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

Technical Note **LIGO-T2100259-v1-** 2021/09/24

System Identification and Optimal Control of Mirror Suspensions

Bhavini Jeloka

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

1.1 LIGO

The Laser Interferometer Gravitational-Wave Observatory, abbreviated as LIGO, has successfully made advancements in the scientific community by detecting gravitational waves and a large number of black holes and neutron star mergers. Owing to the extreme sensitivity of the instrument [1][2], it becomes crucial to ensure that the activity of the suspended mirrors used for the reflection of the laser is controlled, especially in the presence of noise and seismic activity. Obtaining data regarding the system's behaviour will give us information on the current state of the system. By optimizing the Fisher information about our system's parameters (e.g. poles and zeros) under the constraints that there are finite limits on the total excitation energy, the readout noise, and the measurement time, it is possible to improve the sensitivity of the system.

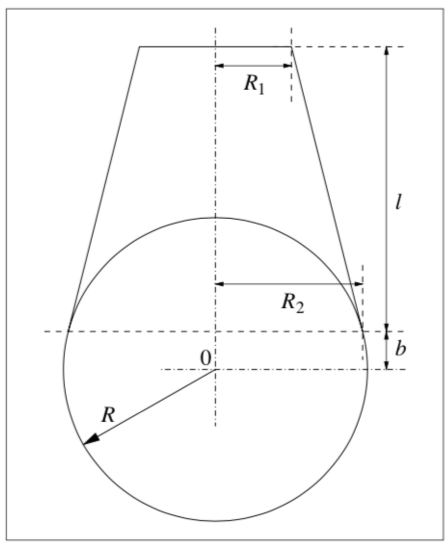

Figure 1: LIGO Single Loop Suspended Mirror [3]

1.2 System Identification

In order to fully understand a system, it is important to formulate a mathematical model corresponding to the dynamics and behaviour of the system. Since reality is complex, it is often difficult to obtain an exact model and we must make suitable simplifying assumptions to obtain a mathematical model. This allows us to quantify the parameters of interest. The resultant mathematical model must be able to closely mimic reality and must be robust to noise and disturbances. Thus identification theory comes into play. System identification deals with concepts relevant to these aforementioned ideas.

Several systems can be modelled as linear and time invariant systems. Under this assumption, the transfer function corresponding to the system can be leveraged to obtain the aforementioned parameters [4]. For instance, in the case of a single suspended mirror system, one can obtain the frequency response, the impulse response and the step response through the Multi-Input Multi-Output (MIMO) transfer function.

As we are working with measurements, we will have to estimate the parameters based on experimental data to obtain useful information. Apart from obtaining the information about the system, we would want to ensure that the noise and errors present in the system are kept to a minimum. This calls for the use of the various estimation techniques that have been developed over the years depending upon the prior information we have about the system. Well known techniques include Least Squares Estimation (no prior information available), Weighted Least Squares Estimation (covariance matrix of the noise is known), Maximum Likelihood Estimator (probability distribution function of the noise is known) to name a few. Since, the number of measurements we have are finite, the accuracy and precision becomes limited. This is formally stated through the Cramer-Rao lower bound that gives a lower limit on the covariance matrix representing the parameters. This bound also dictates the maximum efficiency of the system.

$$CR(\theta_o) = \left(I_{n_\theta} + \frac{\partial b_\theta}{\partial \theta} \right)^T Fi^{-1} \left(I_{n_\theta} + \frac{\partial b_\theta}{\partial \theta} \right) \tag{1}$$

$$Fi(\theta_o) = \mathbb{E}\left\{ \left(\frac{\partial l(z|\theta)}{\partial \theta} \right)^T \left(\frac{\partial l(z|\theta)}{\partial \theta} \right) \right\} = -\mathbb{E}\left\{ \frac{\partial^2 l(z|\theta)}{\partial \theta^2} \right\} \tag{2}$$

Where $b_\theta = \mathbb{E}\{\hat{\theta}\} - \theta_o$ is the bias associated with the estimator. The Fisher information matrix Fi is an important indicator to the information we have about a system. The size of the matrix is directly related to the amount of information we have. Since the covariance matrix inversely depends on Fi, a larger Fisher information matrix (large values, element-wise) would imply a smaller covariance matrix, limited by the Cramer Rao lower bound. For a fixed parameter system, the Fisher information can be maximized by maximizing the likelihood of estimating the observed data. This comes under the umbrella of Optimal System Identification. Another technique that has been used in the past is the local polynomial approximation where the noise transients are discarded so that the data is independent and information is uncorrelated and maximized [5].

1.3 Optimal Linear Control

In order to control the system, one can apply techniques in order to minimize the noise and use optimal control efforts to meet the objective of maximising the information obtained. Two important components to this configuration include the cost function and the feedback loop. The formulation of the cost function assists the optimization of the design variables. On the other hand, through a typical feedback controller, the current state of the system is fed back to the controller and the control minimizes deviation from the reference state. LIGO already employs several feedback as well as feed forward loops for the control of the suspended mirror systems [6].

Once the system has been modelled, we can employ standard optimal control techniques like Pontryagin's maximum principle or the Hamilton-Jacobi-Bell (HJB) equations. Some common controllers that have been used in the past are: Linear Quadratic Regulator, $\mathcal{H}_\infty$ and μ synthesis. $\mathcal{H}_\infty$ is a classical controller and has been used for several decades. It works on the principle of minimising the $\mathcal{H}_\infty$ norm denoted by $||F(j\omega)||_\infty = \sup \sigma F(j\omega)$ where σ is the largest singular value of the transfer function [7]. μ synthesis introduces robustness to the system and is often used alongside $\mathcal{H}_\infty$ controllers. LQR however, operates by minimizing a quadratic cost function for the system.

$$\dot{\mathbf{e}} = A\mathbf{e} + B\mathbf{u} \tag{3}$$

and is often represented as:

$$J = \mathbf{e}^T(t_1)F(t_1)\mathbf{e}(t_1) + \int_{t_o}^{t_1} (\mathbf{e}^T Q\mathbf{e} + \mathbf{u}^T R\mathbf{u})\,dt \tag{4}$$

Here, $\mathbf{e}$ represents the error $\mathbf{u}$ is the control. The matrices Q and R are constant throughout the computation and the entire state space is formulated in the time domain. Hence, we don't get information about the frequency dependence of the system.

2 Objective

Our goal is to model the suspended mirror system as a quadruple pendulum with 24 degrees of freedom (DoF) and optimally obtain the response of the system under the constraints of flat excitation, white readout noise and constant SNR excitation.

