LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

| **Technical Note** | **LIGO-T2000291–vX** | 2020/06/30 |
|---|---|---|

# LIGO SURF Proposal

Ella Buehner Gattis

**California Institute of Technology**
**LIGO Project, MS 18-34**
**Pasadena, CA 91125**
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

**Massachusetts Institute of Technology**
**LIGO Project, Room NW22-295**
**Cambridge, MA 02139**
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

**LIGO Hanford Observatory**
**Route 10, Mile Marker 2**
**Richland, WA 99352**
Phone (509) 372-8106
Fax (509) 372-8137
E-mail: info@ligo.caltech.edu

**LIGO Livingston Observatory**
**19100 LIGO Lane**
**Livingston, LA 70754**
Phone (225) 686-3100
Fax (225) 686-7189
E-mail: info@ligo.caltech.edu

http://www.ligo.caltech.edu/

# 1 Introduction

The Laser Interferometry Gravitational Wave Observatory (LIGO) aims to detect gravitational waves predicted by Einstein's theory of relativity in 1916, which predicted that gravitational waves would cause ripples in space-time, which can be measured as contractions and expansions in space [1]. LIGO consists of two L-shaped detectors, with arms 4km in length. The basic concept they employ in order to detect changes in length caused by ripples from gravitational waves is the same as that for a Michelson interferometer. Each arm of the detector is an exact integer number of wavelengths long, so when monochromatic light travels down each detector arm and is reflected by the mirror at the end, the reflected beam is exactly in antiphase with the initial beam and no light returns to the detector. As such, any microscopic change in length of the LIGO detector arm will cause the transmitted and reflected beam to no longer be in antiphase, and the resulting change in length can then be measured by the detector.

However, seismic movement will also cause movement in the detector arms, introducing noise into the detector readings. We aim to create a heat map of seismic activity in the area surrounding LIGO to better understand and account for seismic activity in detector readings.

Seismometers measure ground movement and are usually placed 2 meters below the surface, where temperature is steady with very few or negligible fluctuations. However, the seismometers around LIGO are above ground in order to measure seismic activity at the surface. Because of this, they are subject to above-ground temperature fluctuations, which affect the seismometer readings and introduce noise into the seismic data. In order to minimise the effect of temperature fluctuations on seismometer readings, the seismometer is placed in a steel enclosure surrounded by a resistive heater and insulating foam.

# 2 Objectives

The aim of this project is to design a feedback system that will keep the temperature within the seismometer enclosure constant, to ensure there is as little noise in the seismic data due to temperature fluctuations as possible. The first step is to design a model of the system, and test the system response to different changes in temperature.

Once an accurate model of the system has been created, it will be possible to test the response of the system with linear control, such as PID (proportional-integral-derivative) control. In linear control, the output signal is proportional to the input signal supplied, which limits the flexibility of control that can be implemented to only be linear.

Later, we will test non-linear feedback systems such as a machine learning mechanism using a neural network for control. In non-linear control the output signal is not necessarily proportional to the input signal. The aim of this is to further reduce noise from temperature fluctuation by minimising overshoot, which is sometimes a problem with linear control methods such as PID, and reduce the time constant.

The efficacy of a linear vs nonlinear temperature control system will then be measured and compared. Our final aim is to design a feedback system that can achieve and maintain temperature stability to within the sensitivity of the temperature sensor being used.

With the seismic data obtained, we aim to plot a "heat map" to visualise the intensity of seismic activity in the areas surrounding the LIGO detectors.

# 3 Approach

We will first model the seismometer in stainless steel system in Python, using the heat equation to model temperature changes as a function of applied power. This modelling will be done in the Laplace domain, and also in the time domain to model non-linearities.

Once the first step of modelling the system has been achieved, it will be possible to also model the effect of PID for temperature control, and to create data with which to train a neural network for temperature control.

## 3.1 Modelling the system

The physical system consists of a seismometer in a stainless steel enclosure wrapped in resistive heating mesh with a resistance of 30 Ohms, covered with an insulating foam. The temperature sensor is an AD590 temperature sensor.

The pulse-width modulation (PWM) circuit used to control the heating system consists of a MOSFET which is used to control current flow to the resistive heater

Using the heat equation, the system can then be modelled by

$$P_{in} = M_{ss}C_{ss}\frac{dT_{ss}}{dt} + P_{loss} \tag{1}$$

where $P_{in}$ is the input power supplied as heat, $M_{ss}$ is the mass of stainless steel, $C_{ss}$ its heat capacity and $T_{ss}$ its temperature at a given time $t$. $P_{loss}$ is the power lost through conduction and radiation.

The power lost by conduction will be

$$P_{\text{conductive loss}} = \frac{k_{foam}A_{foam}}{t_{foam}}(T_{ss} - T_{amb}) \tag{2}$$

where $k_{foam}$ is the thermal conductivity of the insulating foam, $A_{foam}$ is the outward facing surface area of foam, and$t_{foam}$ the thickness. The difference in temperature between the stainless steel enclosure and the ambient temperature of its surroundings is given by $(T_{ss} - T_{amb})$. Then, the power lost by radiation is given according to the Stefan-Boltzmann law by

$$P_{\text{radiative loss}} = \sigma A_{foam}(T_{ss}^4 - T_{amb}^4) \tag{3}$$

We will model this system in Python. To model the non-linearities such as radiative losses and other noise, it will be useful to model the system in the time domain rather than in the Laplace domain.

### 3.2 Linear control- PID

In PID control, there are 3 parameters to be tuned, according to the PID equation:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \tag{4}$$

### 3.3 Nonlinear control using Neural Network

Reinforcement learning vs supervised learning. 5 layers? Set number of neurons in hidden layer and initially assign weights and biases randomly. activation function? loss function? calculates error - compares actual output to ideal "correct" output to adjust weights and biases

#### 3.3.1 Software

We will develop the neural network for temperature control in Python using TensorFlow, an existing open-source machine learning library distributed by Google.

What will be the learning set? Could either use data from the model of the system we build in python, or real experimental data (although this will be harder to obtain remotely), or a combination of the two.

input variables: real temperature, target temperature output variable: voltage supplied to resistive heater

#### 3.3.2 Supervised Learning

Supervised learning uses a given set of labelled data to learn a function that maps the input onto the output without changing the input space. The neural network will then learn to adjust parameters for a given set of input variables to get the desired output. This means we need a labelled set of data for current supplied to restive heater and temperature response of steel enclosure.

#### 3.3.3 Reinforcement Learning

Reinforcement Learning, in contrast to Supervised Learning, doesn't need a labelled data set, and uses trial and error with game-like scenarios operating in discrete time steps find optimum control. RL interacts directly with its environment to create its own data, meaning it can use environments such as the openAI gym for training.

A Reinforcement Learning model takes much longer to train, so in this context Supervised Learning will be more useful.

# 4 Plan

Our first step is to design a model of the system in jupyter notebooks - this is the foundation for all subsequent things, and should take a few weeks, although because I am not starting until later because of my exams being delayed, this may already be completed when I rejoin the project.

Then we must build/design a linear control system (using PID control) also in a jupyter notebook?

fetch real temperature data to measure the system response to changes in ambient temperature

train neural network with training set test efficacy of neural network

anything about the seismic heat map here?

Usually project would run from June 16 for 10 weeks until August 29(?), but because my exam period has been delayed and runs until June 24 can instead run for 10 weeks from then. (June 29 - September 4?)

# References

[1] Albert Einstein. *The Foundation of the General Theory of Relativity*, Annalen der Physik, volume 49, 10.1002/andp.200590044, 1916