

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

Technical Note	LIGO-T2500074-v1	2025/11/02
Suspension Modelling Research		
J.R. Vishwanath		

California Institute of Technology
LIGO Project, MC 100-36
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

Contents

1	Overview	2
2	Motivation	2
3	Introduction	2
3.1	Gravitational Waves	3
3.2	LIGO	3
3.3	40m Prototype at Caltech and Small Optic Suspensions	4
3.4	Transfer Function from Analytical Model	5
3.5	Transfer Function from Active Control Loop Signals	7
4	Objective	9
5	Approach	9
6	Collection and Processing of Data from the 40m	9
6.1	Collection of Data by Manual Optimisation of Excitation.	9
6.2	Processing of Data	11
7	Modelling Details	13
7.1	Overview of Simscape Multibody	13
7.2	Simscape Multibody/Matlab Model Configurations	18
7.3	State-Space Representation and Function Data andfrom Simscape	28
8	Attempt at Automating Excitation Optimisation.	29
9	Conclusion and Recommendation for Future Work	30

1 Overview

This project explored the suitability of using Simscape, or more specifically, Simscape Multibody (see [subsection 7.1](#)) to model suspensions as a first step towards building a digital twin (an online virtual interferometer that digitally mimics the operation and dynamics) of the 40m prototype of LIGO at Caltech. The following steps, were followed in an attempt at making a digital twin of the small optic suspension. First, a Simscape Multibody model based on a simplified schematic of the 40m small optic suspension assembly which has an analytically known solution was made. Then, transfer function data from the Simscape Multibody model and the analytical solution, was compared to see the accuracy of the Simscape Multibody model. Next, open loop transfer function data was processed from the active control loop signals of the damping filters employed in the small optic suspension in the 40 m prototype. The Amplitude Spectral Density of the excitation used to obtain this transfer function data, is optimised manually, so as to improve the coherence at each frequency bin. The transfer function data obtained experimentally using manual optimisation was also compared against the transfer function generated by the Simscape multibody model. Finally, the effectiveness of parameters characterizing the transfer function data from Simscape multibody model to be used as a reference for optimising the excitation automatically and self-updating using Pintelon and Schoukens approach was tested.

2 Motivation

Building a digital twin is of interest, as it can then be used for computing the optimum excitation needed to obtain accurate transfer function data, which in turn is needed for building active controls. The excitation calculated using a digital twin will be more optimum as compared to excitation calculated based on an analytical model, as analytical models cannot account for deviations of the system from ideal conditions arising due to assembly errors, operational wear, environmental conditions etc. Further, a digital twin also enables testing of new active control strategies without causing downtime or any risk to the physical system. In this context, Simscape Multibody is a good candidate for modelling suspensions as it is built on top of simulink, a software widely used for control system design and matlab, a software used for modelling systems with cross - couplings addressed through matrix formalism. Additionally, the ability to intergrate with Simscape foundation library provides the possiblity of modelling externalities such as vacuum and cryogenic conditions, which may be relevant for designing the next generation of suspensions and associated active controls for Cryogenic Gravitational Wave detectors.

3 Introduction

3.1 Gravitational Waves

From the theory of general relativity proposed by Einstein in 1916, we understand that massive and energetic objects, can cause curvatures of spacetime as governed by the equation:-

$$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \quad (1)$$

Where $G_{\mu\nu}$ encodes the curvature of spacetime, $T_{\mu\nu}$ represents the matter-energy content of spacetime, G is Newton's gravitational constant ($6.674 \times 10^{-11} \text{ m}^3\text{kg}^{-1}\text{s}^{-2}$) and c is speed of light in vacuum. This causes other less massive, less energetic objects in the vicinity to follow the curved path (geodesics) created by the massive, energetic body which we perceive as gravity. Although gravity causes masses to accelerate, such acceleration cannot always produce gravitational waves. Only time varying quadrupolar moment or matter accelerated in an asymmetric manner causes gravitational waves as monopolar and dipolar radiation are prohibited by law of conservation of mass and momentum. These gravitational waves due to their peculiar origin, acts on test masses in such a way so to create asymmetry in its distribution. This effect can be characterised by a quantity called strain amplitude h defined by $h = \frac{2\Delta L}{L}$, where ΔL is the change in separation of two masses a distance L apart. Due to the extremely small value of G and $\frac{1}{c^4}$ in Eqn 1, detectable levels of h is only produced when large compact masses are accelerated in an asymmetric fashion. This condition is met when astronomical events like, supernova formation, coalescence of compact binaries occur. Despite, these extreme astrophysical conditions necessary to produce gravitational waves, the detectable levels of h is still very tiny of the magnitude of 10^{-20} . Gravitational wave observatories like LIGO are built to study such events.

3.2 LIGO

The Laser Interferometer Gravitational-wave Observatory, abbreviated as LIGO is an earth-based network of two gravitational wave detectors located in Hanford and Livingston. LIGO is essentially a large Michelson Interferometer, with fabry-perot cavities along the 4-km-long arms and with mirrors coated onto test masses that respond to gravity. However, environmental disturbances can cause displacements much larger in magnitude as compared to the displacements gravitational waves can induce on the test masses. To make sure only gravitational waves, can move these test masses i.e. to isolate them from environmental disturbances. Various active and passive control strategies are employed by LIGO. One of these passive control strategy involve, hanging the test masses and other optics from a chain of pendula with their characteristic frequencies carefully chosen, so as to act like a mechanical low pass filter against environmental disturbances [19]. During the various observation runs, LIGO has detected gravitational waves from different sources revealing various facts.

- The first detection titled GW150914 was inferred to be from the merger of two stellar-mass black holes by matching the observed dynamics of the signal with the dynamics modelled using the theory developed Kerr, Schwarzschild and Einstein. The Initial mass of the two black holes (showed heavy black holes can exist), the need for weak stellar winds and low metallicity for heavy black holes to form, their combined final mass, the deficit mass expelled as gravitational waves, their final spin, upper bound to

its distance from us were all determined from the signal and found to be consistent with the predictions of general relativity in the strong- field regime, therefore becoming its first experimental validation.[4]

- The second confirmed detection titled GW151226 was also from a binary black hole coalescence. This was the first candidate signal that passed the threshold for an alert to the electromagnetic partners. Coarse sky localization enabled the first attempt at multi-carrier astronomy. A small probability that the mass of the secondary black hole falls in the mass gap between that of neutron stars and black holes was posited.[5]
- GW170817 was the first detection event to be simultaneously detected by the electromagnetic partners of LIGO through gamma ray burst 170817A enabled by the extremely precise sky localization of the event. The data revealed the GW was from a Binary neutron star merger, which was further confirmed by the observation of gamma ray bursts. This simultaneous observation of GW and gamma ray bursts have enabled us to test the difference between speed of light and speed of gravity, such tests enabled us to place new bounds on local Lorentz invariance violations. The improved constraintment of sky location of GW170817 also helped us prove that GW has pure tensor polarization modes.

However, the current LIGO architecture has many factors that limit to what extent noises like quantum radiation pressure noise, shot noise, mirror thermal noise, control noise, suspension thermal noise, and Newtonian gravity noise can be reduced:-

- An upper threshold to how much optical power can be stored in the arms before, thermally induced wavefront distortion occurs determined by the thermal conductivity of the test masses.
- Increase of thermo-elastic noise, with increasing conductivity at room temperatures.
- Low moment of inertia, making the masses vulnerable to radiation pressure torques, leading to fluctuations in alignment and controls noise thereof.
- High dissipation in amorphous materials used to make mirror coatings.

3.3 40m Prototype at Caltech and Small Optic Suspensions

A 40 m prototype of the LIGO interferometer is located on the Caltech Campus. It has the same dual- recycled Fabry-Perot Michelson Interferometer configuration as the LIGO. It is used to test upgrades, new control and length sensing schemes before being implemented on the LIGO sites[20]. Unlike the sites however, the 40m prototype employs a single-loop wire suspension, made from steel music wire for many of its optics [10][9]. The schematic of the same can be seen in Figure 3 and Figure 1. The Analytical Model, detailing the various parameters of the small optic suspension such as its mass, its effective length etc can be found in the LIGO document **LIGO-T000134** [18]. The procedure to obtain transfer function plots from the Analytical model is presented in subsection 3.4. The various resonant modes of the suspension lie between the range of 0.1Hz - 10Hz. Therefore, all comparisons between

transfer functions obtained from the different models (Analytical, Simscape Multibody) and experimental data in this project will also be made in the same range. The experimental data, needed for this project is collected from Active control loop of the damping filter used in ETMX (the test mass at the end of X- ARM with mirror coated onto it) as well as from SRM (Signal-Recycling Mirror) with the help of softwares like Diagnostic Test Tools GUI (diaggui), Foton and getdata function in cdsutil package in python [14] [11] [15] [16]. The algebra behind obtaining transfer function from the signals of Active control loop is presented in [subsection 3.5](#) and the technical details regarding how the data is obtained and processed is presented in [subsection 6.1](#).

3.4 Transfer Function from Analytical Model

As discussed in [section 1](#), we need to compare the transfer functions obtained from the Simscape Multibody model against transfer functions obtained from an analytical model to validate the Simscape Multibody model. The Analytical model, used in this project for comparison is the one described in the LIGO document **LIGO-T000134** [18]. [Figure 1](#) is copied from the same document and shows the schematic of the Small Optic Suspension. The values of the different dimensions in the schematic such as $R1$ $R2$, l , b etc are listed in **Table 1** of the document. These dimensions were used for building the model in Simscape Multibody. [Figure 2](#) is also copied from the same documents and it illustrates the co-ordinates system used to develop the equations of motions for the system. The spherical co-ordinates θ , ϕ and the cartesian co-ordinate x are called *pitch(PIT)* and *YAW* and *position(POS)* as per LIGO naming convention. Transfer functions are mathematical relationships, between the different inputs and outputs of a system in the Laplace/Fourier domain. For example, the transfer function we will obtain from the Simscape Multibody model, will relate the force/torque applied along the centre of mass of the suspended mirror (input) to the displacements along θ , ϕ and x (output). However in an analytical model, a system is initially described using equations of motion. The equations of motions in the document T000134 that capture this situation are equations 69-71. They are reproduced here as equations 8-10. Of which, [Equation 2](#) and [Equation 3](#) are coupled equations.

$$\ddot{x} + \gamma_x \dot{x} + \omega_x^2 x = \omega_x^2 (x_{sp} + b\theta) + f_x(t) \quad (2)$$

$$\ddot{\theta} + \gamma_\theta \dot{\theta} + \omega_\theta^2 \theta = \frac{\omega_\theta^2}{l+b} (x - x_{sp}) + f_\theta(t) \quad (3)$$

$$\ddot{\phi} + \gamma_\phi \dot{\phi} + \omega_\phi^2 \phi = \omega_\phi^2 \phi_{sp} + f_\phi(t) \quad (4)$$

In these equations, ω_x^2 , ω_θ^2 and ω_ϕ^2 are the natural frequencies of free oscillations of the mirror in the respective co-ordinates, and they can be calculated from $\frac{g}{l}$, $\frac{mg}{I_\theta l} b(l+b)$ and $\frac{mg}{I_\phi l} R_1 R_2$ respectively. The numerical values of these natural frequencies have been tabulated in **Table 3** of the reference document. Similarly, the terms γ_θ , γ_ϕ and γ_x in the equations are viscous damping co-efficients. The term x_{sp} denotes translation of the suspension point, however in our case, this can be considered equal to zero. The value of these damping coefficients have not been explicitly mentioned in the document. To obtain, transfer function from these equations of motion, we need to laplace transform these equations and keep the force term