3 Approach

To model the system as a quadruple pendulum, the first and foremost step will be that of system identification. Instead of directly jumping to the case with 24 degrees of freedom, we will first start by modelling a simple harmonic oscillator. Then, we will move onto a single suspension, modelled as a pendulum. Once again, this will entail the identification of the system, followed by an estimator to make the system robust to noise.
The linearized dynamics for the single loop system in terms of the position of the centre of mass (x), pitch (θ) and yaw (ϕ) will be directly adopted from [3] and are written here for quick reference:

$$\ddot{x} + \gamma_x \dot{x} + \omega_x^2 x = \omega_x^2 (x_{sp} + b\theta) \tag{5}$$

$$\ddot{\theta} + \gamma_\theta \dot{\theta} + \omega_\theta^2 \theta = \frac{\omega_\theta^2}{l+b}(x - x_{sp}) \tag{6}$$

$$\ddot{\phi} + \gamma_\phi \dot{\phi} + \omega_\phi^2 \phi = \omega_\phi^2 \phi_{sp} \tag{7}$$

Using these equations we can obtain the corresponding state space representation and the frequency response. We can also study the impulse and step responses for the MIMO system.

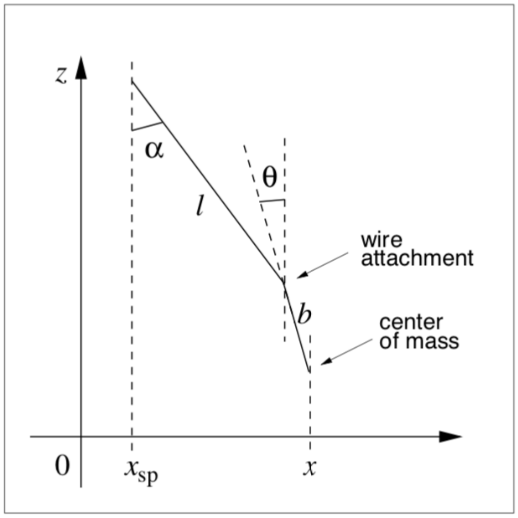

Figure 2: Side View of the LIGO Single Loop Suspended Mirror [3]

As established in Section 1.3, the controllers used previously do not yield results that can tell us about the frequency dependence of the system. Hence, we make use of time varying matrices (Q and R) for the signals . Since we are only interested in certain ranges of frequencies, we integrate over a finite range of weighted signals. The objective of the optimization problem is to maximize the Fisher information and minimize the covariance matrix.

$$J = \int_{\omega_o}^{\omega_1} (\mathbf{e}^T Q(\omega)\mathbf{e} + \mathbf{u}^T R(\omega)\mathbf{u})\, d\omega \tag{8}$$

Before moving onto Eq. (8), we will first implement standard methods for optimal system identification. As mentioned in Section 1.2, we will develop a Least Squares and Weighted Least Squares Estimator. This will allow us to obtain the optimal values for parameters representing our system's transfer function. Once the plant model and parameters (decision variables for our optimization problem) have been decided, we will add noise to the system. In a physical scenario, noise would be attributed to various sources like seismic vibrations and other vibrations. In our simulation model, we assume the nosie to be flat Gaussian noise. The true model is known to us and corresponds Eq.(5)-(7). Once these values are fed into the plant model, we will use optimization to solve for the maximum value of the cost function, which for our case, is the determinant of the fisher information matrix. (The optimizer will result in the optimal plant model. This optimal model will then be tested against the true model and an iterative process will be used until the models match with some preset tolerance).

Once we have established the 6 DoF system, we can extend the approach to the 24 DoF system which is the quadruple pendulum case. This system will once again, follow a similar approach.

