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

Technical Note	LIGO-T1500228-v1	2015/09/29

Final Report: Sensing and Control of Suspended Optics Breadboard in the Crackle2 Experiment

Sai Kanth Dacha | Mentors: Gabriele Vajente, Rana Adhikari

Distribution of this document:

LIGO Scientific Collaboration

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 NW17-161
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/

Contents

Abstract

Crackling noise arises when a system responds to changing external conditions through discrete, impulsive releases of energy. Various locations in the Advanced LIGO setup have been identified over the years as suspected to produce Crackling noise. Of specific interest is the Crackling noise arising in maraging steel blades that are used to suspend the Interferometer end mirrors in a quadruple pendulum stage, where vertical displacement noise due to Crackle events couples to horizontal displacement noise along the arm of the Interferometer.

In order to study this noise experimentally, a setup was designed a couple of years ago: a Michelson Interferometer configuration to measure vertical displacement noise at the tip of a maraging steel blade from which a mass is suspended. One of the major difficulties in measuring Crackling noise using this setup has been the high sensitivity required [2], and having a good seismic isolation system with a good control over the motion of the interferometer is essential.

This paper focuses on my work towards establishing a feedback damping control loop around the suspended optics breadboard which houses the interferometer. I describe the whole process of the control system design, starting from sensing and actuation to compensator design. I also discuss certain miscellaneous things I have worked on to improve speed and efficiency for further rounds of implementing the control system, upon any possible modifications to the setup. I conclude by talking about the improvement in seismic isolation this control system has brought about.

1 Introduction

1.1 What is Crackling Noise?

James P. Sethna et. al. define Crackling noise as follows: ”Crackling noise arises when a system responds to changing external conditions through discrete, impulsive events spanning a broad range of sizes”. [1] This is a very broad definition, and what we are interested in is a specific kind of Crackling noise: that arising in maraging steel blade springs under stress.

Let us try to get an idea of what Crackling noise actually is and how it occurs in the said system. Below is a simple explanation for intuitive understanding, and what follows is a more formal treatment.

1.1.1 An intuitive explanation

For the purpose of the current discussion, let us start from the definition by James et. al. quoted above. It says that a system exhibiting crackling noise responds to *changing external conditions* in the form of *impulsive releases of energy*. Focus on the italicized parts. Now, consider the following: pick up a piece of paper and start crumpling it slowly. Notice that as your hand crumples

it, you can *hear* the sound of paper being crumpled. Why does this happen? If you haven't already guessed, the answer is: Crackling Noise!

In this case, the motion of the hand (*changing external conditions*) causes the system (paper) to respond in the form of sound (*pulses of energy*). At this point, it is very important to note the following: the frequency of motion of the hand is of the order of a few hundred mHz, but the response (sound) is in the audio regime! This frequency up-conversion is typical of Crackling noise, and it shows the nonlinear behavior of dislocation interactions. (More on this can be found in a more formal treatment below.)

What the reader must take away from this explanation is the frequency up-converting trait of Crackling noise. As we shall discuss later, low-frequency seismic motion up-converts into the aLIGO detection band through Crackling mechanism, and this is detrimental for GW detection. An explanation for how exactly this happens follows a little later.

1.1.2 A more formal treatment

A more formal way of understanding this is as follows: In metals, dislocations are 'pinned' by obstacles like grain boundaries or other surfaces. Under small oscillatory stress, these dislocation lines bow in and out, but the response of the complex network of a large number of such dislocations on the whole is known to act nonlinearly through long-range interactions. This nonlinear behavior, among a broad class of other nonlinear phenomena, is known to be the cause of "crackling". "Crackling" here refers to impulsive releases of energy, acoustic emissions, or changes in the geometry of attachments between suspension elements.

In this paper, I will not digress further into the technical details of the modeling of Crackling noise and its underlying mechanisms. Reference [3] discusses mathematical modeling, the interested reader can take a look.

1.2 Why does Crackling Noise matter?

It has been suspected that this "crackling noise" in various components and suspensions might produce excess noise in aLIGO. [3]

Many possible locations of crackle have been identified, some of which are:

- The maraging steel blades used for vertical isolation
- The silica fibers which suspend the test masses from the penultimate masses
- The welds which attach the fibers to the ears
- The clamps which hold the suspension wires to the steel blades

Figure 1 shows the test mass suspension scheme in Advanced LIGO.

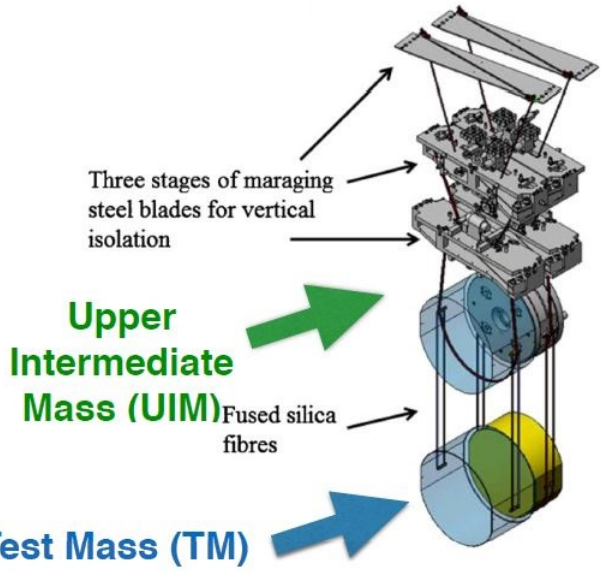


Figure 1: Suspension scheme in Advanced LIGO

The maraging steel blades have been under investigation in the first crackle experiments, as they present a mechanical system which can be driven and stressed easily. In addition, mechanical crackling noise is sufficiently generic that an experiment capable of measuring this noise in maraging steel blade springs will be well suited for investigating crackling in other components used in aLIGO as well.

As crackling noise is inherently nonlinear, there is the potential for noise to be upconverted. Specifically, motion of the suspension at microseismic frequencies may induce blade motion, causing the blades internal stresses to fluctuate, resulting in an avalanche of crackle events with high-frequency content. This results in displacement noise in the stressed (vertical) direction.