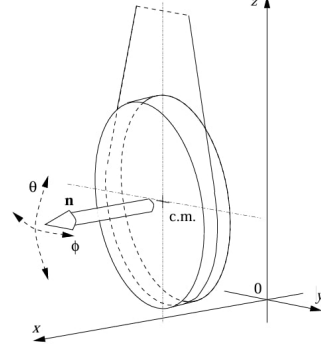
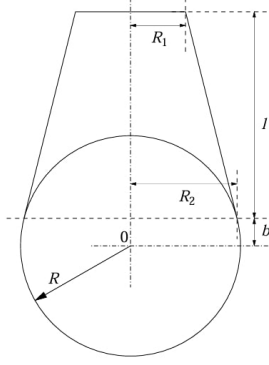


Figure 1: Schematic of SOS from T000134 Figure 2: Schematic of the co-ordinate system showing the various parameters of the suspension such as R_1 , R_2 , l , and b used to define equations of motion

alone in the RHS. Then, the coupled Equation 2 and Equation 3 is cast into a matrix form as shown in Equation 5. While, Equation 4 becomes Equation 6. The inverse of the first

$$\begin{bmatrix} s^2 + \gamma_x s + \omega_x^2 & -\omega_x^2 b \\ -\frac{\omega_\theta^2}{l+b} & s^2 + \gamma_\theta s + \omega_\theta^2 \end{bmatrix} \begin{bmatrix} X(s) \\ \Theta(s) \end{bmatrix} = \begin{bmatrix} F_x(s) \\ F_\theta(s) \end{bmatrix} \quad (5)$$

$$(s^2 + \gamma_\phi s + \omega_\phi^2) \Phi(s) = F_\phi(s) \quad (6)$$

matrix in Equation 5 gives us the 4 transfer functions related to x and θ . Explicitly, it can be written down as follows:-

$$\mathbf{H}(s) = \frac{1}{\text{determinant}} \begin{bmatrix} s^2 + \gamma_\theta s + \omega_\theta^2 & \omega_x^2 b \\ \frac{\omega_\theta^2}{l+b} & s^2 + \gamma_x s + \omega_x^2 \end{bmatrix} \quad (7)$$

where determinant is given by the expression $(s^2 + \gamma_x s + \omega_x^2)(s^2 + \gamma_\theta s + \omega_\theta^2) - \frac{\omega_x^2 \omega_\theta^2 b}{l+b}$. Assigning the standard notation of $a_{11}, a_{12}, a_{21}, a_{22}$ i.e. $(a_{\text{row}_{\text{position}}, \text{column}_{\text{position}}})$, to the elements of matrix $\mathbf{H}(s)$, each terms can be interpreted as follows:-

- $\frac{a_{11}}{\text{determinant}}$ is the transfer function from $F_x(s)$ to $X(s)$ or POS(Force) - POS(Displacement)
- $\frac{a_{12}}{\text{determinant}}$ is the transfer function from $F_\theta(s)$ to $X(s)$ or PIT(Force) - POS(Displacement)
- $\frac{a_{21}}{\text{determinant}}$ is the transfer function from $F_x(s)$ to $\theta(s)$ or POS(Force) - PIT(Displacement)
- $\frac{a_{22}}{\text{determinant}}$ is the transfer function from $F_\theta(s)$ to $\theta(s)$ or POS(Force) - PIT(Displacement)

Similarly, for Equation 6 the transfer function from $F_\phi(s)$ to $\Phi(s)$ or YAW(Force) - YAW(Displacement) is:-

$$\mathbf{H}(s) = \frac{1}{(s^2 + \gamma_\phi s + \omega_\phi^2)} \quad (8)$$

Now since $s = i\omega$ we can obtain plots for all the transfer functions analytically which can be compared to the plots generated from the Simscape Multibody model.

3.5 Transfer Function from Active Control Loop Signals

Active controls are employed in LIGO for multiple purposes such as seismic isolation through vibration compensation, locking the cavities of the interferometer etc. However, in this project the active control of interest is the one employed to damp the resonances of the suspension pendula. The basic idea is as follows the co-ordinate of interest (POS, PIT, YAW in our case see Figure 2 and Figure 1) of the suspended mirror is constantly monitored using a **Sensor**, when any deviation from the desired position with frequency around the mechanical resonance of the suspension is detected, the **Actuator** acts on the suspended mirror with force just enough, to counteract the undesired displacement. A genral active control loop is shown alongside the CAD drawings of the 40m Small Optic Suspension in Figure 3. The function of the **Sensor** and **Actuator** in the control loop is performed by the OSEMs (cylinders marked by black arrows in the CAD drawings) in Figure 3. The magnitude and direction in which the counteracting force by the **Actuator**, needs to be applied is calculated by the **Kontroller** based on the signal from the sensor and the knowledge of how the system will respond to a particular magnitude and direction of force. This knowledge, can be represented by the open loop transfer function of the system comprising the **Sensor**, the **Plant** (the suspended mirror) and the **Actuator**.

The following equations detail, how we can obtain the open loop transfer function of the system represented by *SPA* from the signals *IN1*, *IN2* and *EXC* shown in Figure 3. The addition of signals in the mixer and the propogation of signal *IN2* along the loop can be represented using Equation 9 and Equation 10 respectively. Here *K* is the effect of **Kontroller** on the signal.

$$IN2 = IN1 + EXC \quad (9)$$

$$IN1 = -SPA K(IN2) \quad (10)$$

If we now substitute Equation 10 in Equation 9 we get the following equation :-

$$IN2 = -SPA K(IN2) + EXC \quad (11)$$

This Equation 11 can be re-arranged as follows :-

$$\frac{IN2}{EXC} = \frac{1}{(1 + SPA K)} \quad (12)$$

Similarly we can get $\frac{IN1}{EXC}$ by multiplying $\frac{IN1}{IN2}$ and $\frac{IN2}{EXC}$. From re-arranging Equation 10 we get $\frac{IN1}{IN2}$ as $-SPA K$, while from Equation 12, we already know $\frac{IN2}{EXC}$. Therefore we get :-

$$\frac{IN1}{EXC} = \frac{1}{(1 + SPA K)} SPA K \quad (13)$$

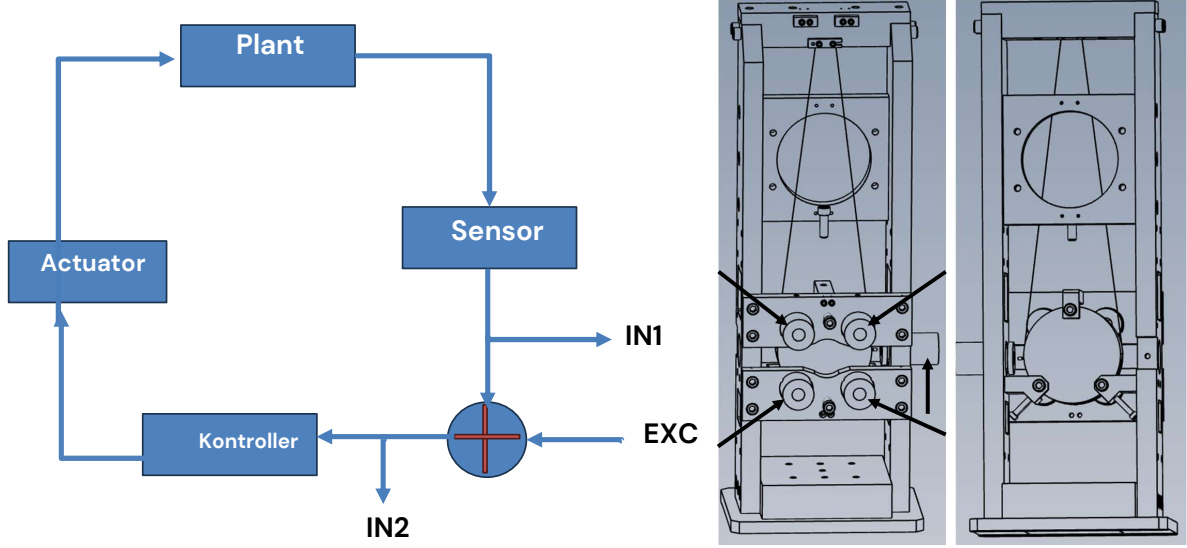


Figure 3: Schematic representation of a general control loop (left) alongside the CAD drawings of the back and front side of the 40m Small Optic Suspension

Now if we multiply [Equation 13](#) by the inverse of [Equation 12](#) and the inverse of the controller, we get SPA or the desired system response as shown below:-

$$\frac{IN1}{EXC} \frac{EXC}{IN2} K^{-1} = \frac{1}{(1 + SPAK)} SPAK (1 + SPAK) K^{-1} = SPA \quad (14)$$

[Equation 14](#) shows the mathematics behind one way of obtaining the open loop transfer function between a pair of force-displacement co-ordinates of the system experimentally. The same equation, when implemented in a matrix format gives the open loop transfer function between all 9 pairs of force-displacement combinations arising from the 3 co-ordinates of interest (POS, PIT and YAW). More technical details regarding the same, is addressed in [section 6](#). It is also possible, to obtain the transfer function using another signal OUT , which would not involve inverting, K , however that method was not followed in this project.

4 Objective

Our objective is to take the first step towards building a real-time digital clone of the 40m prototype by simulating the physics underpinning the SOS suspensions using Simscape modelling. These models will then be updated in such a way that the transfer function data obtained from them, matches the transfer function data obtained from the 40m SOS assembly. This will enable us to account for the deviations from idealised construction during actual assembly which would not be possible through analytical models.

5 Approach

The objective stated earlier will be achieved through the following stages:-

- **Modelling of SOS Assembly in Simscape:** Initially a simplified version of the current 40 m suspension assembly will be modelled on Simscape Multibody. The model will be simulated, with excitation along one degree of freedom at a time, with simulation time set to zero to find initial steady state operating point. The model, is the linearised around this operating mode using command in Matlab. This linearised model is stored as a time domain representation in the form of ***ABCD*** matrix, which is then converted into frequency domain representation and plotted as bode plots of transfer function.
- **Collection of SOS Assembly data from 40m prototype:** Data will be collected from the 40m SOS assembly, operated with properly filtered excitation along one degree of freedom at a time, in .xml format. Once data corresponding to the three possible excitations are collected. It is structured into two 3-D transfer function matrices and two 3-D Signal to Noise Ratio matrices corresponding to (Residual Input Power(EXC)—Residual Background Noise(IN1)) and (Residual Input Power(EXC)—Output Power (IN2)) and then stored in a .mat file using a python script.
- **Processing of data in Matlab:** The data is processed in Matlab to obtain transfer function data plots that will be comparable to those generated from the simulation.
- **Matching of Simscape Model, Analytical Model and Measurement:** The processed data sets will be compared against each other and the model will be modified to generate simulation data that matches the measurement data

6 Collection and Processing of Data from the 40m

6.1 Collection of Data by Manual Optimisation of Excitation.

The collection of data from the 40m prototype is performed using diaggui. [Figure 4](#) copied from [\[14\]](#) shows the *Measurement* tab of diaggui software[\[12\]](#). The first box in the *Measurement* tab is also denoted by the same name. Here, we specify the type of transfer function measurement we wish to make. In this project , we used *Fourier Tools* option as it is the

fastest way to measure transfer function as all frequencies are measured simultaneously. *Swept sine response* on the other hand, measures responses for each individual sine wave in increments of frequency and therefore gives more accurate results at the expense of time. Further, *Fourier Tools* is also reasonably accurate for measuring a small range of frequencies, like in our case (0.1 Hz - 10 Hz) as the issue of signal spreading over a large space of frequency and therefore becoming weak does not arise [11] [14]. Moreover, since we expect large resonance peaks in the range of frequencies we are measuring, using *Swept Sine Repsone* will not be ideal as the suspension will continue to be in resonance while frequencies next to the resonant peaks are measured. In the next box in the tab labelled *Measurement Channels* we enter the name of channels from which we want to collect signals. From the discussion in subsection 3.5, we saw that in order to get the transfer function of the system represented by *SPA*, we need to record the signals IN1, IN2 and EXC. Further, since we are interested in 3 degrees of freedom (POS, PIT, YAW) this implies that we have to collect data from 9 channels. These channels follow the naming convention of C1:SUS-ETMX__SUSdegree-of-freedom__signal-name. In the *Fourier Tools* box we keep the default settings in most of the parameters. The only changes made are *Start* is set to 0.005Hz *Stop* is set to 200Hz and *BW* is set to 0.01Hz and *Averages* is set to 10. A discussion regarding the acceptable values that can be entered for the parameters can be found in section 4.2 of [12] and section 3.1 and 4.4.3 of [11]. Figure 5 shows the next tab we must configure in the diaggi software. We perform one excitation at a time along each of the co-ordinates of interest (POS, PIT, YAW). The *Waveform* option is set as gaussian noise. Designing, the *filter* using *Foton* is the most important aspect of getting reliable transfer function data. The filter, is initially designed by making sure through inspection of eye that amplitude/ power of the Injected signal (Excitation-EXC) significantly exceeds the power/amplitude of the background noise (IN1) in every frequency bin between 0.1 Hz -10Hz when plotted together. This is done to naively ensure the Signal/Noise ratio is high. Once the first results are obtained, we can in the results tab plot the coherence function between IN1-EXC and IN2-EXC. Since, coherence measures how related the variation of two signals are to get reliable transfer function data we want the signals, to be as coherent as possible. Therefore, we further optimize the filter and/or change the *amplitude* of the excitation until coherence is above 0.7 in every frequency bin between. The choice of achieving coherence value above 0.7 can also be explained through the following equations.