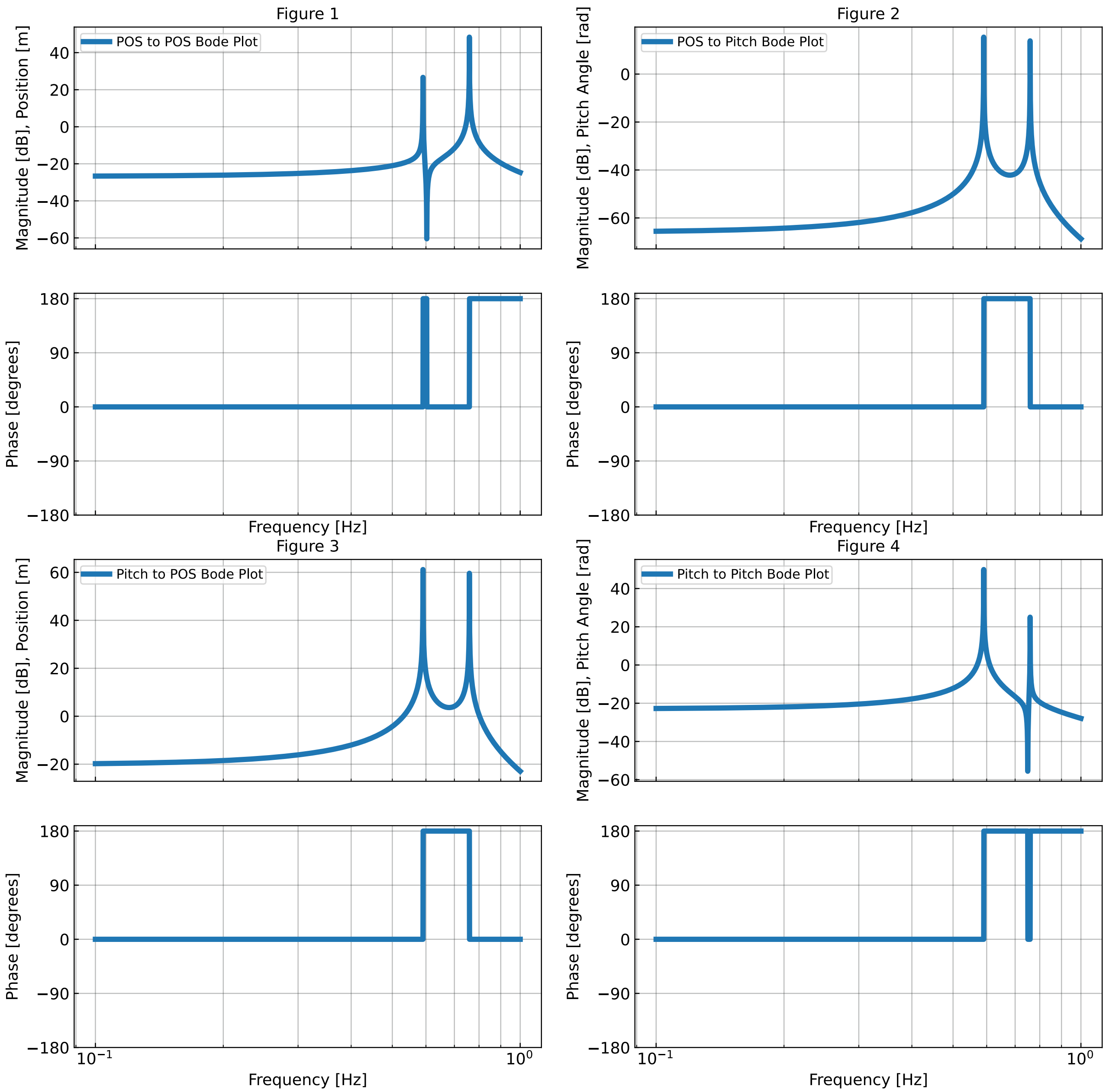

Figure 3: Obtained Bode Plots for the LIGO Single Loop Suspended Mirror

4 Simple Harmonic Oscillator

4.1 System Identification

A simple harmonic oscillator is a standard model used to represent second order mechanical and electrical systems. A spring mass damper system is a good mechanical analogy and a series RLC circuit is the electrical counter part. Since it is a second order system, it is easy to obtain the frequency response and Fisher matrices both analytically and numerically. As a preliminary system identification task, we will use least squares curve fitting to obtain the best experimental approximation to our second order system.

Our entire analysis will be in the frequency domain. We will transform the dynamics of the spring mass damper system from the time domain to the frequency domain and will obtain the corresponding transfer function. This will be our true model. Mathematically, the time domain dynamics are:

$$m\ddot{x} + c\dot{x} + kx = F(t) \tag{9}$$

Where x is the displacement (output) due to the force input. The mass is given by m, the spring constant is given by k and the damping is represented as c. The Laplace transform gives us the following transfer function:

$$G(s) = \frac{X(s)}{F(s)} = \frac{\frac{1}{m}}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{10}$$

The above equation represents a standard second order system where the parameters correspond to $\omega_n^2 = \frac{k}{m}$ and $2\zeta\omega_n = \frac{c}{m}$. ζ is the damping ratio and ω_n is the natural frequency of the system. For a quality factor Q of 10, we get the parameter values as $m = 5kg$, $\omega_n = 2rad/s$ and $\zeta = 0.05$.

We can re-write this equation in terms of the quality factor using $Q = \frac{1}{2\zeta}$. This is because the quality factor is easier to understand intutively and obtain experimentally. The equation becomes:

$$G(s) = \frac{X(s)}{F(s)} = \frac{\frac{1}{m}}{s^2 + \frac{\omega_n}{Q}s + \omega_n^2} \tag{11}$$

As mentioned earlier, Eq. (11) will be used as our true model and we will obtain the displacements corresponding to a force of constant magnitude across all frequencies. The phase will be randomly assigned. Noise will be added to these obtained displacements. The noise will be Gaussian white noise, which means that the magnitude will be constant in the frequency domain the the phase will be random. Thus $\hat{X} = X + n$ where n is the noise. Since the input is noise free (only measurement noise exists), our experimental values of the transfer function would be of the form:

$$\hat{G}(j\omega) = \frac{\hat{X}(j\omega)}{F(j\omega)} \tag{12}$$

We will then use standard least squares curve fitting to obtain the experimental values of Q and ω_n. Throughout our analysis, we will assume that the mass is exactly known. The least sqaures cost function would minimize $|\hat{G}(j\omega)-G(j\omega)|$. The reduced covariance matrix can be directly obtained by taking the inverse of the Hessian (of the objective function) multiplied by the residual sum of squares (RSS). This is further divided by the degree of freedom, i.e., the difference between the number of output (M) and the number of parameters (N). In case of the uncertainties, the square root of the diagonals of the reduced covariance matrix can give the required values.

$$cov = \frac{RSS}{M-N}\mathcal{H}^{-1} \tag{13}$$

$$unc_{ii} = (cov_{ii})^{\frac{1}{2}} \tag{14}$$

4.2 Optimizing the Fisher Matrix

Once we have succeeded with a uniform input model, we will move onto the design of an optimal input spectrum. This technique allows us to redistribute power (or the focre input) such that information received is maximised. This directly translates to maximising the Fisher Matrix and minimising the covariance matrix. The Fisher matrix can be calculated using the negative log likelihood function as given in [8]. We will maximise the determinant to ensure that the Fisher Matrix is maximised. For doing so, we will have to ensure that the least squares fit matches the true model closely (with a preset error bar). Otherwise, the Fisher information matrix will not be defined correctly [9]. More importantly, we will need the error estimates corresponding to each frequency so that the Fisher matrix can be appropriately derived. Once the optimal input is known, we will once again follow the steps given above to approximate our transfer function and get a least squares fit. This will validate the hypothesis that the covariance is lower in the optimal excitation case in comparison to the white excitation case.

The Fisher Matrix will depend on the frequencies we study and the total Information Matrix will be the linear sum of the Fisher matrices at each frequency. For the system to be physically realizable (due to actuator constraints) the input force is constrained such that the net force is constant (and equal to the maximum force in Newton that can be obtained). We start with input force at each frequency to be distributed about a mean with a known variance. The algorithm redistributes the input force magnitude such that our determinant is maximised. In terms of the Bode plot, the magnitude of the force is higher around the resonant frequency for maximum information. Standard optimization algorithms can be used. This is a constrained optimization problem since we're trying to ensure that the input power (represented by input force, the ASD) is constant. The fisher matrix at each frequency observation (b) has the following expression.

$$\mathcal{I}_b = \begin{bmatrix} \frac{1}{\sigma_b^2}Re\left(\frac{\partial f_b}{\partial Q}\frac{\partial \bar{f_b}}{\partial Q}\right) & \frac{1}{\sigma_b^2}Re\left(\frac{\partial f_b}{\partial \omega_n}\frac{\partial \bar{f_b}}{\partial Q}\right) \\ \frac{1}{\sigma_b^2}Re\left(\frac{\partial f_b}{\partial Q}\frac{\partial \bar{f_b}}{\partial \omega_n}\right) & \frac{1}{\sigma_b^2}Re\left(\frac{\partial f_b}{\partial \omega_n}\frac{\partial \bar{f_b}}{\partial \omega_n}\right) \end{bmatrix} \tag{15}$$

