Draft version July 6, 2022
First Interim Report for Caltech LIGO SURF 2022

GW strain calibration of LIGO detectors with high precision and low latency

Naomi Shechter,[1]
Mentors: Ethan Payne,[2] and Dr. Alan Weinstein[2]

[1] *Department of Physics and Astrophysics, DePaul University*
[2] *Caltech LIGO*

I. PRINCIPLES OF OPERATION AND CALIBRATION FOR THE LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY

1.1 - Operation of Laser Interferometer Gravitational Wave Observatory

The Laser Interferometer Gravitational Wave Observatory (LIGO) at Hanford and Livingston is among the most sensitive detectors ever built. This makes it all the more remarkable that LIGO functions similarly to the archetypal Michelson Interferometer. Due to this high sensitivity, the detector must be well calibrated to produce meaningful, unbiased results. However, calibration process cannot be adequately described without first explaining how LIGO operates. Laser light enters the detector and encounters the beam splitter, which sends it into two perpendicular, 4 km long arms (Fig.1). Each arm includes a Fabry-Perot cavity. The first mirror in the cavity is 99% reflective, so only 1% of the incident light enters the cavity. It builds in power significantly due to the nearly 100% reflective test mass at the other end of the arm. However, rather than produce a pattern of constructive interference after the split laser beams of the interferometer recombine, the light reflected by the arm cavities largely cancels the fresh light emitted by the laser. In other words, LIGO is an active null instrument that relies on destructive interference to maintain a baseline output from which deviations may be measured. When a GW passes, a transverse strain $\Delta L/L$ is induced in arms, which causes a minuscule differential arm (DARM) length change on the order of 10^{-20} meters. This causes a small phase shift in the light recombining at the beamsplitter, and thus a small amount of light exits the beamsplitter in the direction of the readout port.

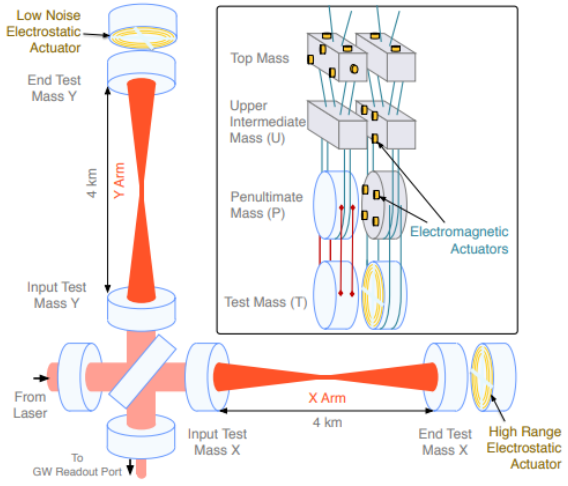

Figure 1. A simplified figure of the LIGO detector (1). Laser light enters from the left and encounters the beamsplitter. The test masses (inset) are constantly actuated to keep the detector on resonance and nearly complete destructive interference. If a gravitational wave passes, the strain will cause a transverse change in arm length. The differential arm length is detected from the readout port as a change in intensity.

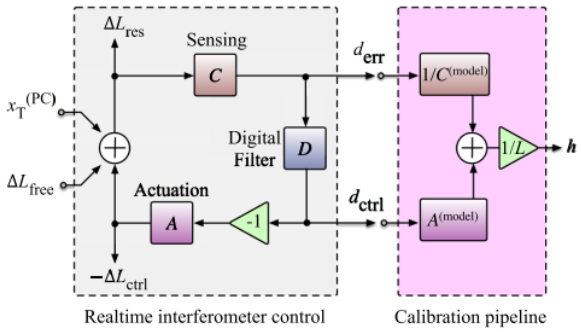

Figure 2. The LIGO control loop and the LIGO calibration pipeline (1). The measures the change in length as the error signal d_{err} which the digital filters convert into d_{ctrl}. The control signal is then used to actuate the test masses by the control length ΔL_{ctrl}. This ensures that the mirrors remain in their nominal positions. The calibration pipeline passes these same signals through filters, which are fit by MCMC and GPR methods that compare the measurements to a model. The output of the calibration pipeline is the reconstructed strain and the uncertainty estimate.