$$\gamma(f) = \sqrt{\frac{SNR(f)}{1 + SNR(f)}} \quad (15)$$

Where, $\gamma(f)$ stands for the frequency spectrum of coherence function and $SNR(f)$ is the frequency spectrum of the signal to noise ratio. Equation 15 is a well known result in signal processing [13]. Now consider, what happens when the value of $\gamma(f) = \frac{1}{\sqrt{2}} \approx 0.7$. Substituting this value of $\gamma(f)$ into Equation 15 and solving for $SNR(f)$ we get $SNR(f) = 1$. In other words, by obtaining a coherence above 0.7 in every frequency bin, we are making sure the signal-noise ratio is greater than 1. An example of how changing the Amplitude Spectral Density (ASD) by modification of the filter in *Foton* improves coherence is illustrated by Figure 7 and Figure 6. This process of excitation optimisation, is repeated for the other 2 co-ordinates of interest, collecting data from all the 9 channels each time. This results in 3 sets of data, each corresponding to excitation along a different co-ordinate amongst the three co-ordinates of interest. These 3 sets of data are saved as three files in .xml format.

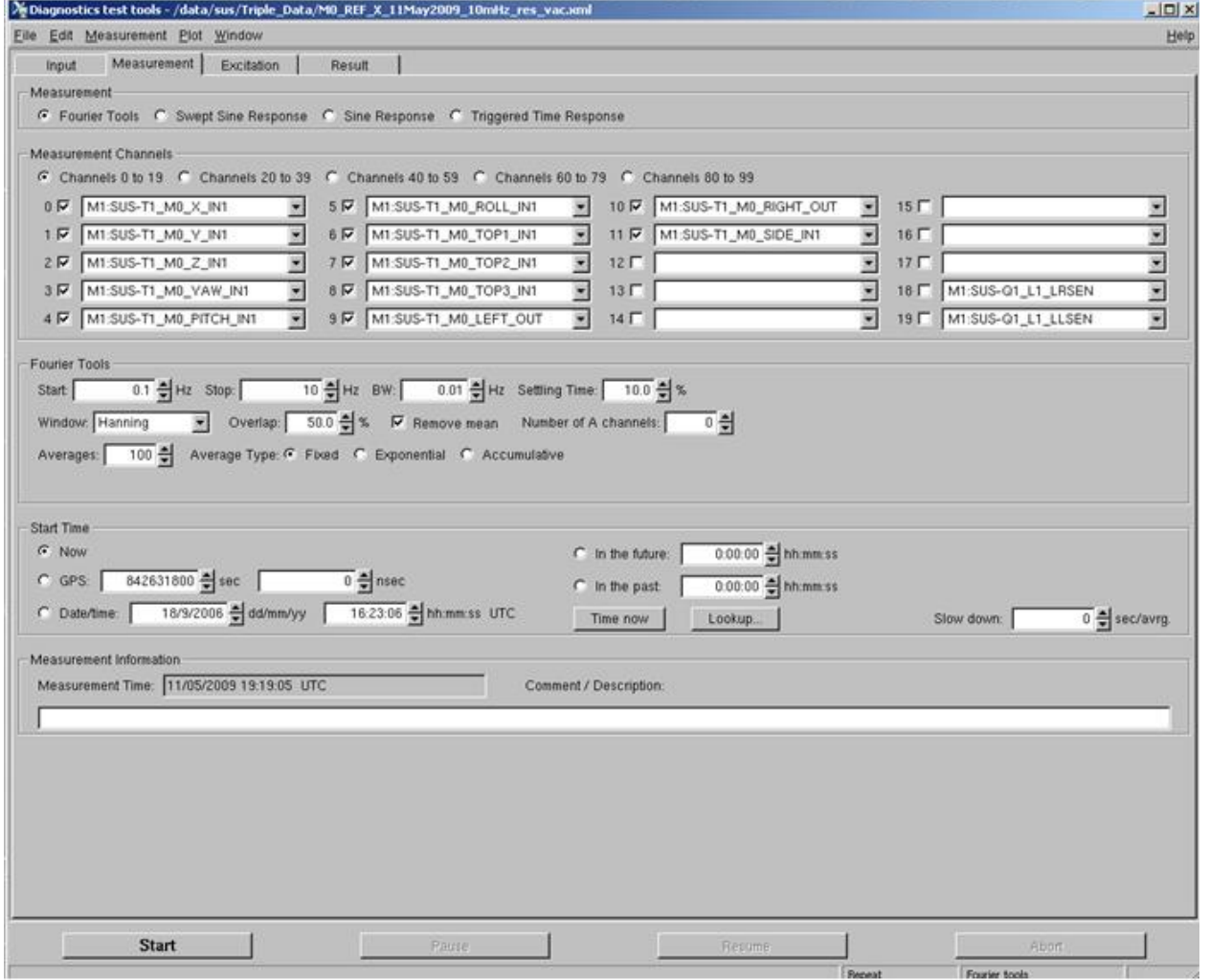


Figure 4: An image of the measurement tab inside daiggui software copied from [14]

6.2 Processing of Data

In subsection 3.5 the mathematics behind, how to obtain transfer function of a system from control signals IN1, IN2 and EXC was worked out. Practically implementing Equation 14 for a multiple input $(F_x(s), F_\theta(s), F_\phi(s))$, multiple output $(X(s), \theta(s), \phi(s))$ system like the small optic suspension involves, obtaining transfer function data representing $\frac{IN1}{EXC}$ and $\frac{IN2}{EXC}$ for each force-displacement pair from the data stored in the three files in .xml format and assembling them into appropriate matrices. The required structure of $\frac{IN1}{EXC}$ matrix is given by Equation 16 and that of $\frac{IN2}{EXC}$ is given by Equation 17. We notice, that the co-ordinate of excitation changes for each column in the matrices. This implies, that the data corresponding

$$\begin{bmatrix} TF(\text{POS-IN1, POS-EXC}) & TF(\text{POS-IN1, PIT-EXC}) & TF(\text{POS-IN1, YAW-EXC}) \\ TF(\text{PIT-IN1, POS-EXC}) & TF(\text{PIT-IN1, PIT-EXC}) & TF(\text{PIT-IN1, YAW-EXC}) \\ TF(\text{YAW-IN1, POS-EXC}) & TF(\text{YAW-IN1, PIT-EXC}) & TF(\text{YAW-IN1, YAW-EXC}) \end{bmatrix} \quad (16)$$

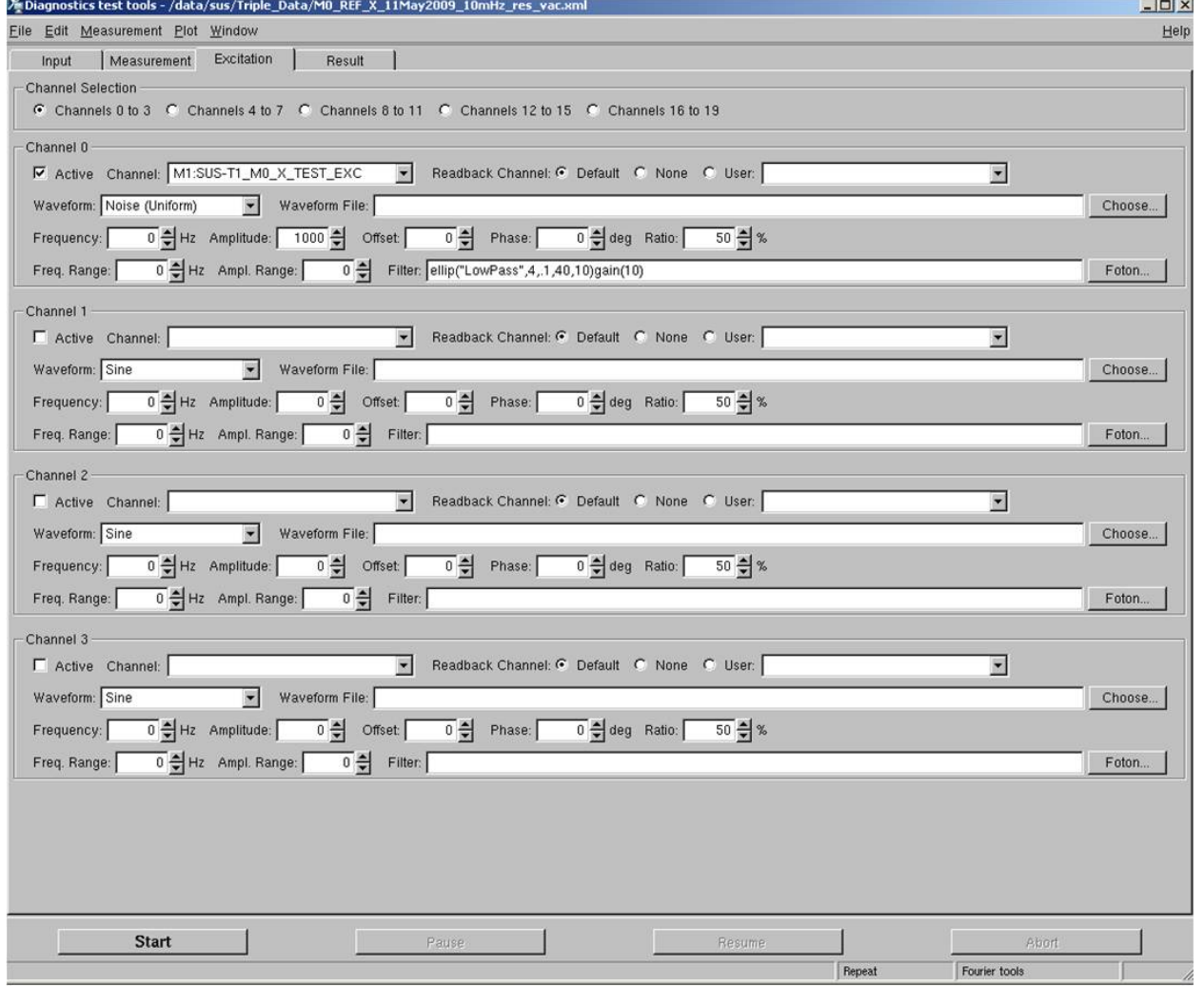


Figure 5: An image of the excitation tab inside daiggui software copied from [14]

$$\begin{bmatrix} TF(\text{POS-IN2, POS-EXC}) & TF(\text{POS-IN2, PIT-EXC}) & TF(\text{POS-IN2, YAW-EXC}) \\ TF(\text{PIT-IN2, POS-EXC}) & TF(\text{PIT-IN2, PIT-EXC}) & TF(\text{PIT-IN2, YAW-EXC}) \\ TF(\text{YAW-IN2, POS-EXC}) & TF(\text{YAW-IN2, PIT-EXC}) & TF(\text{YAW-IN2, YAW-EXC}) \end{bmatrix} \quad (17)$$

to the three different columns must be obtained from the three different files corresponding to excitations along the three different co-ordinates of interest. The TF or transfer function operation is performed for each term in the matrix using the **.xfer(chA,chB)** method of **dttxml** package [17]. The **.xfer()** method, returns a transfer function object from which, the array of frequencies, the complex array of the values of transfer function at each frequency bin, and the array of values of SNR estimate at each frequency bin can be obtained. Figure 8 is a plot of data obtained in this manner for the first term in the matrices 16 and 17. The Kontroller response is obtained using the **.FilterDesign(design, rate)** method in **foton** package. It is then assembled into a diagonal matrix. Now if we multiply the matrix in 16 with the inverse of the matrix in 17 and then by the inverse of Kontroller response we obtain, the open loop transfer function data of the small optic suspension as shown by Equation 14. This open - loop transfer function data, can be compared to the transfer functions obtained