$f_b = G(j\omega)F(j\omega) + \hat{n}$ where the terms have the same meanings as described in the previous section. Moreover, $\bar{x}$ and $Re(x)$ represent the conjugate and real parts of x respectively. σ_b^2 is the variance associated with each observation. While differentiating the likelihood function we use the magnitudes of the deviation. By deviation, we mean the difference between the

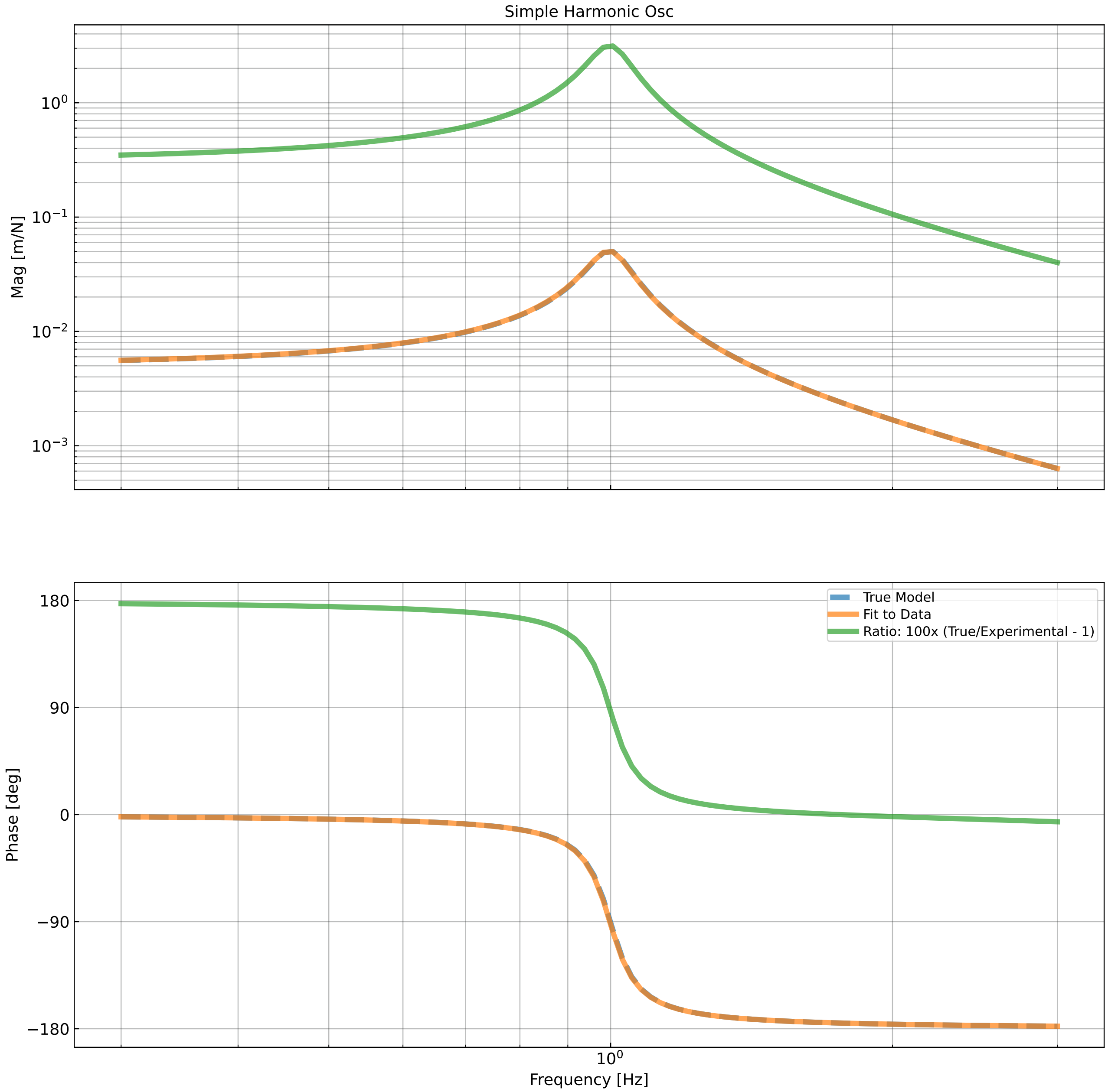

Figure 4: Least Squares Fitting for the Simple Harmonic Oscillator

experimental data $\hat{y}$ and the model y (Let $x = \hat{y} - y$). Since we're dealing with complex numbers, $|x|^2 = x\bar{x}$. Therefore the simplification yields the second order partial derivatives of x and $\bar{x}$ using the product rule. Using the property of complex numbers that $c + \bar{c} = 2Re(c)$ we get all the Fisher matrix terms to be real (illustrated mathematically in the appendix).

The PSD of the noise signal estimates the variance in the observation. Since the noise is Gaussian white noise, it has a constant magnitude for all frequencies.

Once we add the values of the Fisher matrices at each frequency and take its determinant, we will get our objective function. The constraint of the total input force $\mathcal{F}_{max}$ will be written as $(\sum_{k \in \mathbb{K}} |F(k)|^2)^{\frac{1}{2}} = \mathcal{F}_{max}$. $\mathbb{K}$ is the set of frequencies at which we are obtaining the measurements.

4.3 Least Squares Fitting

Python has several in-built functions to perform a least-squares fit. We look at multiple such algorithms and select the most promising one that satisfies our purpose. As a caution, it is important to note that we are dealing with complex data and need to come up with ways to represent it as real numbers (using residuals or stacking). This is important as all the solvers deal with real numbers alone. To reiterate, our problem is a non linear least squares fitting one. There are several parameters that go into deciding the algorithm. For instance, we'd like the covariance matrix to be returned along side the optimal parameters. This would save us the effort of writing an entirely new code followed by its testing and debugging. We would also like to see the optimizer give results that are not equal to the initial conditions (since we know that our initial conditions are not optimal).

After analysing several functions available under *scipy.optimize* like the curve-fit, least-squares, leastsq and the differential evolution, we narrowed down to least-squares and curve-fit. The least-squares function yields the results in Figure (5). However, it has a downside that it does not return the covariance matrix. Therefore, we moved on to the curve-fit algorithm as shown in Figure (6). leastsq did not return the reduced covariance and required back end coding to fix for the same. Our constraints were nonlinear and in a form different from differential evolution and that is why that particular method was not used.

4.4 Testing and Debugging

To test the accuracy of the Fisher Matrix code, it is always a good idea to compare it with results obtained on pen and paper. Hence, the matrix was evaluated at a single frequency (0.5Hz) analytically and the values obtained were compared with the pieces of code. The *sympy* and *numdifftools* libraries were the two tools that were used to calculate the information and both of them gave consistent results.