1.2 - DARM Control Pipeline and Calibration Pipeline

A feedback control loop constantly attempts to maintain a null position of the interferometer. ΔL_{free}, the actual change of length in the arms, is measured in the presence of the feedback control loop which actively suppresses it through actuation. The DARM length ΔL_{res} is passed to the sensing function, which produces the error signal d_{err}. This is the input for the DARM control pipeline's digital filters and actuation function, which help maintain the nominal positions of the test masses and resonance in the arm cavities (Fig.2). The d_{err} signal is also fed into the calibration pipeline, which uses a detailed model of the sensing and actuation functions to reconstruct the gravitational wave strain by comparing the model to the measured response over different time epochs (2). From the resultant response function and the outputs of various statistical algorithms, one may characterize the complete detector response along with the calibrated strain time series $h(t)$. The strain is comprised of Gravitational Wave data, displacement noise and sensing noise. To more fully understand the quality of $h(t)$, we also produce the strain uncertainty estimate, which is dependent solely on Gravitational Wave information.

1.3 - The Importance of Precision Low Latency Calibration and Uncertainty Estimates

Calibration is a crucial process to ensure that the aforementioned pipeline provides accurate data, allowing us to more fully understand GW sources. Given how sensitive LIGO must be to measure such a miniscule strain, it must also be incredibly accurate. Currently, a code called pyDARM is being utilized to model the DARM control loop and strain uncertainty estimation in a more accessible manner than has been done in O3 (2). Testing and supplementing pyDARM for the purpose of developing a reliable low latency uncertainty estimate for O4 forms the basis of my research.

As of the third observing run (O3) which lasted from April 2019 to March 2020, our calibration uncertainty within the sensitive band of 20-2000 Hz was known to $\sim 2\%$ in magnitude and $\sim 2^\circ$ in phase (1). There are several reasons why we continue to seek higher precision, although all are motivated by the prevention of biased astrophysical results (3, 4, 5, 6, 7, 8). Concerning tests of general relativity in the strong regime, only evidence collected with extremely reliable calibration would be capable of shedding doubt on the theory. For future observation runs such as O5, we expect an increase in sensitivity so significant that uncertainty must be lowered to 1% levels in order to reliably reconstruct signals of SNR greater than 100. Additionally, calibration uncertainty directly affects the parameter estimation uncertainty for both GW-based and multi-messenger astronomy(1). To summarize, many data analysis pipelines and research projects rely on high quality detector calibration. This makes it not only a crucial piece of LIGO's intricate construction, but one which must be well-understood, precise and efficient to produce reliable data.

Understandably, sub-optimal calibration introduces many issues; verifying calibration during O3 was a particularly time-intensive process (3). The $\sim 2\%$ level of uncertainty was only released after months of verifying the initial uncertainty estimate which was then estimated to be of order 5%. Furthermore, by the time the calibration model C01 was completed, much of the parameter estimation had already occurred with the less precise C00 model (which actually lacked a reliable uncertainty estimate). In preparation for O4, the calibration group is working to expedite the process to verify the low latency strain uncertainty estimate. My research project will contribute to the realization of this goal.

II. APPROACH AND OUTLINE FOR THE RESEARCH PROCESS

In order to expedite the process of generating calibrated strain uncertainty with high precision and low latency, I will be writing a monitoring program and producing a suite of diagnostic tools. The software will characterize the uncertainty output by pyDARM. Should it fall outside of satisfactory limits, the program will raise a specific warning as to what source of error must be further examined, along with plots of the calibration lines, time dependent correction factors (TDCFs) and other values which contribute to the overall uncertainty budget (1). The software must operate reliably, without crashing, produce these diagnostic tools with a latency of less than an hour. Once the monitoring software is able to correctly diagnose the issues found in O3, it will be automated with documentation in order to make it accessible to anyone seeking to understand the state of LIGO's calibration.