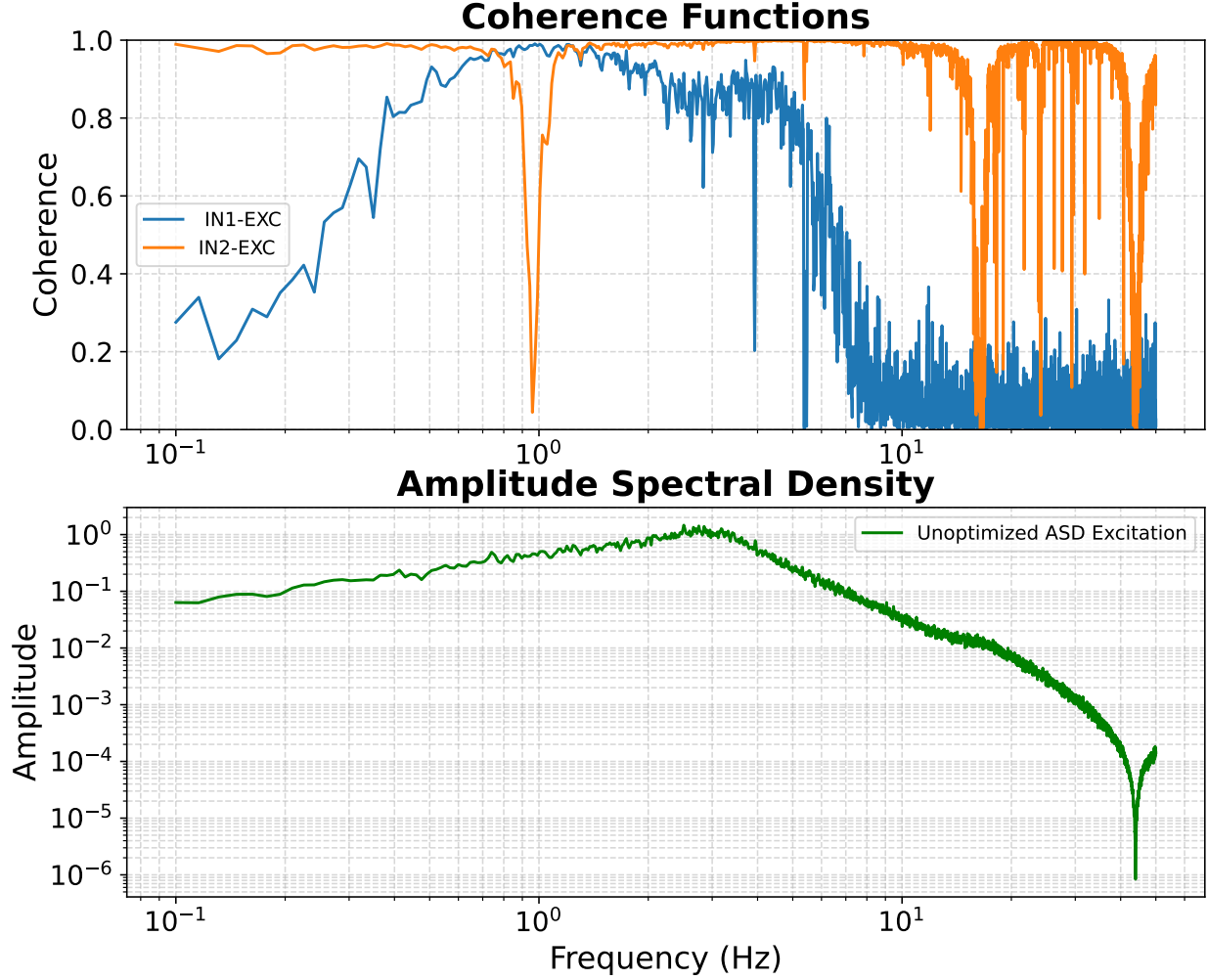


Figure 6: Illustration of how changing the excitation filter improves coherence, here we see that, coherence is below 0.7 between 0.1Hz and 0.4 Hz, in [Figure 7](#) increasing the filter is modified to increase amplitude in that band and therefore coherence increases.

from the Analytical model(as shown in [Figure 9](#)) and Simscape Multibody model with an appropriate scaling factor (to account for calibration errors).

7 Modelling Details

7.1 Overview of Simscape Multibody

Simscape Multibody (*R2025b*) is a GUI based Multibodyphysics simulation software built on top of Matlab that derives and solves the differential algebraic equations of motions of the model of a system, defined through its interface automatically[21][22]. There are two interfaces through which we interact with Simscape Multibody. A *Model Builder* interface where we connect together icons (called as *Blocks*) as shown in [Figure 17](#). These icons and the connections defined between them are then converted into a visualisation of the system and its dynamics in the *Multibody Explorer* window. Different classes of *Blocks*, represent a

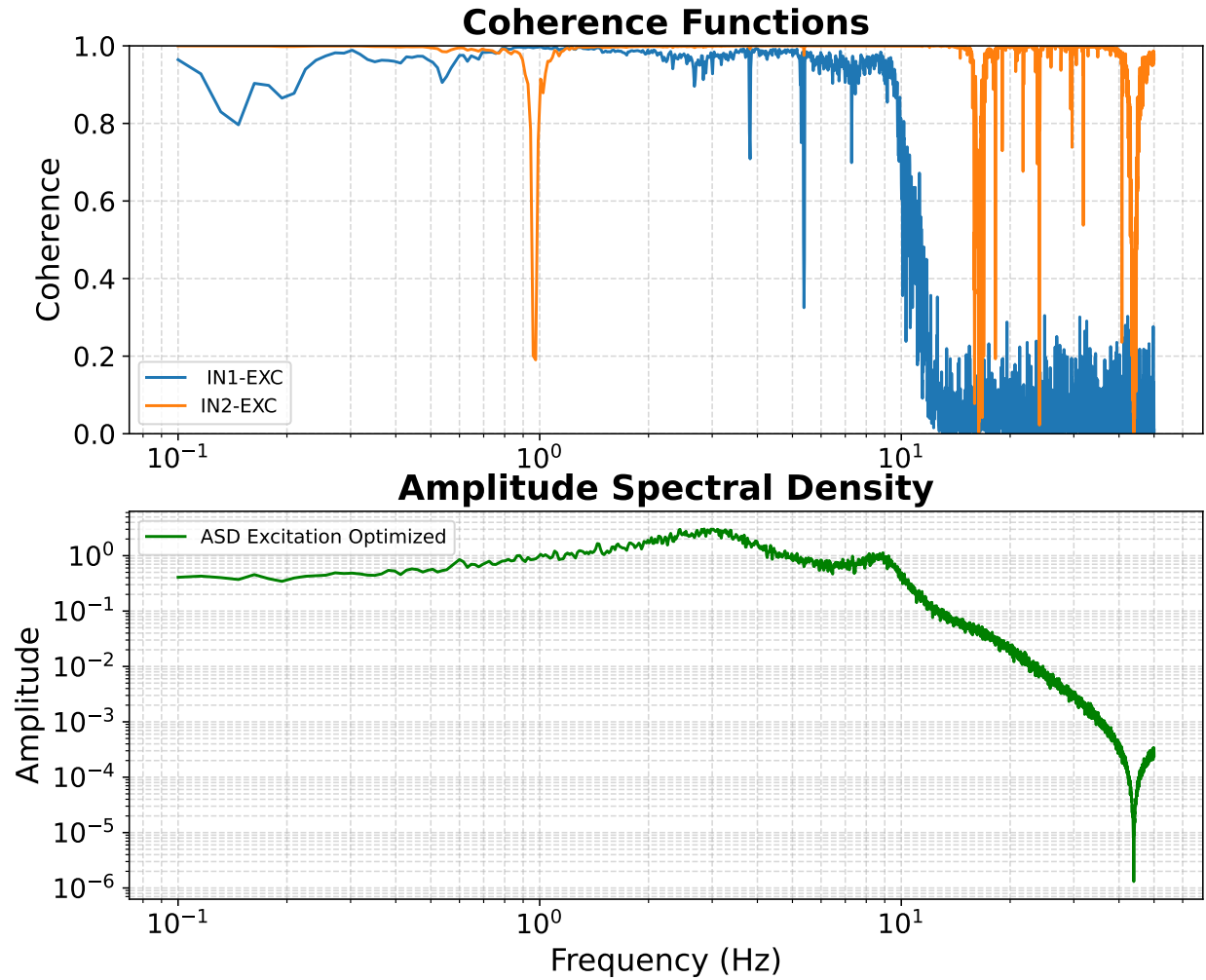




Figure 7: Illustration of how changing the excitation filter improves coherence, here we see that, coherence is above 0.7 between 0.1Hz and 0.4 Hz, as the filter was modified to increase amplitude in that band and therefore coherence increases.

different aspect of the system we want to model. The roles of the various classes of *Blocks* in the modelling workflow is listed below :-

- There are three blocks, present by default in a new *Model Builder* window they are :-

-  **Mechanism Configuration Block**, which in our case, is used to specify the direction and magnitude of gravity. It is connected to the rest of the system using the *C* port seen in the icon.
-  **Solver Configuration Block**, which specifies the parameters the solver should adopt for solving the differential equations of motion numerically.