The reader might wonder why *vertical* displacement noise would matter at all: the Gravitational Wave would stretch/contract the arm in the ”horizontal” directions. The explanation is as follows: because the earth is curved over the length of each arm (4km!), the input and end mirrors are no longer parallel to each other; instead, they are perpendicular to the ground, along gravity. In this situation where the mirrors are not parallel to each other, the laser beam will not ”stay” in the cavity, and it would bounce off the mirrors a few times and exit the cavity.

But because we want the cavity to be *locked*, where the light is *stored* in the cavity due to repeated reflections off the mirror surfaces, the mirrors are *held* parallel to each other (and perpendicular to the laser beam) by external force. The configuration is shown in figure 2 below.

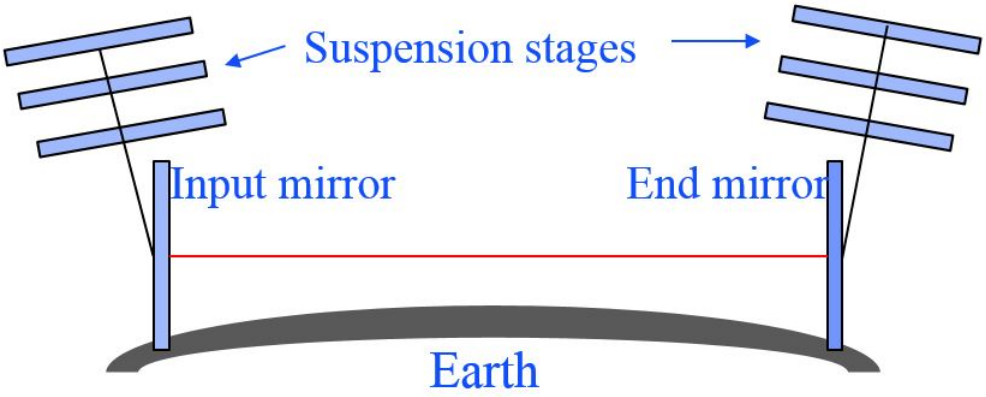


Figure 2: Alignment of input and error mirrors in each arm of the aLIGO interferometer

Now, there's a problem: "vertical" displacement noise (vertical is along gravity, not along the length of the mirror, because the mirror is now held such that it it perpendicular to the laser beam) in the end mass suspension system has a "horizontal" coupling component.

Looking back at the whole development presented above, it is clear that all of this revolves around the Crackling noise that arises in maraging steel blade springs. The motivation of the Crackle experiment is to measure this noise experimentally.

In the next section, I describe the basic measurement strategy for Crackling noise, and the setup that has been designed and being used so far. I then set the context for and describe the problem statement of my work. The sections that follow describe my work in detail.

2 Starting point for my work

This section describes the work that was already done before my arrival. I picked up one part of the Crackle2 experiment - suspension damping of the breadboard - and I started off in direction of establishing a feedback damping control loop for it.

2.1 Measurement strategy

With reference to figure 1, below the upper intermediate mass (UIM) in the quadruple suspension system, there is no more spring blade isolation, thus any crackling noise in the UIM maraging steel blades will propagate directly to the test mass. Therefore, one would want to ensure low enough crackling noise at the UIM blade tip itself.

A direct measurement of crackling noise is very difficult. However, one can make measurements of the blade displacements directly using a Michelson interferometer with end mirrors mounted to loaded blade springs which are driven with a low frequency, common-mode force. Since crackling

noise occurs incoherently in each blade, it will show up in the Michelson's displacement signal. In order to ensure repeatability and applicability of results, the setup has been made to be similar to the existing aLIGO configuration.

2.2 Experimental design

The setup consists of a Michelson interferometer using blade-suspended masses as end mirrors. A crackle event will change the differential displacement of the mirrors, and hence be reflected in the interferometer output. The events are excited by a low frequency, common-mode, drive on the two blades. [2] The apparatus is going to be housed in a vacuum chamber to mitigate acoustic noise.

Figure 3 is a picture of the full setup.

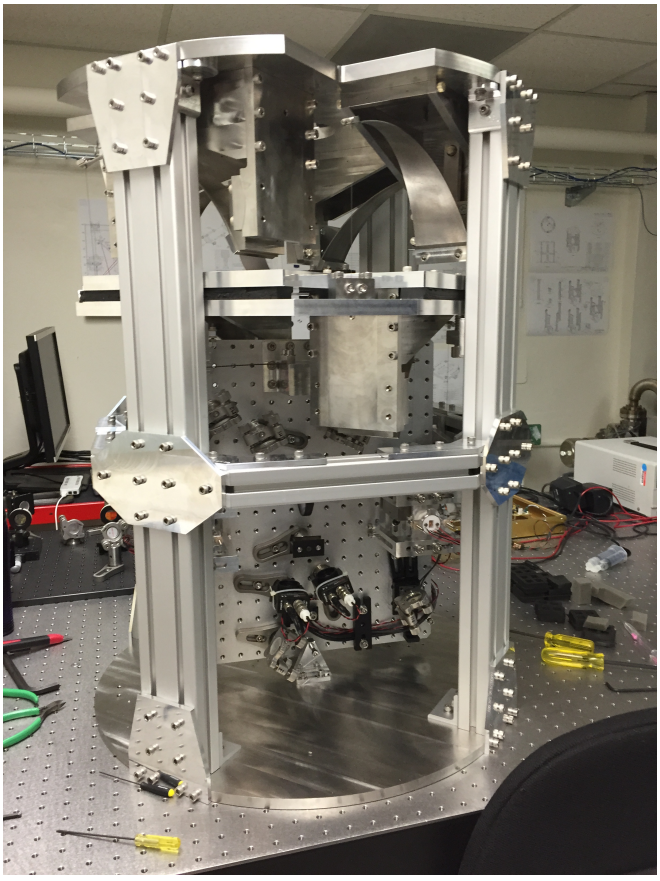


Figure 3: Complete setup

2.2.1 Optical layout

The optical layout schematic in figure 4 below.