A thorough understanding of pyDARM will be necessary if I seek to accurately characterize the uncertainty it produces. Therefore, I have spent the first week or so stepping through a Jupyter notebook by Evan Goetz and colleagues, which contains various demonstrations on sensing and actuation function models, as well as Markov Chain Monte Carlo (MCMC) and Gaussian Process Regression (GPR) algorithms. My understanding of the algorithms has been supplemented with several papers (9, 10). In the process of producing plots and encountering errors, I have

become more familiar with the various pyDARM classes. I have also done some introductory learning about control systems and null instruments to more fully understand the LIGO control pipeline (11).

Currently, I am in the process of compiling several pyDARM-compatible model files for the epochs in O3 which have yet to be investigated with pyDARM. Once that is completed, I will run these model files through pyDARM to validate the process and the established model configurations. An estimate of the strain uncertainty will be produced for each model, as well as plots of the sensing and actuation functions, digital filters, MCMC residuals and uncertainty from the GPR. Once this is complete, I will begin to work on automating strain uncertainty generation and move onto the next phase of my project. Due to the emergent nature of this project, the exact manner in which automation will occur and the monitoring software will be constructed is as of yet unknown. However, it will likely include sampling the results of the GPR once an hour, and saving only as much data as is needed to produce a reliable model of the overall response. From this information, I will be producing graphs which keep track of important TDCFs. These include quantities such as $\kappa_c(t)$, $\kappa_{tst}(t)$ and $f_{cc}(t)$ which are the scalar gain factor of the sensing function and the test mass actuation function, and the coupled cavity pole frequency respectively.

III. PROGRESS IN ACQUIRING BACKGROUND INFORMATION AND PROJECT MILESTONES

Most of my first week and a half was spent reading papers, exploring the pyDARM code and building the background knowledge necessary to carry out my research. In this section, I will outline my understanding and share plots which are illustrative of the various subjects I have been introduced to thus far.

3.1 - Linear Time Invariant Systems, Control Systems and Null Instruments

Before interacting with pyDARM, I sought out resources about linear time-invariant (LTI) systems, control systems and null instruments (11). Consider a mass on a spring, to which you provide an impulse by hitting it suddenly. The function for driven dampened simple harmonic oscillators defines the impulse response, and the result would be a decaying sinusoid as the oscillating mass eventually comes to rest. If you model a more complicated input signal as a sum of impulses, the output of an LTI system will be the linear superposition of the impulse response from each individual impulse. In mathematical terms, the output is the convolution of the impulse response with the input. For LTI systems, convolution becomes multiplication after performing a Fourier transform, so it is simpler to work in frequency domain. Therefore, rather than the impulse response determining the output of the system, we use the Fourier transform of the impulse response: the transfer function. In the context of LIGO, rather an an impulse, the excitation signal is a swept sine at a single frequency, which is varied across the sensitive band of 20-2000 Hz.

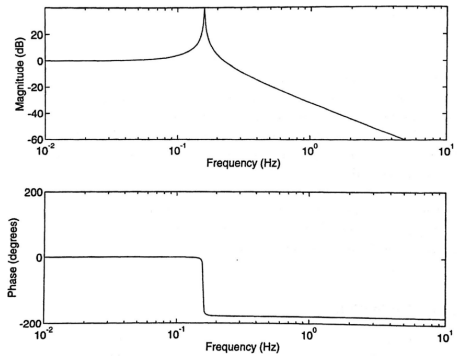

Figure 3. Bode plot of a mechanical oscillator from Fundamentals of Interferometric Gravitational Wave Detectors by Peter R. Saulson. The large peak indicates the resonant frequency, at which the phase passes through 90° (11).