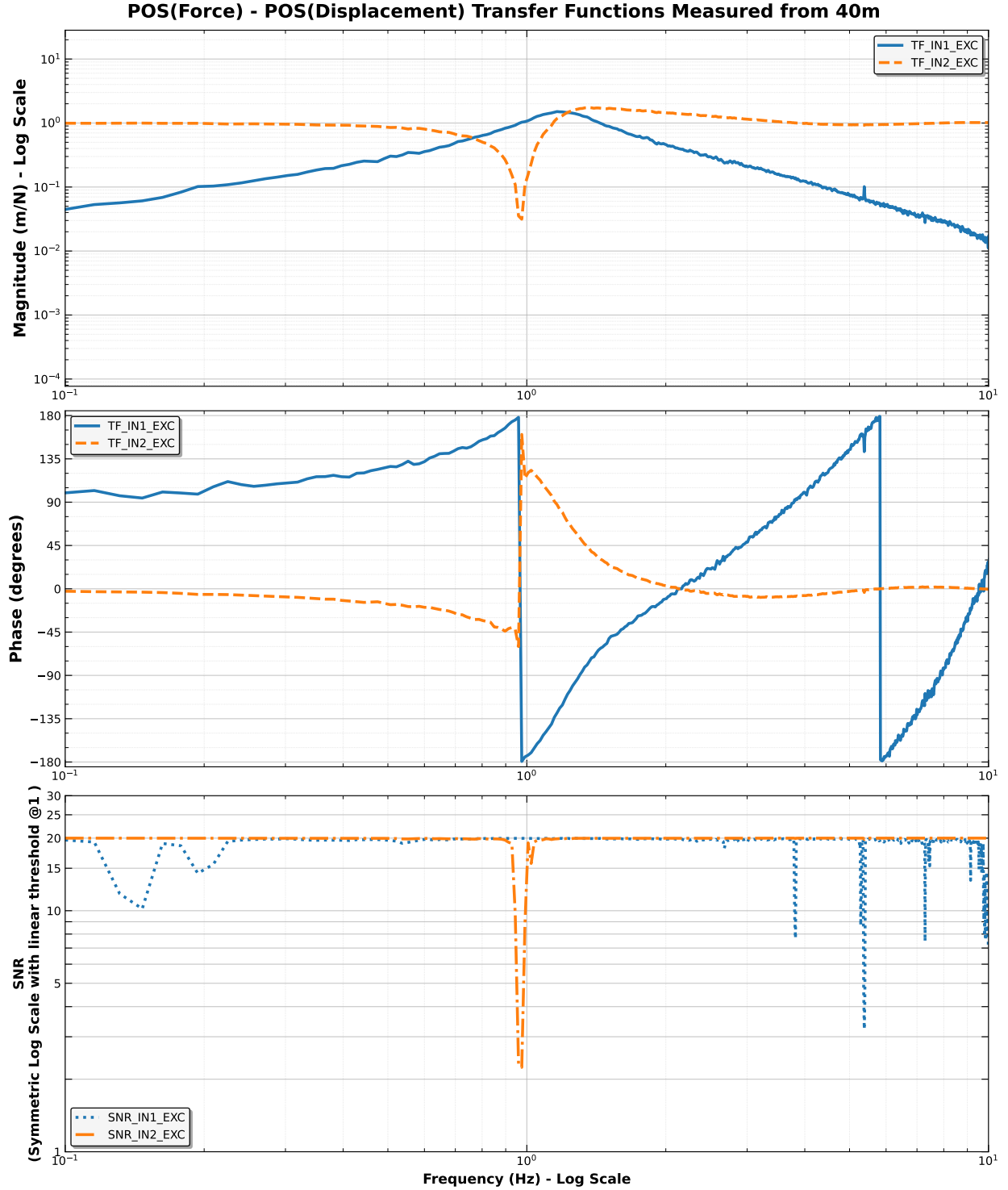


Figure 8: Bode plot of transfer function data between (IN1(POS) and EXC(POS)) as well as (IN2(POS) and EXC(POS)) along with their signal to noise ratio. These are transfer functions, that can be measured directly from the active control loop. These transfer function data are then processed according to Equation 14 to obtain the open-loop transfer function that can be compared against the transfer function from Analytical and Simscape Model.

Comparison of POS(Force) - POS(Displacement) Transfer function obtained experimentally vs analytically

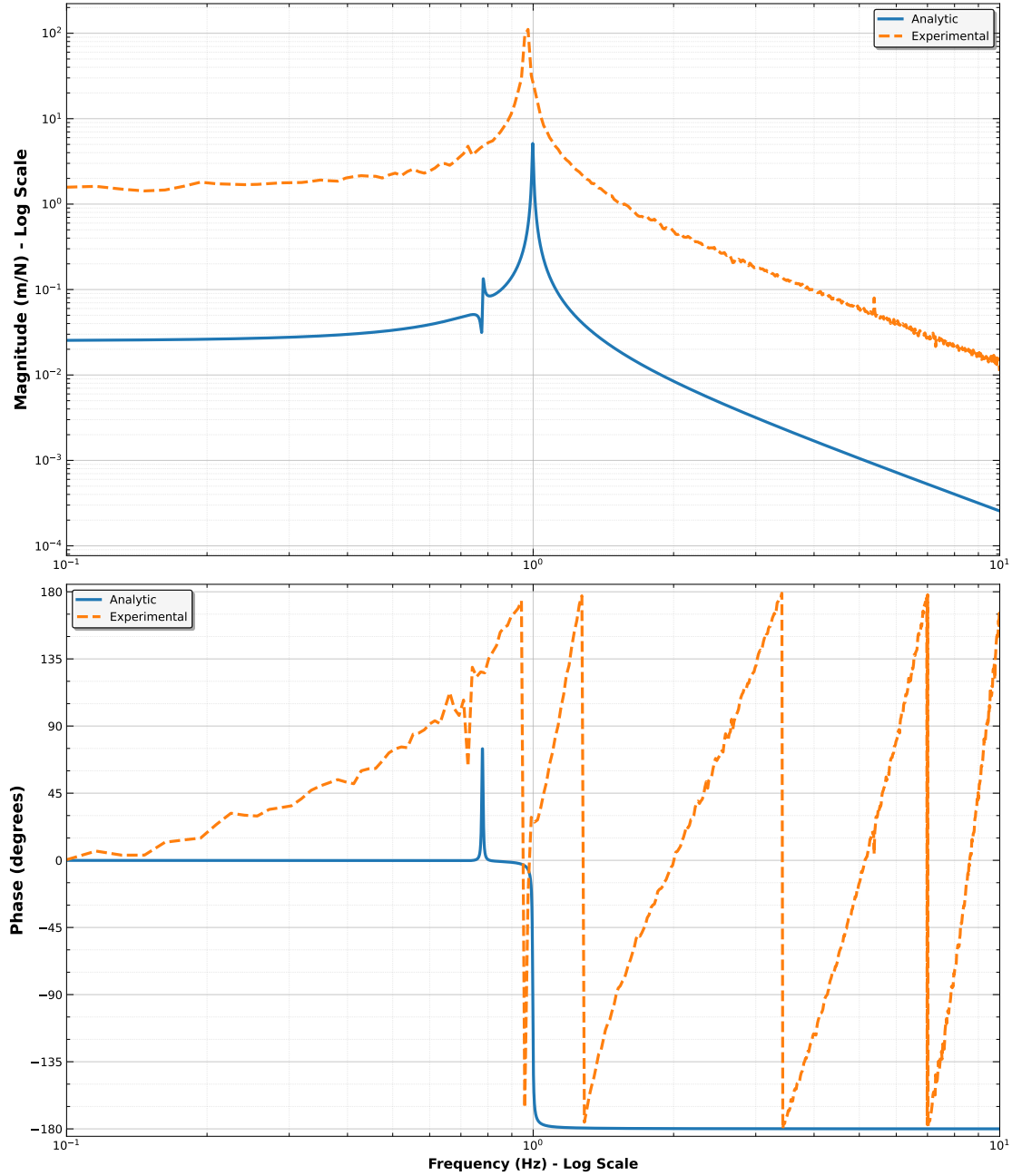


Figure 9: Bode Plots comparing the open loop transfer function (obtained by processing the transfer function data collected from 40m as per [Equation 14](#)) and the analytical transfer function (obtained through procedure detailed in [subsection 3.4](#))



World Frame Block, with its W port in the *Model Builder* interface creates a global origin *Frame* in the *Multibody Explorer* interface, with respect to which other *Frames* are positioned and oriented. A *Frame* is made up of a triad of axes orthogonal to each other, (x,y,z axes) beginning from a common origin. *Frames* are used to specify, points on the parts of a system, where they

are connected to each other or where inputs/outputs are applied/collected from the system.

- **Solid Blocks** represent different parts of a system in our case, the steel wire and the

mirrors are represented using **Cylindrical Solid Block**



The R port shown in the icon of **Cylindrical Solid Block**, corresponds to a *Frame* rigidly attached to the cylindrical body(also called *Local Frames* in the *Multibody Explorer* interface. Displacing and re-orienting the *Local Frame*, will cause the solid body itself to get reoriented and displaced. When **Solid Blocks** are visualized in *Multibody Explorer*, their centre of mass, is placed at the origin of the frame corresponding to R port of the **Solid Blocks** in the *Model Builder* interface. Further, every **Solid Block** in Simscape Multibody is treated as an extrusion of a 2-D body. For instance, **Cylindrical Solid Block** is created by extruding a circle. These extrusions, are always carried out along the z -axis of the frame corresponding to the R port of the **Solid Blocks**. We can specify the *geometry* and *inertia* parameters of the **Solid Blocks** to match the dimensions and mass of the wires and mirror of the SOS system we are trying to model.




- **Flexible Cylindrical Beam** would have been a better choice to represent the steel wires employed in the suspension. Its two ports A and B represent frames at the end of the cylinder and in addition to *geometry* and *inertia* parameters we can also specify the *stiffness*, *damping* and *discretization* as parameters of **Flexible Cylindrical Beam**. These parameters, would have enabled us to better define the steel wires used in the suspension. However, the reason for not using **Flexible Cylindrical Beam** is discussed in the next subsection.






- **Rigid Transform Block**, is connected between two *ports* in *Model Builder* interface so as to displace and reorient the corresponding *Frames* they represent in *Multibody Explorer* interface. We can understand the use of the **Rigid Transform Block** in the modelling workflow with the following example. It has two ports B and F . When the W port of the **World Frame Block** is connected to B port of **Rigid Transform Block**, it creates an exact copy of *World Frame*. This copy can then be displaced and re-oriented using the values we specify in the *Translation* and *Rotation* parameters of the *Block Parameters* dialogue box (box that open by double clicking the icon of **Rigid Transform Block**). This displaced and re-oriented *Frame* is now represented by the F port. If another block, such as the R port of **Solid Blocks** is now connected to the F port. The R port will represent a *Frame* which is an exact copy of the frame represented by F port. Now, by modifying the *Translation* and *Rotation* parameters in the *Block Parameters* box of the **Rigid Transform Block**, we will be able to displace and re-orient the **Solid Block** itself as R port is a local frame. **Rigid Transform Block** can not only be placed between the *World Frame* and an R port of a local body. They can also be placed between two R ports of two **Solid Block** to orient one with respect to another. When two *Frames* are connected using rigid transforms, their position and orientation with respect to each other is constrained so as to remain constant throughout the simulation time. Any motion of the system in

the simulation, that requires this fixed relation to be broken, will not be permitted by Simscape Multibody software. If conflicting requirements of motion and constancy between two frames arises, Simscape Multibody will not simulate the system and will output a *Rigidity Cycle* error.

- If motion is desired between two *Frames* during simulation/assembly, then they can be connected using **Joint Blocks**, In our model, we make use of three **Joint Blocks** :-

-  The **Weld Joint** which provides, zero degrees of freedom of rotation and translation.
-  The **Revolute Joint** which provides, 1 rotational degrees of freedom about the common z axis.
-  The **Spherical Joint** which provides 3 degrees of rotational freedom about the common origin.

-  **External Force and Torque Block** and the  **Transform Sensor Block** represent the various actuation inputs and sensing output that can applied to/obtained from these parts respectively. In our case, the inputs are the forces/torques along $(F_x(s), F_\theta(s), F_\phi(s))$, while the outputs are the displacements $(X(s), \theta(s), \phi(s))$. These inputs and outputs are interfaced with Simulink using **PS-Simulink** convertor blocks  so as to compute transfer function data from the model (see [subsection 7.3](#)).

The various blocks, discussed above are connected together based on the schematic from **LIGO-T000134** [18] shown in [Figure 10](#) to create a Simscape Multibody/Matlab model of the SOS suspension. The model is created using simplifications of the physical SOS system, leading to multiple possible configurations based on the simplifications adopted (see [Figure 11](#), [Figure 13](#), [Figure 15](#)). The most accurate model amongst these configurations, can be determined by comparing the transfer function data obtained from each of these configurations against the transfer function data obtained from the analytical model for a specific input/output co-ordinate. The following subsection discusses the specific simplifications chosen and their resulting configurations.