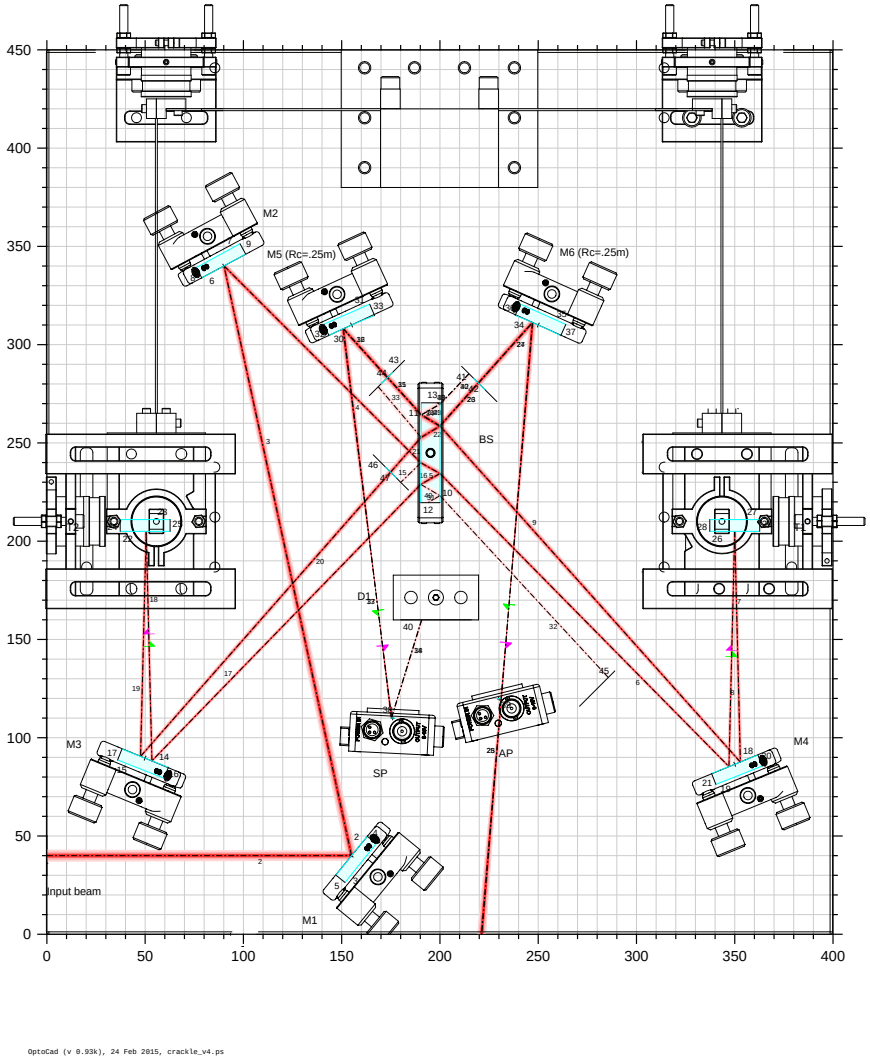


Figure 4: Optical layout

Light from the laser comes in from the bottom left corner through a viewport. Two folding mirrors then redirect the beam to the beam splitter. The two arms of the Michelson interferometer are folded in such a way that the beams impinging on the end mirrors are almost vertical, but tilted enough so that the beams propagating in opposite directions (before and after reflection from the mirror) are separate. The end mirrors of the Michelson arms are horizontal.

2.2.2 Seismic isolation system

Earlier versions of the setup used a stack of two steel plates resting on blocks made of rubber to provide isolation of the Michelson from the ground motion. Ideally, seismic motion of the optical setup wouldn't couple to the Michelson signal because the motion would be common to both mirrors. However, any differential motion of the blades would result in a spurious signal.

Lately, a suspension system has been employed for the breadboard with the Michelson interferometer. A basic control of the breadboard has already been established, and a more detailed modeling of the system and damping in all six d.o.f. is to be implemented.

Figure 5 below shows a simplified scheme of the suspension system. Vertical isolation has been achieved using maraging steel spring blades. A two stage system has been designed. The upper stage is composed of four such blades, each one supporting a wire which is attached to an intermediate stage.

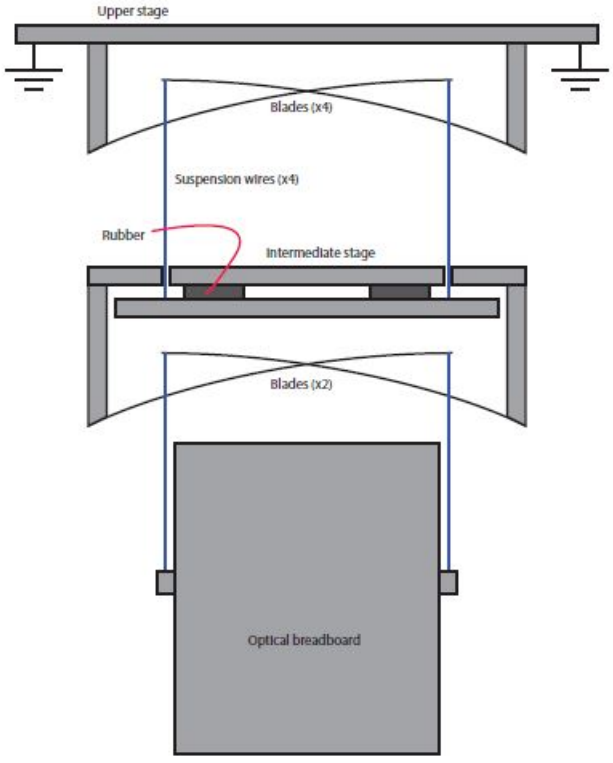


Figure 5: Scheme of the suspension system

3 Problem Statement

The breadboard shown in figure 5 above is, clearly, suspended from a two stages and is "free" to move. Even though the suspension stages provide isolation from seismic noise, the same suspen-

sion stages also cause two resonances elsewhere. This is simply because the system acts like a double spring-mass or pendulum system which has two resonant frequencies. What the values of these resonant frequencies are, in each degree of freedom, depends on the mechanical design of the whole setup. In this case, the setup has been designed such that the resonant frequencies are around 1 Hz, for all d.o.f.