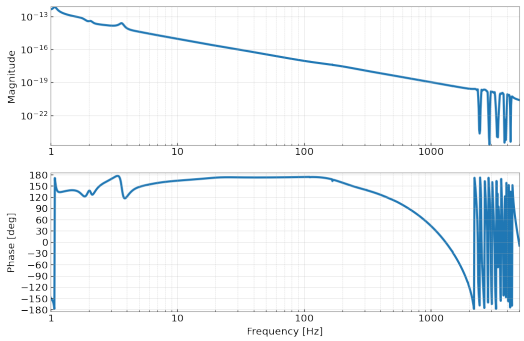

Figure 4. The complete actuation transfer function for the LIGO Hanford calibration model produced on April 16th 2019. The decreasing slope of the upper magnitude graph is similar to the decreasing slope which occurs in Figure 3 at frequencies higher than resonance.

The transfer function of a driven damped simple harmonic oscillator, such as a pendulum, is plotted over a range of frequencies in Figure 3. The large spike in the magnitude of oscillation is resonance, at which the driving frequency equals the natural frequency of the oscillator. As the two frequencies become completely out of phase, the magnitude becomes negligible. This means that there is a pole located at resonance, or that (should we act this transfer function

upon a signal), frequencies around resonance would be amplified. There are also zeroes, at which the signal goes to zero and nearby frequencies are attenuated. In other words, poles and zeroes are a very useful quality of transfer functions, and allow us to construct filters. For example, actuation transfer function is modeled in such a way to reduce noise from the movement of the test masses (Fig.4). The upper intermediate mass and penultimate mass each contribute to the overall transfer function, and much like in Figure 3, they attenuate high frequencies. Effectively, the entire actuation transfer function works as a digital low pass filter.

3.2 - Markov Chain Monte Carlo and Gaussian Process Regression

After I produced several plots of the the transfer functions in pyDARM, I moved on to understanding the ways in which we compare and fit the modeled detector response with measured data. The first of these which I encountered was the Markov Chain Monte Carlo (MCMC) method (9). Given a prior distribution and a likelihood function of our model values and measured data, MCMC generates a distribution for the model parameters which are sampled for the maximum a posteriori (MAP) values. These values are of maximum posterior probability between the parameters, and they are used to construct the best fit curve. However, the case of the LIGO Hanford model produced for April 17th 2021 (Fig.7), it is clear that the MCMC alone does not adequately fit the data. The remaining errors are handled by Gaussian Process Regression (GPR), which produces a probability distribution of many possible functions which may fit the data (9). The variance from GPR more fully characterizes the uncertainty between the model and measurements than that achieved with MCMC.

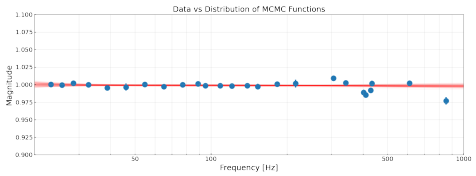

Figure 5. A graph of 100 possible fits to the sensing function's model and measurement residuals. Produced by the MCMC for April 16th of 2019, in the sensitive range from 20-2000 Hz. While the uncertainty in each individual data point is small, the overall uncertainty from a possible best fit curve is significant, especially surrounding 500 Hz. The MCMC alone does not adequately fit the data or quantify the uncertainty. This is why we must employ Gaussian Process Regression.

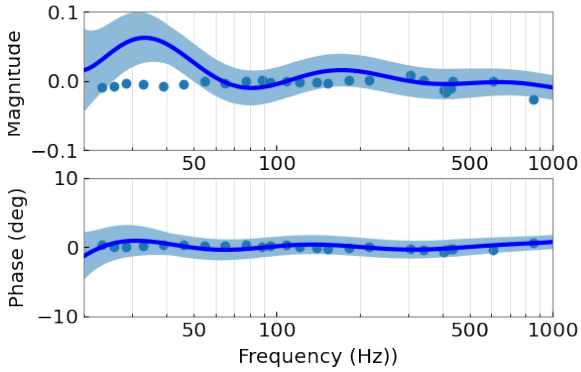

Figure 6. The corresponding GPR graph for the sensing function's model and measurement residuals, in the sensitive region from 20-2000 Hz. The envelope represents the standard deviation away from the blue line, which is the best fit. The uncertainty is more reliably characterized across the majority of the frequency band, although issues remain below 50 Hz.