7.2 Simscape Multibody/Matlab Model Configurations

- In reality, a single steel wire is rigidly attached to a suspension point, looped tightly around the mirror, and then rigidly attached to a second suspension point offset from the first by a distance $2 \cdot R_1$. However, in all the simplified configurations discussed below, this single looped wire is represented as two separate wires. One end of each

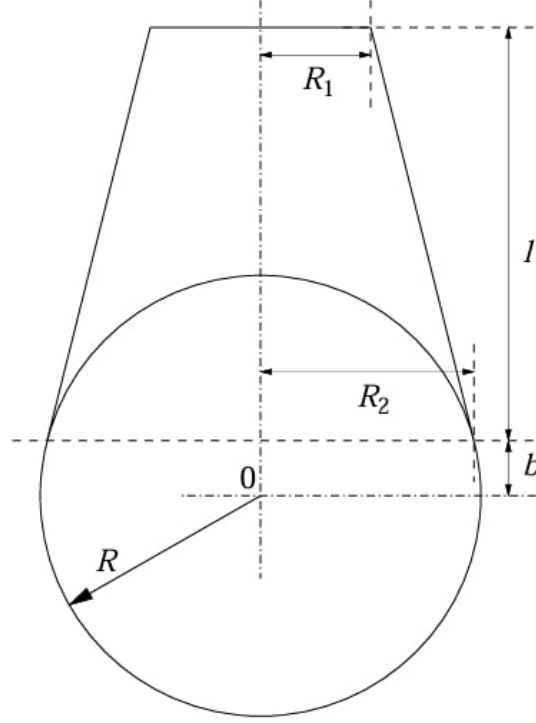


Figure 10: Schematic of SOS from T000134 [18]

wire is attached to one of two suspension points separated by $2 \cdot R_1$, while their other ends are attached to two points on the curved face of the mirror, directly opposite each other.

- In reality, the wire is rigidly attached at all the attachment points making a small angle $\arctan((R_2 - R_1)/l)$ from the normal at the attachment point as seen in [Figure 10](#). All the degrees of freedom of motion of the suspension system, are conferred by the flexibility of the wire itself. However, the various Simscape Multibody/Matlab model configurations, listed below tries to represent the same dynamics through various simplifications. Transfer function data from each of these model configurations for a specific input/output combination(POS(Force)-POS(Displacement)) is compared against transfer function data obtained from the analytical model to determine the most accurate model configuration.

- **Flexible Wire; No Joints Configuration :** The best way to represent the real life assembly of SOS (detailed above) in the model is as follows. Create a suspension point represented by the **Weld Joint** at a distance R_1 (specified by connecting a **Rigid Transform Block**) from the **World Frame**. Connecting another **Rigid Transform Block** to the F port of **Weld Joint**, we can create a new frame, that is tilted at the small angle $\arctan((R_2 - R_1)/l)$. One end of one of the wires represented by the B port of **Flexible Cylindrical Body** can be attached to this frame. If the same logic, is followed for modelling all the four attachment points, we can represent the complete system accurately. [Figure 11](#) shows a Simscape Multibody/Matlab model created in this manner and [Figure 12](#)

compares the transfer function data obtained from this model against the transfer function data obtained from the Analytical model. We notice immediately from Figure 12 that the resonant frequency and phase shift computed by MATLAB are off by 1 Hz compared to the analytical model's predictions. This difference can perhaps be resolved by increasing the number of elements, the **Flexible Cylindrical Body** is *discretized* into. However, when **Flexible Cylindrical Body** is *discretized* into more than 1 element, Matlab returns the following error while trying to generate transfer function data using *linmod* function in Matlab **Error using dlinmod ['No_Joints/Steel Wire L']: Body 'No_Joints/Steel Wire L' caused a simulation error. This might be caused by extreme values in its geometry, material properties, mass and stiffness properties, or deformation. For a flexible beam, this could also be caused by numerical ill-conditioning if the cross-sectional properties were calculated from geometry.**

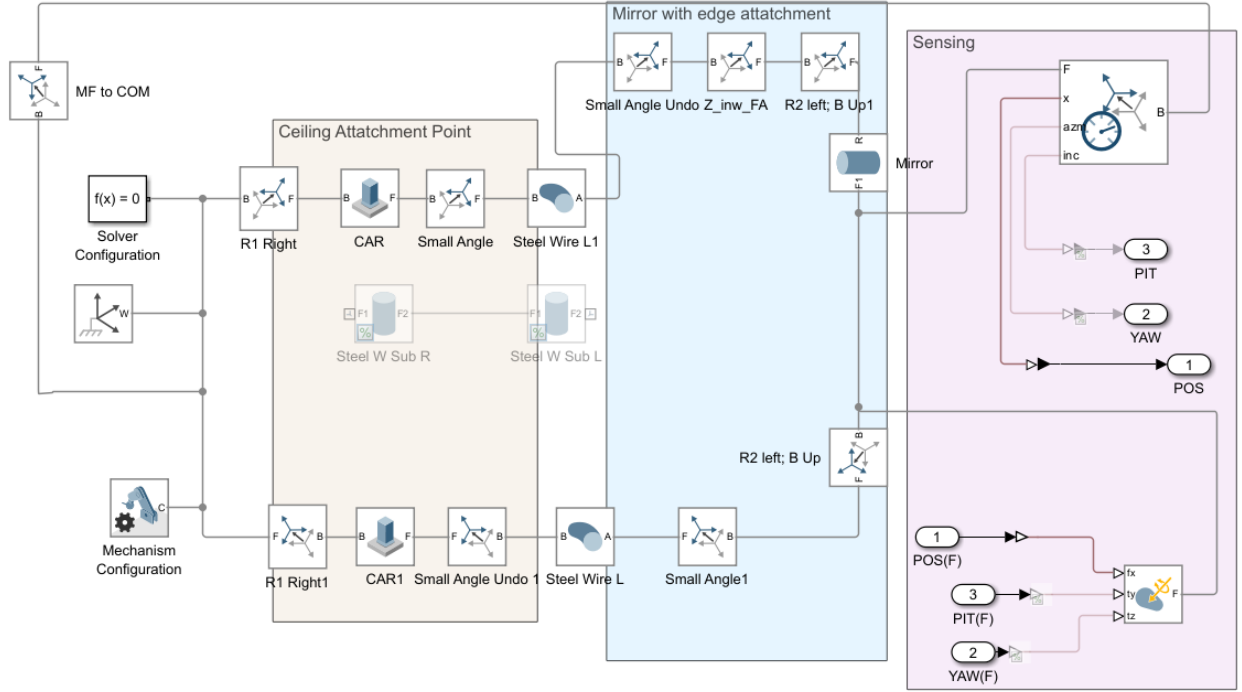


Figure 11: Screenshot of Simscape Multibody/Matlab model that uses only flexible cylindrical solid and angles defined by rigid transformation to represent the way the wires are suspended and attached to the mirror. This configuration, avoids the usage of any joints. The comparison of POS(F) - POS(Displacement) transfer function data obtained from this model against the transfer function data obtained from the analytical model for is made in Figure 12

- **Flexible Wire; 4 Revolute Joints Configuration** : As we saw in the previous paragraph, the model that represent the SOS system in the most ideal way miscalculates the resonance frequency when compared to the Analytical model. Further, updating such a model would be quite difficult since the angle mentioned is only correct for the idealised assembly of the suspension. Since we are interested in updating the model based on experimental data so as to capture

TF data comparison between Analytical and Matlab Model for (POS(Force)/POS(Displacement)-i/o)

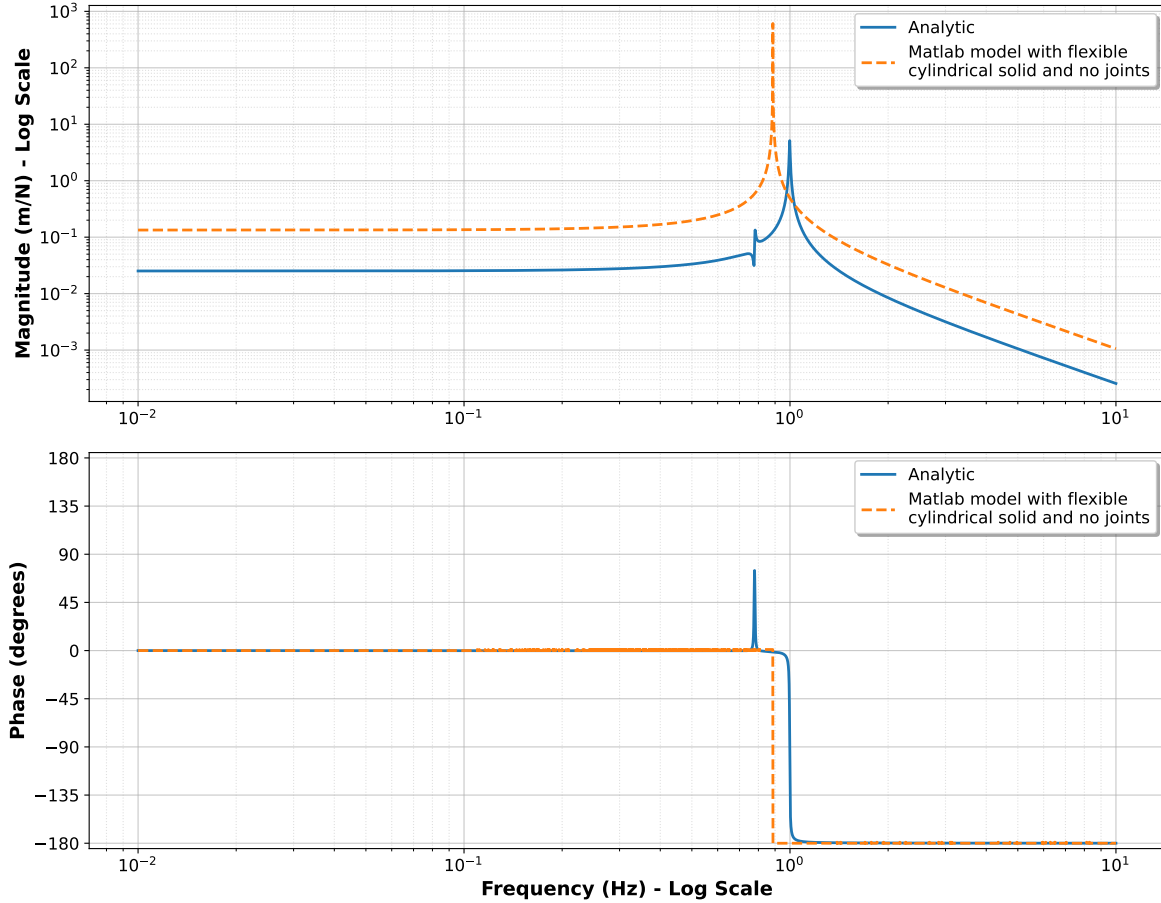


Figure 12: Bode plot comparison of POS(Force) - POS(Displacement) transfer function data obtained from the Analytical model against the transfer function data obtained from the Matlab model (shown in Figure 11) that represents the wire and attachments in SOS system using flexible cylindrical solid and Rigid Transforms only.

the deviations from the idealised assembly it is necessary that Simscape Multibody automatically computes the angle of attachment necessary for any arbitrary point of attachment we specify. In order to do the same, instead of using **Rigid Transform Block** to specify the angle, we use a joint with 1 degree of freedom of rotation (i.e the **Revolute Joint**) to rotate about the common z axis and automatically align the attached frames (the frames at end of the wire and various attachment points). [Figure 13](#) shows a Simscape Multibody/Matlab model created in this manner and [Figure 14](#) compares the transfer function data obtained from this model against the transfer function data obtained from the Analytical model. The two transfer functions, vary markedly from one another. This variation is perhaps caused as **Revolute Joints** restricts, the freedom of movement of the flexible wire. We can ease the restrictions imposed by the **Revolute Joints**, by replacing it with a **Spherical Joints**. However, Matlab does not allow a configuration with all four **Spherical Joints**, replaced by **Revolute Joints** to be assembled and returns a **Implicit 6 degree of Freedom error**. Therefore, we replace only two of the **Revolute Joints** with **Spherical Joints**.