It is important that the code we have written is robust to variations in the input and does not crash while running. Thus, some sort of preliminary testing is always a good practice. For our purpose, we test the code for a subset of the $\mathbb{K}$ frequencies and ensure that the test passes. Only when we have this confirmation do we move on to the actual code block.

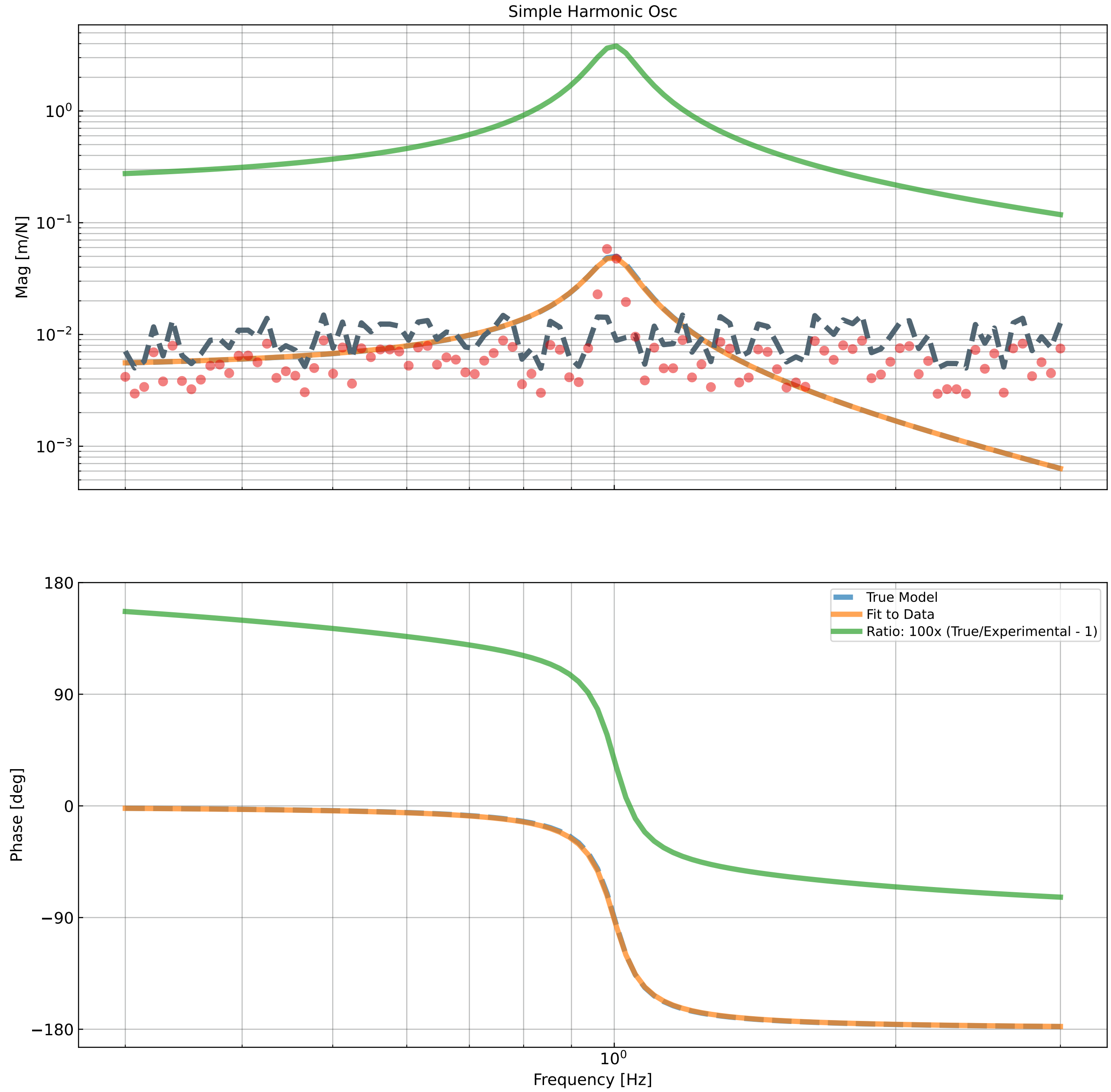

Figure 5: Least-Squares Fitting for the Simple Harmonic Oscillator with Optimal Excitation