As a result of these resonances, any excitation of the breadboard around those frequencies can cause severe motion in the breadboard, leading the Michelson interferometer (that is in place to measure Crackling noise) to be disturbed. Therefore, it is essential to keep the breadboard ”calm” around resonances. Put in more technical terms, the Quality Factory (Q) of the resonant poles must be reduced. This is what forms the problem statement for my work: **to design a feedback controlled damping system for the suspended optics breadboard in Crackle2**.

In the next section, I describe my work in working towards designing, developing and implementing this control system.

4 Progress

In this section, subsections are written in a particular order so as to build up to the said final goal of my work.

4.1 Sensing motion of breadboard using OSEM

This subsection talks about using the suspension OSEM to track the motion of the breadboard in physical degrees of freedom.

4.1.1 Analytical model

Figure 6 shows the mounting scheme of the six OSEMs.

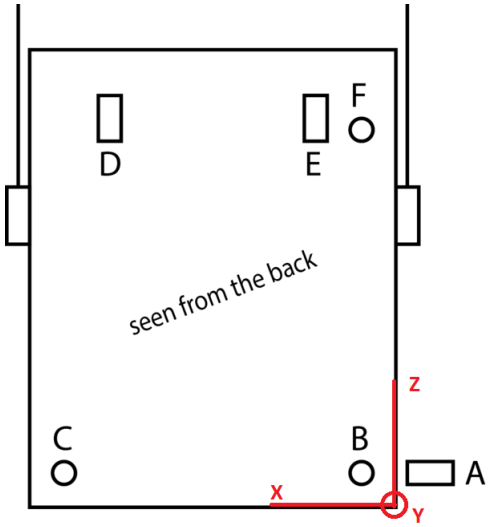


Figure 6: Scheme of OSEMs
Circle, at the OSEM label, indicates that the axis of the OSEM is into the page, and rectangle indicates that the axis points vertically upwards.

It is possible to use the outputs of these shadow sensors to sense motion of the suspended breadboard. Here, I build an analytical model for the same.

I start with the basic relation that gives the velocity $\mathbf{v_p}$ at a point P whose position vector is $\mathbf{r_p}$ on a rigid body whose center of mass (CM) moves with a velocity $\mathbf{v_{cm}}$. The angular velocity is ω. The relation can be written as:

$$\mathbf{v_p} = \mathbf{v_{cm}} + \omega \times \mathbf{r_p} \tag{1}$$

In an infinitesimal time interval, the equation can be rewritten as:

$$\mathbf{s_p} = \mathbf{s_{cm}} + \mathbf{d\theta} \times \mathbf{r_p} \tag{2}$$

where now, $\mathbf{s_p}$ is the infinitesimal displacement vector of the point P, $\mathbf{s_{cm}}$ is the infinitesimal displacement vector of the center of mass and $\mathbf{d\theta}$ represents infinitesimal rotation of the position vector of point P, $\mathbf{r_p}$, about the center of mass.

Equation (2) can be written in the matrix form as:

$$\begin{pmatrix} s_{p,x} \\ s_{p,y} \\ s_{p,z} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & -r_{p,z} & r_{p,y} \\ 0 & 1 & 0 & r_{p,z} & 0 & -r_{p,x} \\ 0 & 0 & 1 & -r_{p,y} & r_{p,x} & 0 \end{pmatrix} \begin{pmatrix} s_{cm,x} \\ s_{cm,y} \\ s_{cm,z} \\ d\theta_x \\ d\theta_y \\ d\theta_z \end{pmatrix} \tag{3}$$

Here, $s_{p,x}$ corresponds to displacement of point P in the x direction.

Now, in the case of interest here, each of the 6 sensors has one such matrix equation. However, note that by construction, **each OSEM can measure displacement for only motion along its axis**. Coupling this with the fact that each of the six sensors are mounted such that their axis is along one of the principal axes (as can be seen in figure 6), it is clear that **only one of the three linear equations represented in equation (3) becomes relevant**.

Let us see this for an example: sensor A. The equation corresponding to $s_{a,x}$ alone becomes relevant because the axis of sensor A is along the absolute (w.r.t ground) x-axis. Therefore, the exact linear equation governing $s_{a,x}$ can be written down by finding out the values of $r_{a,z}$ and $r_{a,y}$. The same can be repeated for all the 6 sensors to obtain a set of 6 linear equations. Notice that the terms on the left hand side of these equations correspond to displacements along sensor axes, and those on the right hand side to motion of the center of mass and rotation about the absolute (w.r.t. ground) axes: roll, pitch and yaw. The desired matrix transformation equation can, then, be established from these six equations, and is the following:

$$\begin{pmatrix} s_a \\ s_b \\ s_c \\ s_d \\ s_e \\ s_f \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & -z_{cm,A} & y_{cm,A} \\ 0 & -1 & 0 & z_{cm,B} & 0 & -x_{cm,B} \\ 0 & -1 & 0 & z_{cm,C} & 0 & -x_{cm,C} \\ 0 & 0 & 1 & -y_{cm,D} & x_{cm,D} & 0 \\ 0 & 0 & 1 & -y_{cm,E} & x_{cm,E} & 0 \\ 0 & -1 & 0 & z_{cm,F} & 0 & -x_{cm,F} \end{pmatrix} \begin{pmatrix} s_{cm,x} \\ s_{cm,y} \\ s_{cm,z} \\ d\theta_x \\ d\theta_y \\ d\theta_z \end{pmatrix} \tag{4}$$

Here, as usual, $x_{cm,A}$ refers to the x-coordinate of A in the CM frame. Also recall that these values can be negative too, since the center of mass is close to the geometric center of the breadboard and OSEMs are placed on either side.

4.1.2 Constructing the transformation: Sensing matrix

As mentioned above, the sensing transformation matrix can be arrived at using the set of 6 linear equations. The positions of the sensors **w.r.t the CM** fix the constants of the matrix.

1. **Finding coordinates of the center of mass (CM)**
   First, we find out the coordinates of the CM. The origin is as shown in figures 4 and 6. The suspension is as shown in figure 5. The axis convention is shown in figure 6.

   The y-coordinate can be simply found to be -27 (in mm) by noticing the position of the suspension wires from the center plane of the breadboard.

   For the x and z coordinates, I have estimated the mass and position of various components of the setup. Table 1 summarizes the same.