As of the beginning of July, this is the progress I have made, along with transferring over the necessary model files from O3 into pyDARM compatible files. Ideally, by end of the fourth week I will have gathered the necessary data to begin work on the monitoring software.

IV. CHALLENGES AND LEARNING EXPERIENCES

Most of the issues I have encountered so far have been about plotting or how certain pyDARM functions store data, and all have been learning experiences about paying attention to detail. For example, after running the MCMC, you reset the parameters of the sensing function to the MAP values. However, this didn't take into account that one parameter (the detuned spring quality factor) should actually be inverse based on the equations for the sensing transfer function. This was something which had to be fixed in order for the MCMC graphs to be accurate. An issue was also encountered with the GPR plots. The previously written code no longer seemed to work, and could have used a more efficient pyplot function to shade in the envelope. I produced a solution which was initially only valid on the left side of the graph and did not properly account for the increased uncertainty at higher frequencies. However, after

discussing with my mentor Ethan, I was able to solve this issue by changing the frequencies over which I was fitting the data in both the MCMC and GPR files.

The challenges I am anticipating involve the learning curve which is sure to become apparent as I begin writing the monitoring software. In particular, I am unfamiliar with automating operations such as reading in files, or with prioritizing the running time of a code. Given that the software must be reliable and robust, these are challenges which I will have to face. I can also expect to learn how to more fully interpret the code and data with which I have been making plots. For example, given a plot such as Figure 8, how does one determine when enough points are sufficiently uncertain to raise a warning? Does 'uncertainty' in this case mean the error bars fall outside the best fit, or must a point fall outside the envelope entirely in order to raise a flag? By learning the answers to questions such as these through experience, I believe I will be challenged in a meaningful, educational way.

IV. ACKNOWLEDGMENT

This report was funded by the National Science Foundation (NSF) and completed as part of the LIGO SURF Program at Caltech, in contribution to the LIGO Scientific Collaboration. Many thanks to my mentors Dr. Alan Weinstein and Ethan Payne.

REFERENCES

[1]Sun, L., Goetz, E., Kissel, J. S., et al. "Characterization of systematic error in Advanced LIGO calibration." Classical and Quantum Gravity, 37, 225008. doi:10.1088/1361-6382/abb14e (2020).

[2]PyDARM Gitlab Repo of the Laser Gravitational Wave Interferometer Calibration Group.

[3]Conversations with Mentors concerning background knowledge and the calibration process, during the first week of the LIGO SURF Program.

[4]Payne, E., Talbot, C., Lasky, P. D., et al. "Gravitational-wave astronomy with a physical calibration model." PhRvD, 102, 122004. doi:10.1103/PhysRevD.102.122004 (2020).

[5]Vitale, S., Del Pozzo, W., Li, T. G. F., et al. "Effect of calibration errors on Bayesian parameter estimation for gravitational wave signals from inspiral binary systems in the advanced detectors era." PhRvD, 85, 064034. doi:10.1103/PhysRevD.85.064034 (2012).

[6]Vitale, S., Haster, C.-J., Sun, L., et al. "physiCal: A physical approach to the marginalization of LIGO calibration uncertainties." PhRvD, 103, 063016. doi:10.1103/PhysRevD.103.063016 (2021).

[7]Essick, R. & Holz, D. E. "Calibrating gravitational-wave detectors with GW170817." Classical and Quantum Gravity, 36, 125002. doi:10.1088/1361-6382/ab2142 (2019).

[8]Huang Y., Chen H.-Y., Haster C.-J., Sun L., Vitale S., Kissel J. "Impact of calibration uncertainties on Hubble constant measurements from gravitational-wave sources." arXiv:2204.03614 (2022).

[9]Cahillane C. et al. "Calibration uncertainty for Advanced LIGO's first and second observing runs." Physical Review D 96.10 102001. (2017).

[10]Wang, J. "An intuitive tutorial to Gaussian processes regression." arXiv:2009.10862 (2020).

[11]Saulson, Peter R. "Fundamentals of Interferometric Gravitational Wave Detectors". Singapore: World Scientific (2017).