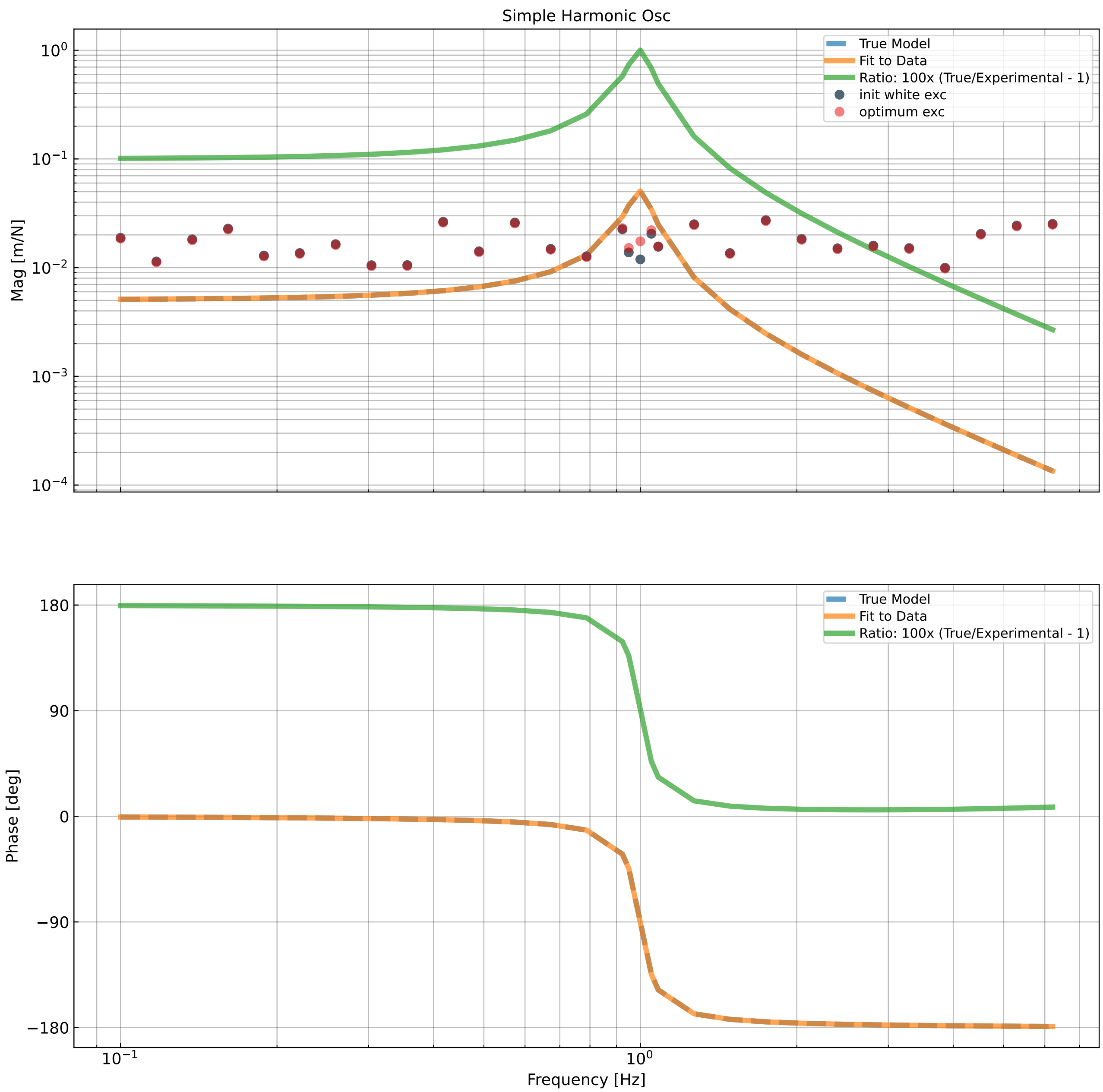

Figure 6: Curve-fit algorithm for the Simple Harmonic Oscillator with Optimal Excitation

For optimizing the Fisher matrix it becomes difficult to verify the correctness of the optimizer analytically. This is because the data size is large and the function is complex. The solver would have to iterate multiple times and this task becomes tedious when done manually. It would involve recalculating the Fisher Matrix at each iteration and taking its determinant. Future work includes obtaining a metric to validate this optimizer analytically.

4.5 Results

After trying various fitting techniques, we settled on the *curve-fit* algorithm. For the simple harmonic oscillator, we arrive at the following numbers. We tabulate the values of the parameters after fitting and the corresponding standard deviations as calculated from the covariance matrix (WE stands for White Excitation and OE stands for Optimal Excitation).

Parameter	True Value	WE	σ_{WE}	OE	σ_{OE}
Q	10	9.97345	0.0185351	9.997	0.00690703
Resonant Frequency [hz]	0.628319	6.27966	0.00056758	6.28006	0.000215723

Table 1: Simple Harmonic Oscillator: Fitting Results

As we can observe from Table 1, the uncertainty associated with the optimal excitation is less than the white excitation. This validates the need for optimizing the excitation spectrum. However, if we closely look at Figure (6) we note that the optimal excitation is not very different from the initial excitation provided. Hence, there is a need for an even better excitation algorithm, that significantly departs from the arbitrary initial conditions, while requiring a low number of function evaluations.

Alternative algorithms that are currently being explored include the *vect-fit* algorithm and the optimization power spectrum algorithm (algorithm 5.15 [5]). The latter follows an iterative approach and the returned optimal spectrum can be used by any estimator.

5 Representations in the Frequency Domain

Most of the experiments done in the time domain are finite time experiments (say, $t = 0$ to $t = T$). Hence, it is often necessary to represent this constraint in the fourier domain. For this, a simple route is to extend the signal to $-\infty$ to ∞. Thus, if we have a signal $f(t)$ defined between $t = 0$ to $t = T$, we can define $h(t)$ such that

$$h(t) = 0 \; t < 0 \tag{16}$$

$$h(t) = f(t) \; 0 < t < T \tag{17}$$

$$h(t) = 0 \; t > T \tag{18}$$

However, another concern over here is the continuity of $h(t)$. This can lead to spectral leakage. Hence, to mitigate this issue and enable a smooth transition, we make use of the windowing function [10]. This function is multiplied to the time domain function $h(t)$ so that we do not get unwanted frequency content in the fourier domain.

Additionally, a point to make here is the representation of the covarince matrix. Since our aim is to minimize the covariance matrix (i.e, maximise the information matrix) we would have to apply an optimization algorithm. Like least squares, and all the other such algorithms present, the input of such solvers are scalar functions. Hence we have to come up with a scalar representation or scalar measure of the matrix. There are several techniques including using the trace, the determinant, eigenvalues, matrix norms, all summarized and compared in [11]. According to this comparison, the determinant can be used in the context of covariance and the information matrix. Most commonly used measures are the trace and the determinant. The determinant would naturally capture more information if the parameters interact with each other. The trace would give information about the variance alone since it only reflects the diagonal elements. We won't get information regarding the correlations. The determinant requires a model before hand (D-optimality) and as that is a part of our analysis, we can incorporate this measure easily.

6 LIGO Suspensions

6.1 Damped and Undamped Models

In reality, the control of the suspended mirror system would work on the principle of feedback. Moreover, to prevent high amplitude oscillations, a damping velocity filter would be employed. For our model, we assume the filter to be a second order system and uniform across the three degrees of freedom. The system has two stable poles at -30Hz and one zero at 0. Its transfer function would then become $G(s) = \frac{k}{(s+\omega)^2}$ where K represents the gain and $\omega = 2\pi f$. Eq. (5)-(7) would be modified to include the feedback terms.

Figure 7 represents the block diagram for the modified system. The model produces a signal which is the mirror displacement (x) represented by the red arrow. The coupling with to the pitch angle and independence with respect to the yaw is also evident from the block diagram. We take the filter output and feed it back as a force. This feedback loop is given by G_x (Gx in the diagram). The other feedback loops are G_θ and G_ϕ. The new equations (neglecting frictional damping), expressed in the Laplace domain would be:

$$s^2X(s) + \omega_x^2X(s) = \omega_x^2(x_{sp} + b\Theta(s)) + G_xsX(S) \tag{19}$$

$$s^2\Theta(s) + \omega_\theta^2\Theta(s) = \frac{\omega_\theta^2}{l+b}(X(s) - x_{sp}) + G_\theta s\Theta(s) \tag{20}$$

$$s^2\Phi(s) + \omega_\phi^2\Phi(s) = \omega_\phi^2\phi_{sp} + G_\phi s\Phi(s) \tag{21}$$

Using these equations with $K = -1000$ (negative sign since the filter should ideally be in the LHS of Eq. (5)-(7)), we obtain a damped sinusoid response in the time domain. In the frequency domain, the behaviour of the Bode plot also changes. The magnitude of the peaks reduces compared to Figure 3. This peak can be tuned according to the requirements