   Upon calculation, the center of mass of the system comes out to be: 20.49kg at **(204, -27, 223)** (in mm).

Table 1: Calculation of CM coordinates of the setup

Item(s)	**Mass (kg)**	**Absolute coordinates (in xz plane, origin as shown in figure 5) (mm)**	**Comments**
Optical posts and components	12 x 0.4 = 4.8	(217.83, 177.75)	0.2kg is the weight of each post, and 0.2kg accounts for the mass of the component placed on it. Figure 2 shows the positions of these posts/components. The coordinates posted here correspond to the CM of the system of posts and components
Suspended block 1	2.2	(356, 210)	
Suspended block 2	2.2	(44, 210)	
Fixed block	1.45	(200, 400)	This is the block to which steel blades are clamped
Breadboard	5.9	(200, 225)	
Reinforcement 1	1.97	(380, 225)	Mass calculated from material and dimensions
Reinforcement 2	1.97	(20, 225)	Mass calculated from material and dimensions

2. **Calculation of positions of sensors w.r.t CM** The next thing I did was to find out the positions of each of the sensors in the absolute (w.r.t. ground) reference frame, and then find their coordinates in the center of mass frame. Refer to table 2. 'X' means that we don't care about that particular coordinate since it is along the axis of the sensor.

Table 2: Positions of sensors

Sensor	**Absolute coordinates (in xz plane, origin as shown in figure 5) (mm)**	**Coordinates in CM frame (mm)**
A	(X, 0, 62.5)	(X, 27, -160.6)
B	(24, X, 62.5)	(-180.16, X, -160.6)
C	(376, X, 62.5)	(171.82, X, -160.6)
D	(316, 25.7, X)	(111.82, 52.7, X)
E	(84, 25.7, X)	(-120.18, 52.7, X)
F	(24, X, 397)	(-180.18, X, 173.91)

The matrix transformation equation can, after converting the length units to meter (so that angles are in micro-rad), then be written as:

$$\begin{pmatrix} s_a \\ s_b \\ s_c \\ s_d \\ s_e \\ s_f \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0.1606 & 0.027 \\ 0 & -1 & 0 & 0.1606 & 0 & -0.18018 \\ 0 & -1 & 0 & 0.1606 & 0 & 0.17182 \\ 0 & 0 & 1 & -0.0527 & 0.11182 & 0 \\ 0 & 0 & 1 & -0.0527 & -0.12018 & 0 \\ 0 & -1 & 0 & -0.17391 & 0 & -0.18018 \end{pmatrix} \begin{pmatrix} s_{cm,x} \\ s_{cm,y} \\ s_{cm,z} \\ d\theta_x \\ d\theta_y \\ d\theta_z \end{pmatrix} \tag{5}$$

On multiplying the above equation on both sides by the inverse of the matrix, we get:

$$\begin{pmatrix} s_{cm,x} \\ s_{cm,y} \\ s_{cm,z} \\ d\theta_x \\ d\theta_y \\ d\theta_z \end{pmatrix} = \begin{pmatrix} 1 & 0.0774 & -0.0767 & -0.6922 & 0.6922 & -6.88 \times 10^{-4} \\ 0 & -0.0034 & -0.5119 & 0 & 0 & -0.4847 \\ 0 & 0.1575 & 0 & 0.5180 & 0.4820 & -0.1575 \\ 0 & 2.989 & 0 & 0 & 0 & -2.989 \\ 0 & 0 & 0 & 4.31 & -4.31 & 0 \\ 0 & -2.87 & 2.84 & 0 & 0 & 2.5478 \times 10^{-2} \end{pmatrix} \begin{pmatrix} s_a \\ s_b \\ s_c \\ s_d \\ s_e \\ s_f \end{pmatrix} \tag{6}$$

Once again, recall that the angles are in micro-rad and displacements in micron.

4.1.3 A brief about sensing and actuation using OSEM

Optical **S**hadow Sensor and **E**lectro**M**agetic Actuator (OSEM) consists of a shadow sensor part and an electromagnetic actuator part. The former consists of an LED and a photodetector mounted opposite to each other, as shown in figure 7. Insertion of an object through the hole would block the light from the LED reaching the photodetector, and this change can be used to quantify the extent to which the object has come inside the hole. The actuator part consists of a coil which can generate magnetic field upon driving current through it.

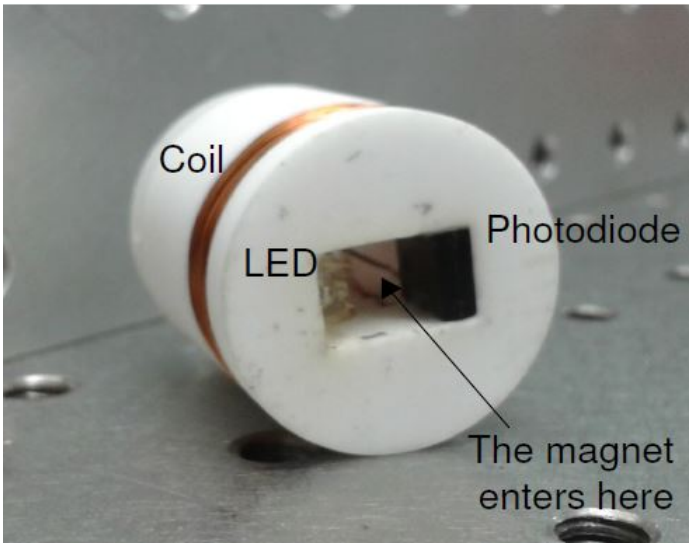


Figure 7: Optical Shadow Sensor and Electromagnetic Actuator

Upon thought, one can easily see that both sensing and actuation can be done by the same unit by simply using a magnetic material for the object whose displacement is to be sensed; and this is precisely what is done: magnets are attached to the points on the breadboard indicated by points A through F. The OSEM modules are mounted outside the suspension, and they are at rest w.r.t the ground.

Then, motion of the magnet can be sensed as usual, and it can even be driven by varying the current through the coil. This is what is used as an actuation for the breadboard.

4.2 Reconstruction of breadboard motion in 6D

Having constructed the sensing matrix, the next thing I did was to try it out on some real data. The sensor data was exported to struct files using the ligoDV interface. (The MATLAB ligoDV interface can be used to acquire sensor data, from a specified period of time, from the relevant server. In this report, I will not delve into any technical details of the interface or the settings.)

