Calculating the Hubble Constant

James Keys[1] and **Bowen Zhang**[1]

[1]*Department of Physics,*
17 February 2021

In this study, a value for the Hubble Constant was inferred from simulated data. This data mocks that of what comes from interferometers and also what physicists record from the sky. Using python, this data was processed to find a recession velocity ($v = (471.2595 \pm 2.5715) * 10^4$m/s) and average proper Distance ($D = (13.7 \pm 1.6) * 10^{-23} m$. Which led to a value of $H_o = 106 \pm 2$ km s^{-1}. The found values are roughly within the same magnitude as previously derived Hubble constants [1] leading to an experimental error of 51.4% when compared.

INTRODUCTION

Henri Poincaré first proposed gravitational waves in 1905 and would later become on of the basis for Albert Einstein's general theory of relativity (1916). They would not be directly observed until 2015 by the LIGO gravitational wave detectors. These detectors require an immense amount of effort to design and develop. They detect gravitational waves form immense events in the universe. Immense events like two black holes colliding. This lab will infer the Hubble constant from gravitational wave signals (simulated data).

$$H_o = v/D \tag{1}$$

Where v is recession velocity and D is distance, from our galaxy to another.

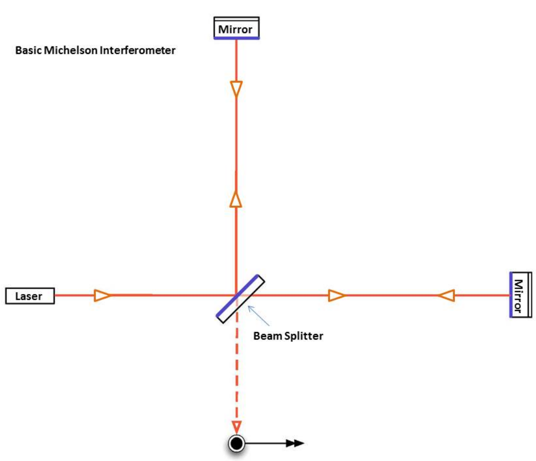

FIG. 1: *The apparatus consisting of a light tight tube.*

METHODS

Treating this simulated data as real, the data would have come from something like LIGO's Interferometer (figure 1). In LIGO, the mirrors are placed 4km apart to measure oscillations with amplitudes around $10^{-18}m$These amplitudes are normalized by the length the detectors arm to find the strain, h and the time is also recorded.

These detectors will measure a immense cosmological event (ie. two black holes colliding) and record h and t. With this it is possible to calculate the proper distance. Physicists must also record the wavelength, λ, and flux of the event to evaluate a recession velocity from the red-shift. The latter of which being less complex to evaluate.

ANALYSIS/RESULTS

The first task was to find the red shift. Using the sodium D absorption line, $\lambda = 5896$ Angstrom, and finding the observed wavelength for it from the plotted spectrum. As seen in figure 2, the wavelength for sodium D is shifted and the observed wavelength is $\lambda' = 5988.68 \pm 1.54$ Angstrom This value was extracted by code to find the minimum value of the flux in the region and wavelength at which it occurs.

The recession velocity is then found from the red-shift:

$$v \approx c * z = c * \frac{\lambda' - \lambda}{\lambda} \tag{2}$$

$$v = (471.2595 \pm 2.5715) * 10^4 \text{m/s} \tag{3}$$

Finding the Distance

The simulated data from an interferometer gave 232609 points of time. t in seconds and the nor-