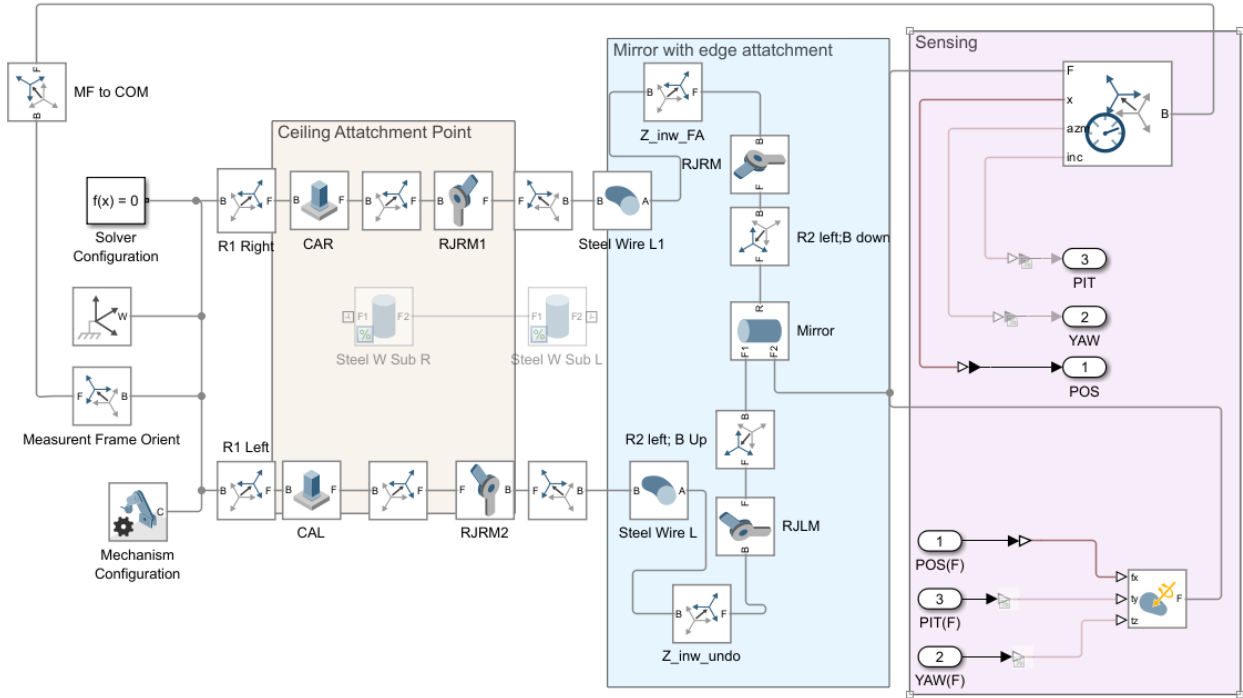


Figure 13: Screenshot of Simscape Multibody/Matlab model that uses flexible cylindrical beam and four revolute joints to represent the way the wires are suspended and attached to the mirror. This configuration, autocalculates the angle at which the attachments to the mirror and suspension point must be made based on the position we specify for the attachments. The comparison of POS(F) - POS(Displacement) transfer function data obtained from this model against the transfer function data obtained from the analytical model for is made in [Figure 14](#)

- **2 Spherical Joints, 2 Revolute Joints Configurations** : First we tried to implement the 2 Spherical Joints and 2 Revolute Joints configuration with the

TF data comparison between Analytical and Matlab Model for (POS(Force)/POS(Displacement)-i/o)

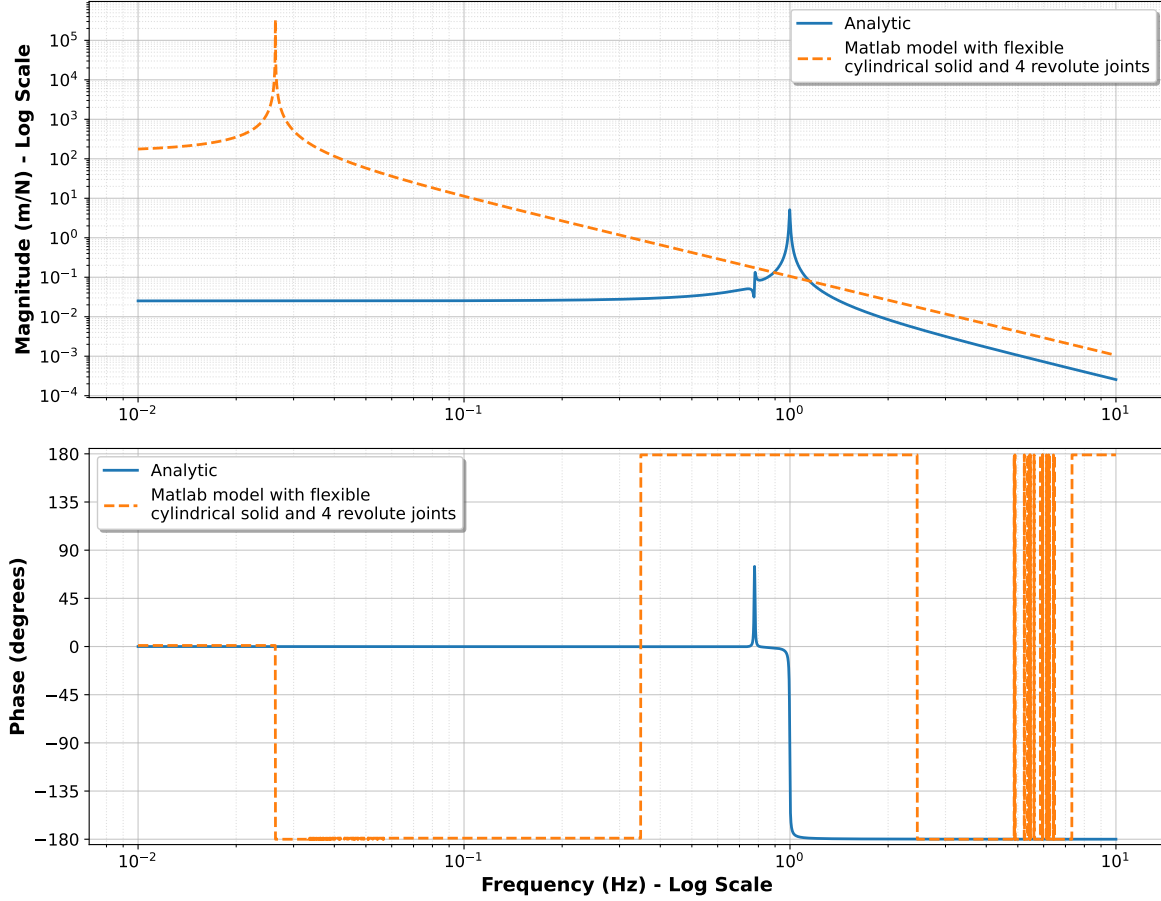


Figure 14: Bode plot comparison of POS(Force) - POS(Displacement) transfer function data obtained from the Analytical model against the transfer function data obtained from the Matlab model (shown in Figure 13) that represents the wire and attachments in SOS system using flexible cylindrical solid beam and four revolute joints.

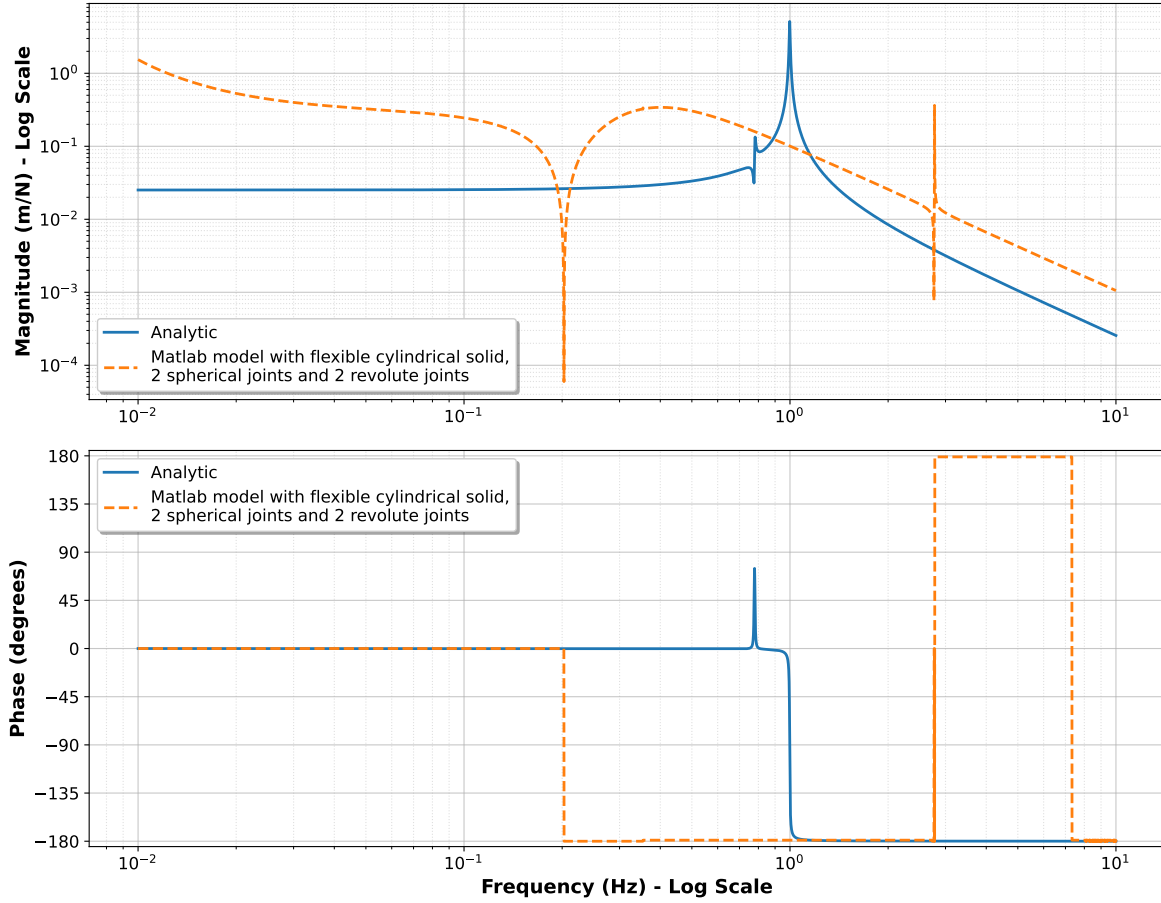
TF data comparison between Analytical and Matlab Model for (POS(Force)/POS(Displacement)-i/o)

Figure 16: Bode plot comparison of POS(Force) - POS(Displacement) transfer function data obtained from the Analytical model against the transfer function data obtained from the Matlab model (shown in [Figure 15](#)) that represents the wire and attachments in SOS system using flexible cylindrical beam, 2 spherical joints and 2 revolute joints.