I wrote a MATLAB script that extracted sensor data arrays in double format from the struct files. Since we are interested in infinitesimal displacements around the equilibrium point, I subtracted from each sensor data array the mean of the data. Then, I inverted the transformation matrix from equation 5 to reconstruct motion in the ground frame.

Figure 8 shows plots of motion in 6 d.o.f.

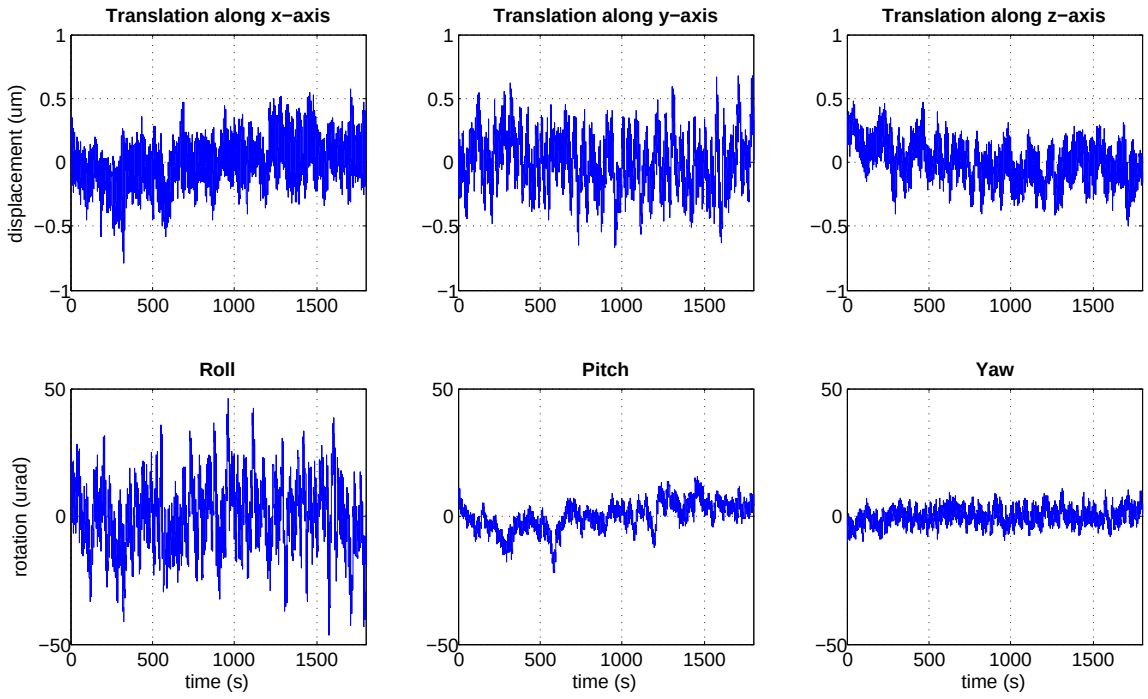


Figure 8: Plot of motion in 6 d.o.f

Notice that the order of magnitude of motion is only a fraction of a micron. These plots cannot really reveal any interesting information. What can be more interesting is the spectra of these motions: one can see the modes and resonance frequencies.

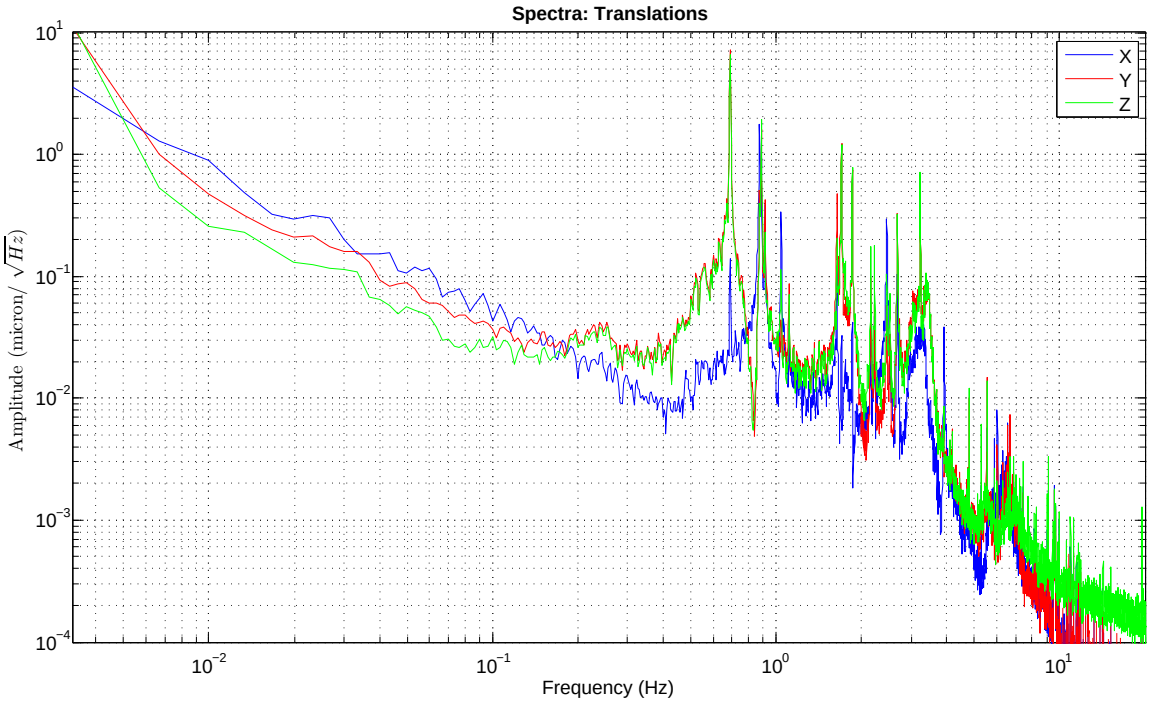
4.3 Spectral analysis

I have used the pwelch function in MATLAB to find the power spectral density (PSD) of the collected motion data. The following points summarize the process.