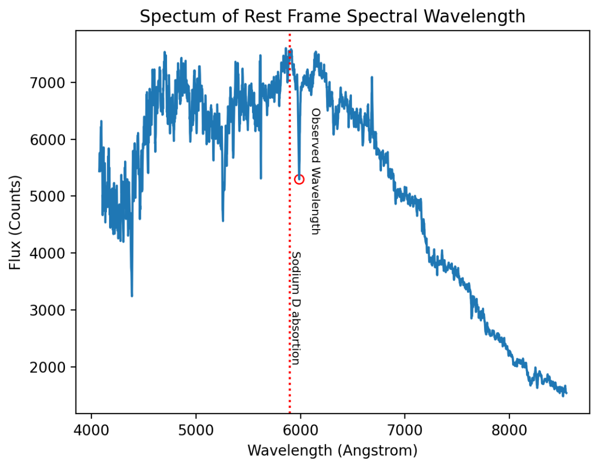

FIG. 2: *Example of the type of interferometer used.*

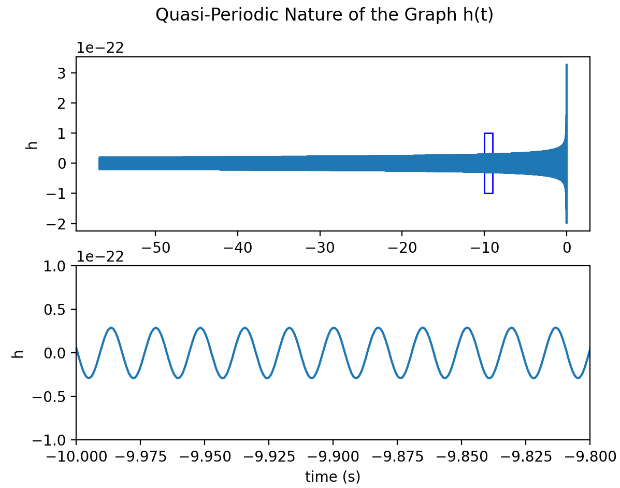

FIG. 3: *Quasi periodic nature of data. The small box is the second plot.*

malized separation distance, h. Using the known formula for proper distance:

$$D = \frac{4c}{|h|}\frac{5}{96\pi^2}\frac{\dot{f}_{\text{gw}}}{f_{\text{gw}}^3} \tag{4}$$

Where f_{gw}^3 is the frequency from the gravitational wave data and $\dot{f}_{\text{gw}}$ is the derivative with respect to time. To get $f(t)_{\text{gw}}$ we would have to represent the data from $h(t)$ in the domain of frequency instead of time. To accomplish this, the Fourier transform of $h(t)$ is needed. Due to the quasi-periodic nature of the data (figure 3) a more local approach is used.

For all t, the frequency around a few oscillations was found around that t. This frequency came from a fast Fourier transform of neighborhood of the few oscillations. Then for all the FFT'ed values, the frequency was found by finding the at what value the max FFT'ed value occurred at. All FFT calculations where preformed by the scipy.fftpack. Initially the generic fft and fftfreq where these stand for discrete Fourier transforms and the discrete Fourier Transform sample frequencies. This initially prodeced results with two frequecies per t. This was due to the real numarical nature of the data, so rfft and rfftfreq where used instead. Which will compute the 1-D discrete Fourier Transform for real input and also fit sample frequencies.

For fast Fourier transformation, let H be the discrete Fourier transform of h

$$H_k = \sum_{n=0}^{N-1} h_n * e^{-i2\pi fTn} \tag{5}$$

This is the applied formula that is used in scipy's fast Fourier transformation algorithm. From this an error propagation can be found as:

$$\text{Var}(H_k) = \sum_{n=0}^{n=N-1} \sigma_n^2 \tag{6}$$

The variance of a discrete FT can be seen to be the total sum of the square of the variances for the measurements of the strain, h_n. Using this logic for the frequency, f_k for the FFT'ed vals $H(f)_k$

$$\text{Var}(f_k) = \sum_{n=0}^{n=N-1} \sigma_n^2 \quad \text{, for } t_n \tag{7}$$

Going back to the data, the quasi-periodic nature of the data leads to some re-occurring frequencies that where calculated. This was due to the sample size of the real FFT. From an sample rate of 4096 samples/s and using the neighborhood of time to be .07 seconds (roughly 4 oscillations). A sample size of $N = 287$ was used. After tossing the repeating frequencies and finding the t at which these frequencies occurred the total set of points dropped from 232609 to 26 points. Each one of these points represents a spot of a roughly unique frequency.

