.set_xlabel('Radial Velocity (km/s)', fontsize=10)
ax2.plot(v, ddd, '-')
ax2.tick_params(axis='x')
fig.tight_layout()
plt.show()
```

8.3 Python Code for Repeating Analysis for all Data Sets

The code below repeats the analysis above for all of the data sets but with fewer plots and returns the corrected data for further use. It also formats important plots with text for ease of access and analysis. Note that the longitudes are mapped in intervals of 5^o in longitude; for any interval that was not able to be observed, the data is set to an array of zeros in those regions.

```
#Loading all data sets
with open('GroupC_2019_03_21/MP_20190321-092330.pkl', 'rb') as f1:
    gal_00 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-092131.pkl', 'rb') as f2:
    gal_00_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-092718.pkl', 'rb') as f1:
    gal_10 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-092555.pkl', 'rb') as f2:
    gal_10_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-093026.pkl', 'rb') as f1:
    gal_15 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-092859.pkl', 'rb') as f2:
    gal_15_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-093346.pkl', 'rb') as f1:
    gal_20 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-093225.pkl', 'rb') as f2:
    gal_20_off = pickle.load(f2, encoding='iso-8859-1')
```

```
with open('GroupC_2019_03_21/MP_20190321-093657.pkl', 'rb') as f1:
    gal_25 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-093536.pkl', 'rb') as f2:
    gal_25_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-093959.pkl', 'rb') as f1:
    gal_30 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-093845.pkl', 'rb') as f2:
    gal_30_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-094316.pkl', 'rb') as f1:
    gal_35 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-094145.pkl', 'rb') as f2:
    gal_35_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-094633.pkl', 'rb') as f1:
    gal_40 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-094455.pkl', 'rb') as f2:
    gal_40_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-095005.pkl', 'rb') as f1:
    gal_45 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-094822.pkl', 'rb') as f2:
    gal_45_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-095232.pkl', 'rb') as f1:
    gal_50 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-095125.pkl', 'rb') as f2:
    gal_50_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-095511.pkl', 'rb') as f1:
    gal_55 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-095409.pkl', 'rb') as f2:
    gal_55_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-095745.pkl', 'rb') as f1:
    gal_60 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-095639.pkl', 'rb') as f2:
    gal_60_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-100045.pkl', 'rb') as f1:
    gal_65 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-095932.pkl', 'rb') as f2:
    gal_65_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-100339.pkl', 'rb') as f1:
    gal_70 = pickle.load(f1, encoding='iso-8859-1')
```

```
with open('GroupC_2019_03_21/MP_20190321-100217.pkl', 'rb') as f2:
    gal_70_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-100725.pkl', 'rb') as f1:
    gal_75 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-100558.pkl', 'rb') as f2:
    gal_75_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-101123.pkl', 'rb') as f1:
    gal_80 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-100935.pkl', 'rb') as f2:
    gal_80_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-101546.pkl', 'rb') as f1:
    gal_85 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-101410.pkl', 'rb') as f2:
    gal_85_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-102023.pkl', 'rb') as f1:
    gal_90 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-101840.pkl', 'rb') as f2:
    gal_90_off = pickle.load(f2, encoding='iso-8859-1')


with open('GroupC_2019_03_21/MP_20190321-102434.pkl', 'rb') as f1:
    gal_100 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-102253.pkl', 'rb') as f2:
    gal_100_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-102805.pkl', 'rb') as f1:
    gal_110 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-102645.pkl', 'rb') as f2:
    gal_110_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-103125.pkl', 'rb') as f1:
    gal_120 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-103005.pkl', 'rb') as f2:
    gal_120_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-103507.pkl', 'rb') as f1:
    gal_130 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-103340.pkl', 'rb') as f2:
    gal_130_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-103759.pkl', 'rb') as f1:
    gal_140 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-103658.pkl', 'rb') as f2:
    gal_140_off = pickle.load(f2, encoding='iso-8859-1')
```