- `[Pxx,f] = pwelch(data, hanning(nfft), nfft/2, nfft, fs)`

  on MATLAB gives the PSD **Pxx** and the corresponding frequency array **f**
- nfft was chosen to be 1/6th of the total number of points, so as to guarantee about 10 averages through hanning windowing.
- fs represents the sampling frequency, it is equal to 2048Hz.
- The square root of the PSD, amplitude, was used in the plots.

Figure 8 shows the spectra corresponding to motion shown in figure 6.



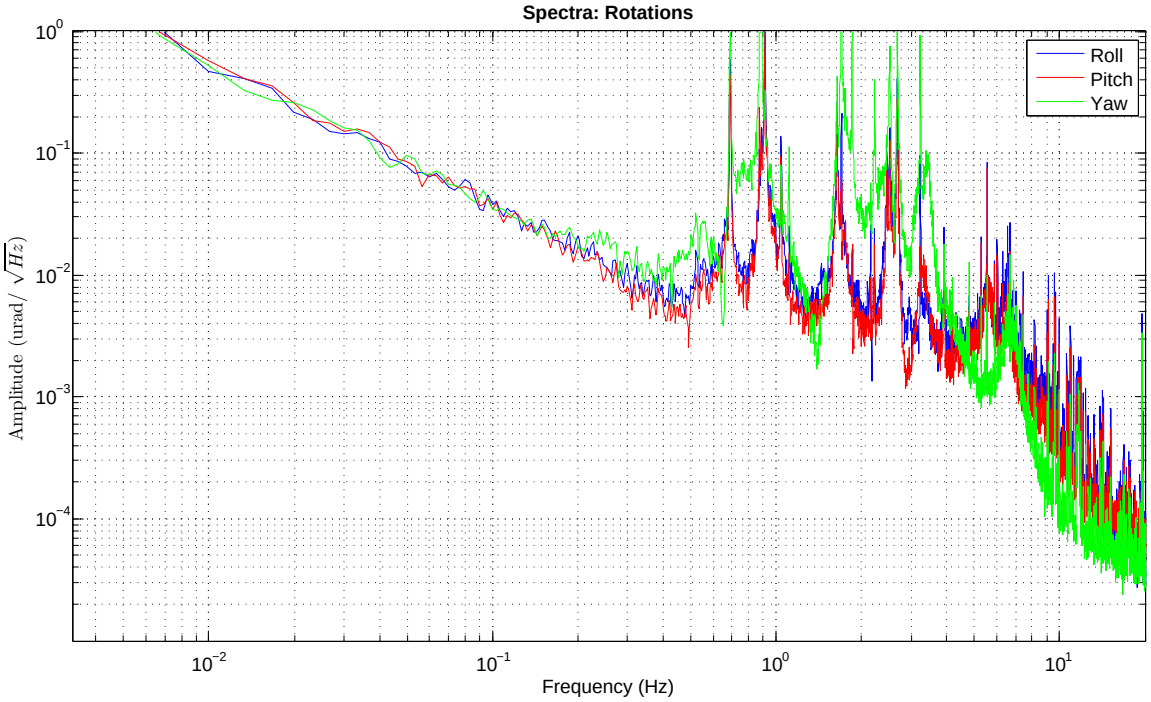


Figure 8: Spectra of motion in 6 d.o.f

From the plots in figure 8, one can infer that:

1. The breadboard is suspended, and the OSEMs are fixed relative to the ground. Most of what is seen in the spectrum is due to motion of the ground (seismic activity); this seems to be concentrated in the $<$10Hz region. **This is what forms the region of interest for us: the aim would be to damp motion at resonant frequencies of the suspension. Seismic isolation has been achieved by using a suspension, but the modes of the suspension are in the 0.1-10Hz region. At these frequencies, the breadboard moves more than the ground, and so motion must be damped.**
2. All the plots peaks at a particular (common) frequency just below 1Hz – one of the mode frequencies of the system. A few more peaks can be seen for a few frequencies in between 1Hz and 10Hz.

The idea is to damp motion of the breadboard at resonant frequencies. This would require us to have an idea of the mechanical response of the system for damping using OSEM coils at positions A through F. This forms the next part of my work.

4.4 Characterization of mechanical response of the suspension

Six OSEMs, A through F, are mounted at different positions in the cage in which the breadboard is suspended. The scheme has been depicted in figure 6. It is important to understand how OSEMs

function in order to get a feel for these measurements.

4.4.1 Transfer Function Measurements: The How

Characterization of the mechanical response of the system consists of exciting one of the coils and recording response in all other sensors or in motion in physical d.o.f. Both are equivalent and are related by the same transformation as the displacements are. I have chosen to directly excite and measure response in physical d.o.f. The following will explain the same in detail and discuss some results.

As I discussed above, by using magnets, it becomes possible for one to modify as well as sense motion of a point on the suspended optical breadboard. Transfer function measurement, then, becomes a matter of actuating breadboard motion along one degree of freedom (using the Driving and Sensing matrices both; the former is described in the next part), and measuring displacements in all degrees of freedom; the LIGO DTT provides a pretty convenient way of doing this. Before I talk about the results, it is important to note the following about the DTT configuration used for the measurements.

- The DTT is operated in a **swept-sine mode** in order to perform the desired measurements. The choice is obvious: we wish to study the behavior of sensor displacement for various frequencies of excitation.
- I have, in the measurements that I describe below, swept the excitation frequency from 0.5 Hz to 5 Hz.
- It is important to realize that the number of counts output to the coil under excitation depends on the frequency of excitation. This is because of the presence of whitening and de-whitening filters. Given the filter shapes, this means that at lower frequencies, due to lower gain from whitening and de-whitening filters, the output counts are lower as compared to that at higher frequencies **for the same input amplitude**. It is also known that a higher amplitude of excitation gives higher coherence. Combining both of the above facts, it is interesting to see that one can improve coherence at lower frequencies by exciting the coil at higher amplitudes than those at higher frequencies for a similar number of counts output to the coil. As long as the number of counts output to the coil is below a saturation limit (which is about 16000), this is perfectly safe.