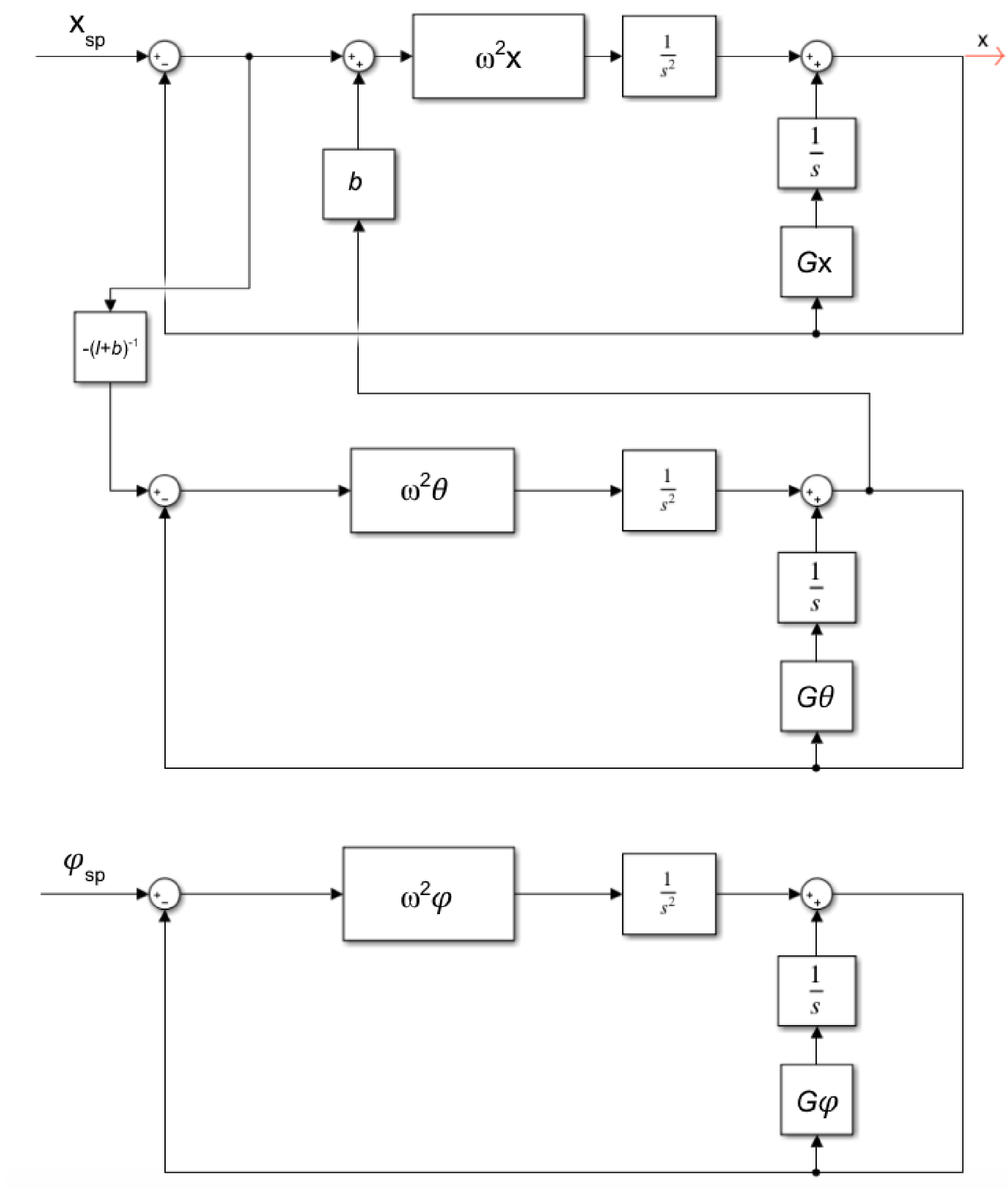

Figure 7: Block Diagram of the Damping Loop

by making use of the gain values. The damped system can now ensure that the excitation response is now minimized and won't disturb the overall set up.

6.2 Estimation Techniques and Future Work

Section 4.5 gave us insights into the various advantages and disadvantages of the employed system identification methods analysed. The framework developed for the Simple Harmonic Oscillators has also been extended and generalised to SISO LTI Systems. However, there is always some scope for improvement. As mentioned, algorithm 5.15 from [5] is currently under study. This optimal excitation can be used alongside basic curve fit algorithms. However, in the future, advanced estimation techniques like the Gaussian Maximum Likelihood Estimator followed by the local polynomial approximation approach from the same reference can also be looked into. A preliminary review on the Gaussian Maximum Likelihood Estimator is given below.

6.2.1 Implementing the Gaussian ML Estimator

As mentioned earlier, one of the system identification techniques we will be implementing is the Gaussian Maximum Likelihood Estimator on the damped (filtered) model. The plant model we will work with is the Position to Position transfer function. Since the original transfer function is already known, we find the frequency domain response to a designed excitation signal in order to generate the desired data. For the first round of implementation, we will be generating 50 multi sines with the fundamental harmonic as $f_o = 1Hz$ and the following frequencies of the form $f = kf_o$ where k goes from 1 to 50. This excitation will be such that the information matrix is maximized. To begin with, this would require optimal distribution of power within the desired frequency range. We will be making use of the iterative algorithm for the optimization of the power spectrum using the dispersion function. Once the optimal PSD is known, we can easily obtain the Amplitude Spectral Density of the input and can feed this input data to the true model for obtaining the output data in the frequency domain. This input-output data will be used for our Gaussian Maximum Likelihood Estimator optimization problem.

The entire optimization problem as described in Eq. (10-58) of [5] will be run in open loop. The open loop procedure does not require a controller and just works on the noise model and the plant model.

Once we have the cost function ready, the optimization problem is solved using the in built *scipy* optimization solver module. The method used for this nonlinear optimization is the Nelder Mead global minimizing algorithm. The advantage of such an algorithm is that it does not require the calculation of the Jacobian matrix. The solution obtained is then used to compare the experimentally obtained model to the true model. We perform this optimization problem under two cases: an ideal case with no measurement noise and a non-ideal case with the measurement noise represented by a white noise model.

1. Ideal Case: In the ideal case, as mentioned before, the measurement noise is absent. Hence, the result of the optimization problem is directly analyzed. In particular, we

look at the frequency response of the experimental and theoretical models as well as the ratio between the two.