```
with open('GroupC_2019_03_21/MP_20190321-104050.pkl', 'rb') as f1:
    gal_150 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-103937.pkl', 'rb') as f2:
    gal_150_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-104329.pkl', 'rb') as f1:
    gal_160 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-104230.pkl', 'rb') as f2:
    gal_160_off = pickle.load(f2, encoding='iso-8859-1')

with open('GroupC_2019_03_21/MP_20190321-110158.pkl', 'rb') as f1:
    gal_170 = pickle.load(f1, encoding='iso-8859-1')
with open('GroupC_2019_03_21/MP_20190321-110044.pkl', 'rb') as f2:
    gal_170_off = pickle.load(f2, encoding='iso-8859-1')

#Defining a function to do the analysis on a set of data
def analysis(data, data_off, f, v, f21cm, l):
    d = data['data'][:,:,0]
    d_off = data_off['data'][:,:,0]
    dnorm = d/d_off
    q = np.where(np.abs(f-f21cm)>0.5)
    deg = 9
    dd = dnorm.mean(axis=0)
    fit = np.polyfit(f[q]-f21cm, dd[q], deg)
    ddd = dd-np.polyval(fit, f-f21cm)

    print('------------------------------------------------------------------------------------------------------------------------------')
    print('                                        ANALYSIS FOR GALACTIC LONGITUDE ' + str(l))
    print('------------------------------------------------------------------------------------------------------------------------------')

    plt.figure(figsize=(10,8))
    plt.imshow(10*np.log10(dnorm), aspect='auto', extent=[np.amin(f), np.amax(f), np.amin(t), np.amax(t)], cmap='gist_gray')
    plt.xlabel('Frequency (MHz)', fontsize=10)
    plt.ylabel('Time (ns)', fontsize=10)
    plt.title('Galactic Plane Residuals Waterfall (l=' + str(l) + ')', fontsize=15)
    plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)
    plt.show()

    print()

    plt.figure(figsize=(10,8))
    plt.plot(f, dnorm.mean(axis=0), '.')
    plt.xlabel('Frequency (MHz)', fontsize=10)
    plt.ylabel('Residuals Ratio', fontsize=10)
    plt.title('Galactic Plane Residuals Spectrum (l=' + str(l) + ')', fontsize=15)
    plt.show()
```

```
print()

fig, ax1 = plt.subplots(figsize=(12,4))
ax1.set_title('Signal at Galactic Plane (l=' + str(l) + ')', fontsize=15, y=1.25)
ax1.set_xlabel('Frequency (MHz)', fontsize=10)
ax1.set_ylabel('Fractional Power Difference (%)', fontsize=10)
ax1.plot(f, ddd[::-1], '-')
ax1.tick_params(axis='x')
ax2 = ax1.twiny()
ax2.set_xlabel('Radial Velocity (km/s)', fontsize=10)
ax2.plot(v, ddd, '-')
ax2.tick_params(axis='x')
fig.tight_layout()
plt.show()

print()

#Plotting the fractional power difference with removed noise
plt.figure(figsize=(10,5))
plt.plot(v[85:140], ddd[85:140], '-')
plt.xlabel('Radial Velocity (km/s)', fontsize=10)
plt.ylabel('Fractional Power Difference (%)', fontsize=10)
plt.title('Signal at Galactic Plane (l=' + str(l) + ')', fontsize=15)
```

```
    plt.show()

    print()

    return ddd

#Performing the analysis for the entire data set
ddd_00 = analysis(gal_00, gal_00_off, f, v, 1420.4, 0)
ddd_05 = np.zeros(256)
ddd_10 = analysis(gal_10, gal_10_off, f, v, 1420.4, 10)
ddd_15 = analysis(gal_15, gal_15_off, f, v, 1420.4, 15)
ddd_20 = analysis(gal_20, gal_20_off, f, v, 1420.4, 20)
ddd_25 = analysis(gal_25, gal_25_off, f, v, 1420.4, 25)
ddd_30 = analysis(gal_30, gal_30_off, f, v, 1420.4, 30)
ddd_35 = analysis(gal_35, gal_35_off, f, v, 1420.4, 35)
ddd_40 = analysis(gal_40, gal_40_off, f, v, 1420.4, 40)
ddd_45 = analysis(gal_45, gal_45_off, f, v, 1420.4, 45)
ddd_50 = analysis(gal_50, gal_50_off, f, v, 1420.4, 50)
ddd_55 = analysis(gal_55, gal_55_off, f, v, 1420.4, 55)
ddd_60 = analysis(gal_60, gal_60_off, f, v, 1420.4, 60)
ddd_65 = analysis(gal_65, gal_65_off, f, v, 1420.4, 65)
ddd_70 = analysis(gal_70, gal_70_off, f, v, 1420.4, 70)
ddd_75 = analysis(gal_75, gal_75_off, f, v, 1420.4, 75)
ddd_80 = analysis(gal_80, gal_80_off, f, v, 1420.4, 80)
```