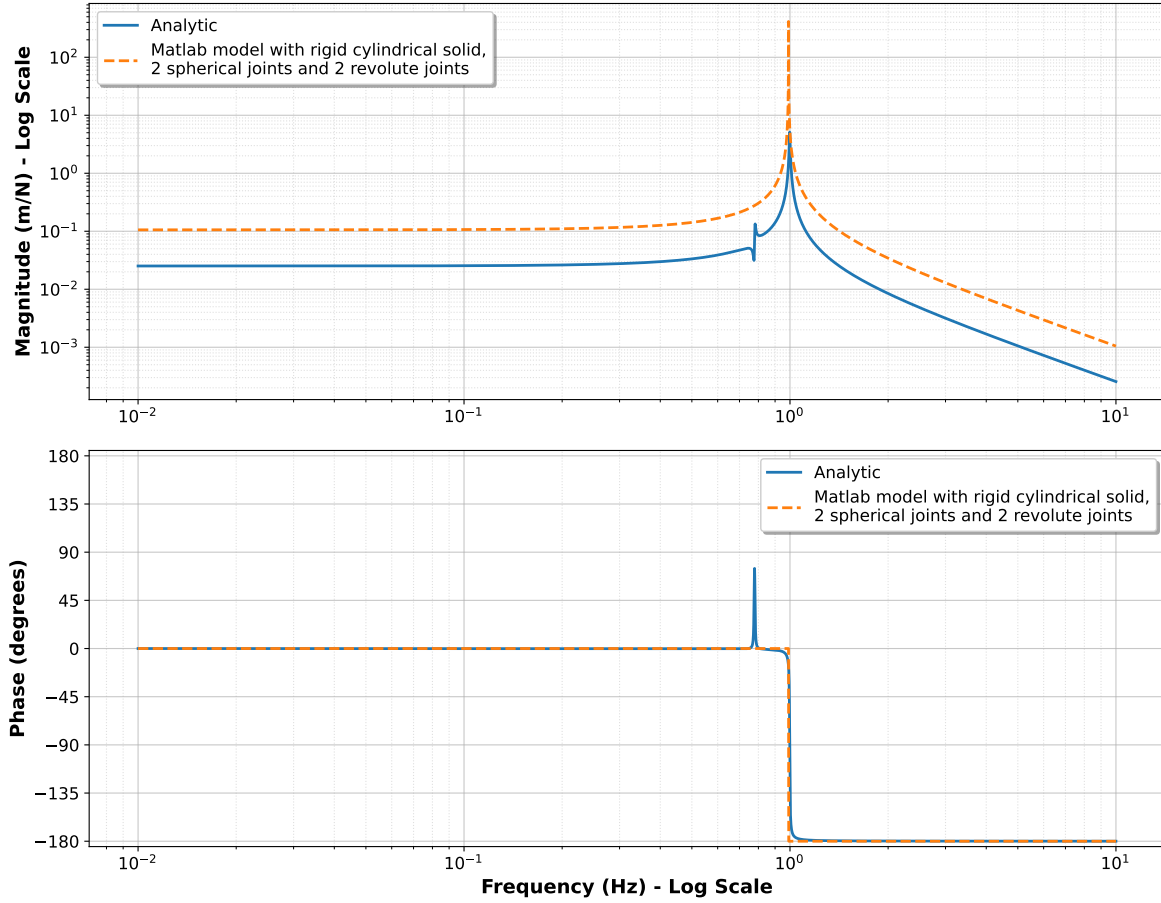
TF data comparison between Analytical and Matlab Model for (POS(Force)/POS(Displacement)-i/o)

Figure 18: Bode plot comparison of POS(Force) - POS(Displacement) transfer function data obtained from the Analytical model against the transfer function data obtained from the Matlab model (shown in [Figure 17](#)) that represents the wire and attachments in SOS system using cylindrical solid block, 2 spherical joints and 2 revolute joints.



7.3 State-Space Representation and Function Data andfrom Simscape

A linear time-invariant (LTI) system can be represented in the time-domain using the following equations: This representation is known as a State-Space representation[23].

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (18)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (19)$$

Where, the terms have the following meaning:-

- $\mathbf{x}(t)$: State vector; In simscape, they are represented by the position of each joint in the system
- $\dot{\mathbf{x}}(t)$: Time derivative of State Vector; In simscape they are represented by the velocity of each joint in the system.
- $\mathbf{u}(t)$: Input vector; In simscape, they are specified using the **inport** block. 
- $\mathbf{y}(t)$: output vector; In simscape, they are specified using the **outport** block. 

When these four vectors are defined in Simscape, we can use **linmod** and **ss** to generate the **ABCD** matrix corresponding an LTI system. However real systems are never LTI, yet they can be approximated to an LTI system around an operating point. To find operating point in Matlab, we let the simulation run with simulation time set to zero. The **Time, States, and Output** check boxes are selected in the **Data Import/Export** pane of model Configuration Parameters settings. This creates an operating point object which can be used by **linmod** [6]. The following code snippet is then used in Matlab to create a frequency domain representation of the same information in the form of a bode plot.

```
mod=linmod( ' Assembled ' );
sys=ss(mod.a,mod.b,mod.c,mod.d);
bode(sys,plotoptions);
```

8 Attempt at Automating Excitation Optimisation.

In the final part of this project, I modified a python script written by Hang Yu and my mentor Rana X Adhikari. The original script, demonstrates how optimisation of excitation needed to obtain accurate transfer function data of a system(see [subsection 6.1](#)) can be automated based on Pintelon and Shoukens approach[24]. The original script is structured in the following way :-

- The parameters characterizing the true system and the approximate model of the system is defined
- The noise in the channels from which signals need to be readout(see [subsection 3.5](#)) for obtaining transfer function data is modelled.
- The root mean square value of noise is computed from the modelled Power Spectral Density of noise.
- The excitation signal is then designed as a white spectrum with uniform power distribution across all frequency bins. The total excitation power is set to twice the measured noise power to ensure adequate signal to noise ratio during measurement.
- Transfer function measurement using the excitation signal designed earlier is simulated.
- The uncertainty/error bounds in transfer function measurement is estimated using coherence
- Weighted fitting is performed on the simulated transfer function measurement data to estimate most likely parameters .
- Fisher information matrix is calculated for the simulated transfer function measurement data.
- The approximate model is updated based on the most likely parameters estimated.
- The excitation is updated based on the most likely parameters estimated.
- These steps are iterated to improve both the parameters of the model characterizing the system as well as the excitation used for obtaining transfer function data.

In the modified script, the parameters characterizing the approximate model is obtained from the Simscape Multibody Model. The noise and transfer function measurements data is collected experimentally from the 40m using *cdsutil*[16] and *dtf*[17] packages in python. However the steps discussed earlier could not be iterated more than once as python returns a **LinAlgError: Singular matrix** while trying to compute Fisher information.

9 Conclusion and Recommendation for Future Work

The default solver configuration in Simscape Multibody/ Matlab is not well suited to model a very thin flexible wire resulting in the error message we saw for the first configuration (**Flexible Wire; 4 Revolute Joints Configuration**) discussed in [subsection 7.2](#). In future rectifying this error by changing the solver configurations may allow us to obtain a more accurate model. The final Matlab model configuration we settled upon (**2 Spherical Joints, 2 Revolute Joints and Cylindrical Solid Block**), computed the resonance frequency and phase shift frequency accurately when compared against the analytical model and also has the advantage of being able to autocalculate the attachment angle for any arbitrary attachment point. This attribute, should enable us to capture one aspect of deviations in the real system from the idealised assembly(i.e. slightly deviated attachment points). However the magnitude computed by this Matlab model configuration does not agree with the Analytical model. This model could not be used for automating the calculation of optimum excitation as discussed in [section 8](#). Further work needs to be carried out to determine and rectify the cause of the errors encountered to make progress in the attempt at making a digital twin.

References

- [1] Abbott, B P et al. *LIGO: The Laser Interferometer Gravitational-Wave Observatory*. Reports on Progress in Physics, vol. 72, no. 7, p. 076901(30 June 2009). <https://doi.org/10.1088/0034-4885/72/7/076901>
- [2] Reid, S., et al. *Mechanical Dissipation in Silicon Flexures*. Physics Letters A, vol. 351, no. 4-5, , pp. 205–211, (Mar. 2006) <https://doi.org/10.1016/j.physleta.2005.10.103>
- [3] Callen, Herbert B., and Theodore A. Welton. *Irreversibility and Generalized Noise* Physical Review, vol. 83, no. 1, pp. 34–40, (1 July 1951) <https://doi.org/10.1103/physrev.83.34>
- [4] Abbott, B. P., et al. *Observation of Gravitational Waves from a Binary Black Hole Merger* Physical Review Letters, vol. 116, no. 6, pp. 061102, (11 February 2016) <https://doi.org/10.1103/physrevlett.116.061102>
- [5] Abbott, B. P., et al. *GW151226: Observation of Gravitational Waves from a 22-Solar-Mass Binary Black Hole Coalescence* Physical Review Letters, vol. 116, no. 24, pp. 241103, (15 June 2016) <https://doi.org/10.1103/physrevlett.116.241103>
- [6] Matlab Operating Point Documentation <https://www.mathworks.com/help/simscape/ug/finding-an-operating-point.html#busm5u4-8>
- [7] Adhikari, Rana X, et al. *A Cryogenic Silicon Interferometer for Gravitational-Wave Detection*. Classical and Quantum Gravity, vol. 37, no. 16, 30 Jan. 2020, pp. 165003–165003, <https://doi.org/10.1088/1361-6382/ab9143>
- [8] Post-O5 Study Group *Report from the LSC Post-O5 Study Group* LIGO technical report T2200287-v3, https://dcc.ligo.org/public/0183/T2200287/003/T2200287v3_P05report.pdf
- [9] Seiji Kawamura and Janeen Hazel *Small Optics Suspension Final Design (Mechanical System)* LIGO technical report LIGO Document T970135-02 <https://dcc.ligo.org/LIGO-T970135/public>
- [10] Daryush J. Dawid; Seiji Kawamura *Investigation of violin mode Q for wires of various materials* Rev. Sci. Instrum. 68, 4600–4603 (1997) <https://doi.org/10.1063/1.1148439>
- [11] Daniel Sigg, Peter Fritschel *Diagnostics Test Software* LIGO technical report T990013-B <https://dcc.ligo.org/LIGO-T990013-x0/public>
- [12] James Batch *Diaggui (DTT) User Interface Reference* LIGO technical report T1700118-v1 <https://dcc.ligo.org/LIGO-T1700118>
- [13] Antonio Mauricio F.L , Miranda de Sá *A note on the coherence-based signal-to-noise ratio estimation in systems with periodic inputs* Journal of the Franklin Institute, Volume 343, Issue 7, November 2006, Pages 688-698 <https://doi.org/10.1016/j.jfranklin.2006.05.002>

- [14] Brett Shapiro *Electronic Setup and Testing of Advanced LIGO Suspensions* LIGO document E1000078-v1 <https://dcc.ligo.org/LIGO-E1000078/public>
- [15] *Github repository of dtt* <https://git.ligo.org/cds/software/dtt>
- [16] *Github repository of cdsutisl* <https://git.ligo.org/cds/software/cdsutils>
- [17] *Github repository of dttxml* <https://git.ligo.org/cds/software/dttxml>
- [18] M. Rakhmanov, D. Reitze, D. Tanner, S. Yoshida and H. Yamamoto *Dynamical Properties of LIGO Single Loop Suspended Mirrors* LIGO technical report T000134 <https://dcc.ligo.org/LIGO-T000134/public>
- [19] Strain, K. A., and B. N. Shapiro. *Damping and Local Control of Mirror Suspensions for Laser Interferometric Gravitational Wave Detectors*. Review of Scientific Instruments, vol. 83, no. 4, 1 Apr. 2012. <https://doi.org/10.1063/1.4704459>
- [20] *Multi-Color Interferometry for Lock Acquisition of Laser Interferometric Gravitational-wave Detectors* [Thesis] - Department of Astronomy - The University of Tokyo <https://dcc.ligo.org/LIGO-P1300001/public>
- [21] The MathWorks, Inc. *Simscape Multibody Documentation R2025b* MATLAB Help Centre https://www.mathworks.com/help/pdf_doc/sm/index.html
- [22] Sider, A. *TOWARDS LOW FREQUENCY SEISMIC ISOLATION OF LARGE CRYOGENIC MIRROR* [Doctoral thesis] - University of Liège]. ORBi-University of Liège. ; 2024 <https://orbi.uliege.be/handle/2268/323741>
- [23] Karl Johan Åström, and Richard M Murray. *Feedback Systems - An Introduction for Scientists and Engineers* Princeton University Press, 22 Feb. 2009. cds.caltech.edu/~murray/books/AM08/pdf/am08-complete_22Feb09.pdf
- [24] Rik Pintelon, and Johan Schoukens. *System Identification : A Frequency Domain Approach; Chapter 5.4* Hoboken, N.J., John Wiley and Sons Inc., Cop, 2012,