2. Non-Ideal Case: In the non ideal case, we apply white noise to the measurement signal. Since the PSD of the white noise is uncorrelated and flat in the frequency domain, we simply add it to the original frequency response. This direct addition is a consequence of the superposition property of linear systems.Like in the previous case, we will again look at the frequency response of the experimental and theoretical models as well as the ratio between the two.

7 Project Timeline

7.1 Week 1-2

Obtain the state space formulation and frequency response of the Single Loop LIGO suspension. Alongside that, study the concepts of System Identification Theory and the Fisher Information Matrix.

7.2 Week 3

Instead of implementing system identification for a 6DoF, implement it on the 2DoF simple harmonic oscillator (SHO). Since the SHO is well studied, it will be easier to compare and analyse the system. Perform simple fitting based on white noise excitation.

7.3 Week 4-5

Understand the implementation and derivation of the Fisher Matrix by solving some toy problems by hand. Once the concept is well understood, extend it to the SHO system and write down its information matrix. Further, optimize this matrix (via determinant) and perform fitting under the resultant optimal excitation. Look for methods to represent the uncertainties associated with the fitting. Debug the code and work on improving the results by tuning the parameters.

7.4 Week 6-8

Generalise the SHO code to a SISO LTI system and look into methods for obtaining the Fisher Matrix without hard coding. Study the various analytical and numerical techniques and compare the performance. Also look into other fitting techniques to improve the SHO fitting results.

7.5 Week 9-10

Continue working on and improving the SISO LTI Identification techniques. Test and the debug the code so that the end product is robust. Work on the end term paper and presentation of whatever was studied and the results obtained.

8 Acknowledgements

I would like to thank my mentors Prof. Rana Adhikari, Rajashik Tarafdar, Anchal Gupta and Hang Yu for their constant guidance and support. I would also like to extend this gratitude towards the LIGO SURF program and National Science Foundation (NSF) for giving me the opportunity to work alongside scientists at LIGO on such a large scale project.

References

[1] Active reduction of residual amplitude modulation in free space eoms.

[2] Extending the reach of gravitational-wave detectors with machine learning.

[3] M Rakhmanov, D Reitze, D Tanner, S Yoshida, and H Yamamoto. Dynamical properties of ligo single loop suspended mirrors. Technical report, Citeseer, 2000.

[4] E. D. Hall. Fisher matrix methods for transfer function measurement, ligo. 2015.

[5] Rik Pintelon and Johan Schoukens. *System identification: a frequency domain approach.* John Wiley & Sons, 2012.

[6] Milind Kumar V. Optimal non-linear control for ligo interferometers, internal working note, ligo. 2019.

[7] Ankit Bansal and Veena Sharma. Design and analysis of robust h-infinity controller. *Control theory and informatics*, 3(2):7–14, 2012.

[8] Larry Price. Optimal experimental design: Introduction and application to system identification. Technical report.

[9] D Wittman. Fisher matrix for beginners. Technical report, UC Davis.

[10] What is windowing and when/why do we need it?

[11] Chun Yang, Lance Kaplan, and Erik Blasch. Performance measures of covariance and information matrices in resource management for target state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 48(3):2594–2613, 2012.

Appendix

8.1 Derivation of the Fisher Matrix

We will look at the following derivative in order to understand how the Fisher Matrix has been derived in [8]:

$$\frac{\partial^2}{\partial\alpha\partial\beta}(|v-w|^2) = \frac{\partial^2}{\partial\alpha\partial\beta}(v-w)(\bar{v}-\bar{w}) \tag{22}$$

Here, $v, w \in \mathbb{C}$ and while, v is a constant and w is a variable. $\bar{x}$ is the complex conjugate of x.

$$\begin{aligned}\frac{\partial^2}{\partial\alpha\partial\beta}(v-w)(\bar{v}-\bar{w}) &= \frac{\partial^2}{\partial\alpha\partial\beta}(v\bar{v}-v\bar{w}-w\bar{v}+w\bar{w})\\ &= \frac{\partial}{\partial\alpha}\Big(-v\frac{\partial\bar{w}}{\partial\beta}-\bar{v}\frac{\partial w}{\partial\beta}+\bar{w}\frac{\partial w}{\partial\beta}+w\frac{\partial\bar{w}}{\partial\beta}\Big)\\ &= -v\frac{\partial^2\bar{w}}{\partial\alpha\partial\beta}-\bar{v}\frac{\partial^2 w}{\partial\alpha\partial\beta}+\bar{w}\frac{\partial^2 w}{\partial\alpha\partial\beta}+w\frac{\partial^2\bar{w}}{\partial\alpha\partial\beta}+\frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}+\frac{\partial\bar{w}}{\partial\beta}\frac{\partial w}{\partial\alpha}\\ &= \frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}+\frac{\partial\bar{w}}{\partial\beta}\frac{\partial w}{\partial\alpha}-(\bar{v}-\bar{w})\frac{\partial^2 w}{\partial\alpha\partial\beta}-(v-w)\frac{\partial^2\bar{w}}{\partial\alpha\partial\beta}\end{aligned}$$

If $(v-w)\to 0$ (in our context, this corresponds to unbiased estimates), we get:

$$\begin{aligned}\frac{\partial^2}{\partial\alpha\partial\beta}(v-w)(\bar{v}-\bar{w}) &= \frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}+\frac{\partial\bar{w}}{\partial\beta}\frac{\partial w}{\partial\alpha}\\ &= \frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}+\overline{\frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}}\end{aligned}$$

Using $x+\bar{x}=2Re(x)$ where $Re(x)$ represents the real part of x, we can simplify the above to:

$$\frac{\partial^2}{\partial\alpha\partial\beta}(v-w)(\bar{v}-\bar{w}) = 2Re\Big(\frac{\partial w}{\partial\beta}\frac{\partial\bar{w}}{\partial\alpha}\Big) \tag{23}$$

Now we extend this to the definition of the Fisher Matrix terms $\mathcal{F}$ where we differentiate the negative log likelihood function with the parameters.

$$\mathcal{F}_{ij} = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\Big(\sum_\alpha \frac{|y_\alpha-\hat{y}_\alpha(\theta)|^2}{2|n_\alpha|^2}\Big)$$

For Gaussian white noise, n_α is constant. Moreover, we can assume unbiased estimates. Hence, using Eq. 23, we get:

$$\mathcal{F}_{ij} = \sum_\alpha \frac{1}{|n_\alpha|^2}Re\Big(\frac{\partial\overline{\hat{y}_\alpha}}{\partial\theta_i}\frac{\partial\hat{y}_\alpha}{\partial\theta_j}\Big) \tag{24}$$