```
ddd_85 = analysis(gal_85, gal_85_off, f, v, 1420.4, 85)

ddd_90 = analysis(gal_90, gal_90_off, f, v, 1420.4, 90)

ddd_95 = np.zeros(256)

ddd_100 = analysis(gal_100, gal_100_off, f, v, 1420.4, 100)

ddd_105 = np.zeros(256)

ddd_110 = analysis(gal_110, gal_110_off, f, v, 1420.4, 110)

ddd_115 = np.zeros(256)

ddd_120 = analysis(gal_120, gal_120_off, f, v, 1420.4, 120)

ddd_125 = np.zeros(256)

ddd_130 = analysis(gal_130, gal_130_off, f, v, 1420.4, 130)

ddd_135 = np.zeros(256)

ddd_140 = analysis(gal_140, gal_140_off, f, v, 1420.4, 140)

ddd_145 = np.zeros(256)

ddd_150 = analysis(gal_150, gal_150_off, f, v, 1420.4, 150)

ddd_155 = np.zeros(256)

ddd_160 = analysis(gal_160, gal_160_off, f, v, 1420.4, 160)

ddd_165 = np.zeros(256)

ddd_170 = analysis(gal_170, gal_170_off, f, v, 1420.4, 170)

ddd_175 = np.zeros(256)

ddd_180 = np.zeros(256)
```

8.4 Python Code for Plotting the Radial Velocity vs. Longitude Flux Map (Figure 5)

Using the data returned above for each longitude interval, it is possible to create the map seen in Figure 5.

```
#Defining a combined data set to plot radial velocity vs longitude

dddd = np.array([ddd_00, ddd_05, ddd_10, ddd_15, ddd_20, ddd_25, ddd_30, ddd_35, ddd_40, ddd_45, ddd_50, ddd_55, ddd_60, ddd_65,

                 ddd_70, ddd_75, ddd_80, ddd_85, ddd_90, ddd_95, ddd_100, ddd_105, ddd_110, ddd_115, ddd_120, ddd_125, ddd_130,

                 ddd_135, ddd_140, ddd_145, ddd_150, ddd_155, ddd_160, ddd_165, ddd_170, ddd_175, ddd_180])
```

```
#Smoothing out the data for a nicer plot

dddd_smooth = gaussian_filter(dddd.T, 2)
```

```
#Plotting radial velocity vs galactic longitude waterfall

plt.figure(figsize=(10,8))

plt.imshow(dddd.T, aspect='auto', extent=[0, 180, np.amin(v), np.amax(v)], cmap='gist_gray', vmin=0, vmax=0.05)

plt.xlabel('Galactic Longitude (deg)', fontsize=10)

plt.ylabel('Radial Velocity (km/s)', fontsize=10)

plt.title('Radial Velocity vs. Longitude Flux Waterfall', fontsize=15)

plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)

plt.show()
```

```
#Plotting it again but with the smoothed data

plt.figure(figsize=(10,8))

plt.imshow(dddd_smooth, aspect='auto', extent=[0, 180, np.amin(v), np.amax(v)], cmap='gist_gray', vmin=0, vmax=0.05)
plt.xlabel('Galactic Longitude (deg)', fontsize=10)
plt.ylabel('Radial Velocity (km/s)', fontsize=10)
plt.title('Radial Velocity vs. Longitude Smoothed Flux Waterfall', fontsize=15)
plt.colorbar().set_label('Power (dB Bits)', rotation=90, fontsize=10)
plt.show()
```

8.5 Python Code for Plotting the Rotation Curve and Mass Enclosed (Figure 6-7)

```
#Defining radius from the center of the galaxy based on longitude
longitudes = np.arange(10, 85, 5)
radii = np.abs(8*np.sin(longitudes*(np.pi/180)))