Using equation (6) gave a $\text{Var}(f_k) = 0.004$, while using a recording deviation of about $2.5e-4$ for each time recording. For reference the value of the highest value frequency was $f_{26} = (386.6853 \pm .0004)s^{-1}$. Using the 26 points and roughly approximating the derivative as so:

$$\left.\frac{df}{dt}\right|_i \approx \frac{\Delta f_i}{\Delta t_i} = \frac{f_{i+1} - f_i}{t_{i+1} - t_i} \tag{8}$$

Where i in this case is the data point in the range of 26. Looking at figure 4 below, we can assume that this is a reasonable approximating for the derivative based on number of points. As $t \to 0$ this becomes a more accurate approx since there involves more frequency change as the event nears is climax.

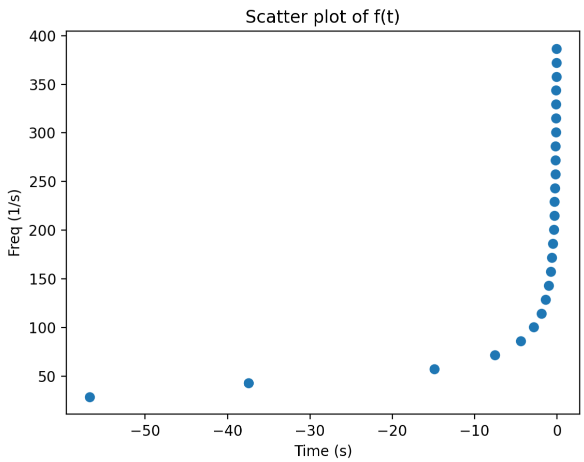

FIG. 4: *Scatter of the frequency and time.*

So far the values for f_{gw} and $\dot{f}_{gw}$ have been found for certain 26 times. Now the local amplitude $|h|$ is left. This was simply found by looking in the neighborhood of a certain t_i. Then calculating the max and min of h and subtracting the values for each t_i.

Since $\dot{f}_{gw}$ requires the point next in the itteration. This another value loss with the result of 25 different calculated distances. After averaging and calculating the standard deviation of the mean the final result can be seen in table below along with the calculations for the Hubble constant. The significant figures of the distance ended up being determined by the significant figures of the strain.

Recession Velocity	v $= (471.2595 \pm 2.5715) * 10^4$m/s
Distance	D $= (13.7 \pm 1.6) * 10^{-23} m$
Hubble Constant	H$_o = 106 \pm 2$ km s^{-1}Mpc^{-1}

DISCUSSION

The approach of using local FFT's to find values for frequency serves as a practical way of evaluating the Hubble Constant from mock interferometer data. Comparing the final value of this experiment's derived Hubble Constant to an actual from Physics today [1].

$$(H_o)_{\text{actual}} = 70 \text{ km s}^{-1}\text{Mpc}^{-1} \tag{9}$$

This lead to a percent error of 51.4%. Most of this error can be explained by the systematic errors involved in the FFT technique applied.

One particular way to increase the accuracy, would be to increase the sample size of the Fourier transforms in the code. The problem with this is that this directly increases the computation time of the code. The error propagation of the frequency from this method is very low. More frequencies needed to be calculated from the times and strain.

One method of making a spectrogram was also tried for this experiment. A spectrogram of the data was printed and a line was drawn on it for $f(t)$. This led to finding an actual function for $f(t)$ instead of points that looks remarkably similar to figure 4. The approach led to even higher error so the direct discrete FT method was done. However, taking the graph from figure 4 and fitting a line on it could also make for a better value of H_o. This way from the fit, $\dot{f}_{gw}$. This would be far more accurate and may lead to a lower percent error.

All systematic error is attributed to the calculation of D. As the recession velocity is simple to calculate and the error involved in v comes directly from the values given. The red-shift approximation is the only method known of finding v.

BIBLIOGRAPHY

[1] Daniel E. Holz, Scott A. Hughes, and Bernard F. Schutz."Measuring Cosmic Distances with Standard Sirens". Physics Today (2018) 71, 12, 34