#Defining the rotational velocity based on longitude and radial velocity of the primary peak
peak1_radial = np.array([-30, -24, -7, 0, 3, 8, 0, -2, -10, -23, -27, -33, -36, -38, -29])
peak1_rotational = np.abs(peak1_radial + 220*np.sin(longitudes*(np.pi/180)))

#Defining the rotational velocity based on longitude and radial velocity of the secondary peak
peak2_radial = np.array([-90, -93, -90, -100, -108, -107, -100, -95, -84])
peak2_rotational = np.abs(peak2_radial + 220*np.sin(longitudes[6:]*(np.pi/180)))

#Defining the rotational velocity based on longitude and radial velocity of the tertiary peak
peak3_radial = np.array([83, 84, 76, 70])
peak3_rotational = np.abs(peak3_radial + 220*np.sin(longitudes[8:12]*(np.pi/180)))

#Plotting the primary peak rotational velocity as a function of radius
```

```
plt.figure(figsize=(10,8))

plt.plot(radii, peak1_rotational, '.', markersize=10)

plt.errorbar(radii, peak1_rotational, xerr=radii_error, yerr= peak1_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)

plt.title('Primary Peak Rotational Velocity Curve', fontsize=15)

plt.xlabel('Radius (kpc)', fontsize=10)

plt.ylabel('Rotational Velocity (km/s)', fontsize=10)

plt.show()


#Plotting the secondary peak rotational velocity as a function of radius

plt.figure(figsize=(10,8))

plt.plot(radii[6:], peak2_rotational, '.', markersize=10)

plt.errorbar(radii[6:], peak2_rotational, xerr=radii_error[6:], yerr= peak2_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)

plt.title('Secondary Peak Rotational Velocity Curve', fontsize=15)

plt.xlabel('Radius (kpc)', fontsize=10)

plt.ylabel('Rotational Velocity (km/s)', fontsize=10)

plt.show()


#Plotting the tertiary peak rotational velocity as a function of radius

plt.figure(figsize=(10,8))

plt.plot(radii[8:12], peak3_rotational, '.', markersize=10)

plt.errorbar(radii[8:12], peak3_rotational, xerr=radii_error[8:12], yerr= peak3_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)

plt.title('Tertiary Peak Rotational Velocity Curve', fontsize=15)
plt.xlabel('Radius (kpc)', fontsize=10)
plt.ylabel('Rotational Velocity (km/s)', fontsize=10)
plt.show()


#Plotting all three peaks together
plt.figure(figsize=(10,8))
plt.plot(radii, peak1_rotational, '.', markersize=10, label='Primary Peak')
plt.plot(radii[6:], peak2_rotational, '.', markersize=10, label='Secondary Peak')
plt.plot(radii[8:12], peak3_rotational, '.', markersize=10, label='Tertiary Peak')
plt.errorbar(radii, peak1_rotational, xerr=radii_error, yerr= peak1_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)
plt.errorbar(radii[6:], peak2_rotational, xerr=radii_error[6:], yerr= peak2_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)
plt.errorbar(radii[8:12], peak3_rotational, xerr=radii_error[8:12], yerr= peak3_rotational_error, linestyle='none', ecolor='g', label='Error', capsize=2)
plt.title('Tertiary Peak Rotational Velocity Curve', fontsize=15)
plt.xlabel('Radius (kpc)', fontsize=10)
plt.ylabel('Rotational Velocity (km/s)', fontsize=10)
plt.legend()
plt.show()


#Mass enclosed as a function of radius in both kg and solar mass units
mass_enc_kg = ((peak1_rotational*1000)**2)*(radii*(3.086*(10**(19))))/(6.67*(10**(-11)))
mass_enc_sm = mass_enc_kg/(1.99*10**(30))
```

```
#Plotting mass enclosed as a function of radius

plt.figure(figsize=(10,8))

plt.plot(radii, mass_enc_sm, '.', markersize=10)

plt.errorbar(radii, mass_enc_sm, xerr=radii_error, yerr=m_enc_sm_error, linestyle='none', ecolor='g', label='Error', capsize=2)

plt.title('Mass Enclosed vs. Galactic Radius', fontsize=15)

plt.xlabel('Radius (kpc)', fontsize=10)

plt.ylabel('Mass Enclosed (Solar Mass)', fontsize=10)

plt.show()
```