  This was exploited, and the excitation amplitude profile *for a single coil* can be modified as shown in Table 3. Note that what is shown in table 3 is only for illustrative purposes; while taking measurements, **I have in fact modified the excitation amplitude profile for one each physical d.o.f. in a similar way**.

  .

It is possible to command DTT to build an envelope around the above points and then use that

Table 3: Excitation amplitude profile for a single coil (*This is for illustrative purposes; read the explanation above)*

Frequency	Excitation amplitude (uN)	No. of counts output to coil
15	400	10000
10	600	10000
5	1200	10000
2	2800	10000
1	4700	10000
0.5	6300	10000

profile to define amplitudes for other frequencies in the range of measurement. Alternatively, one can also feed discrete data points as input.

4.4.2 Driving motion along physical d.o.f.: Driving matrix

Using the sensing matrix in (6), it is possible to sense motion along physical d.o.f. We would, however, also like to drive the suspended breadboard along physical d.o.f.: these are the d.o.f.'s that will be relevant in the feedback damping system.

For this purpose, I have constructed a transformation equation, simply based on Newton's 2nd law along various d.o.f.'s.

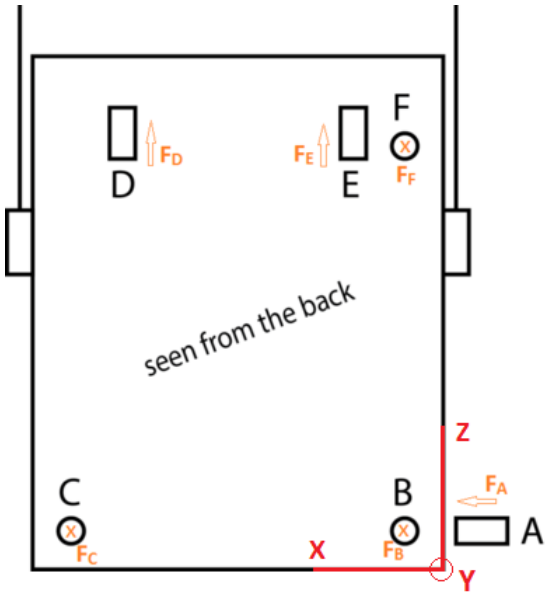


Figure 10: Scheme of OSEM's with force sign convention shown

Figure 10 shows OSEM scheme along with the sign convention of driving force for each OSEM. The convention is that **for a positive drive, the magnet moves OUT of the OSEM**.

One can, by looking at figure 10, easily write down the following equations:

$$F_x = F_A \tag{7}$$

$$F_y = -F_B - F_C - F_F \tag{8}$$

$$F_z = F_D + F_E \tag{9}$$

Similarly for the net torque:

$$\tau_{net} = \sum_{i=A}^{F} \vec{r}_i \times \vec{F}_i \tag{10}$$

where $\vec{r}_i$ and $\vec{F}_i$ are position and force vectors of OSEM 'i'.

Equation (10) can be expanded as below:

$$\tau_{net} = \vec{r_A} \times (F_A \hat{e_x}) + \vec{r_B} \times (F_B(-\hat{e_y})) + \vec{r_C} \times (F_C(-\hat{e_y})) + \vec{r_D} \times (F_D \hat{e_z}) + \vec{r_E} \times (F_E \hat{e_z}) + \vec{r_F} \times (F_F(-\hat{e_y})) \tag{11}$$

F_i corresponds to force by each OSEM; the negative sign in some terms is because of the orientation of the OSEM's. (For example, positive force – which we defined as when the magnet moves out – along B would mean a displacement in negative y direction.)

The x, y, z components of torque can be separated and a **matrix transformation equation** can be written from linear equations 6 through 10 as follows:

$$\begin{pmatrix} F_x \\ F_y \\ F_z \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & z_{cm,B} & z_{cm,C} & y_{cm,D} & y_{cm,E} & z_{cm,F} \\ z_{cm,A} & 0 & 0 & -x_{cm,D} & -x_{cm,E} & 0 \\ -y_{cm,A} & -x_{cm,B} & -x_{cm,C} & 0 & 0 & -x_{cm,F} \end{pmatrix} \begin{pmatrix} F_A \\ F_B \\ F_C \\ F_D \\ F_E \\ F_F \end{pmatrix} \tag{12}$$

Recall: $x_{cm,A}$, for instance, is the x-coordinate of OSEM A in the CM frame.

On plugging in values of the necessary coordinates from table 2 **in meter**, the equation can be written down as:

$$\begin{pmatrix} F_x \\ F_y \\ F_z \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & -0.1606 & -0.1636 & 0.0527 & 0.0527 & 0.17391 \\ -0.1606 & 0 & 0 & -0.11182 & 0.12018 & 0 \\ -0.0270 & 0.18018 & -0.17182 & 0 & 0 & 0.18018 \end{pmatrix} \begin{pmatrix} F_A \\ F_B \\ F_C \\ F_D \\ F_E \\ F_F \end{pmatrix} \tag{13}$$

Therefore, the Driving Matrix equation is:

$$\begin{pmatrix} F_A \\ F_B \\ F_C \\ F_D \\ F_E \\ F_F \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0.0774 & -0.0034 & 0.1575 & -2.9894 & 0 & 2.8664 \\ -0.0767 & -0.5119 & 0 & 0 & 0 & -2.8409 \\ -0.6922 & 0 & 0.5180 & 0 & -4.3103 & 0 \\ 0.6922 & 0 & 0.4820 & 0 & 4.3103 & 0 \\ -0.00068 & -0.4847 & -0.1575 & 2.9894 & 0 & -0.0255 \end{pmatrix} \begin{pmatrix} F_x \\ F_y \\ F_z \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} \tag{14}$$

Here, forces are in μN and Torque in μNm.

4.4.3 Changing legs of the optics table

The legs of the (horizontal) optics table were changed from the longer, rigid legs to new shorter ones which can ”float” the table on pumping air. The idea was to provide better isolation from seismic motion, especially for vertical displacement - which is of much interest. Acoustic noise can be reduced, too, significantly by installing a vacuum system. The vacuum system was not fully commissioned during my measurements, but the bell jar was in place. Here, I describe the effects of the change in table legs.

Effect on Seismometer spectra

The first thing that we observed was how the motion of the optics table itself has changed at various frequencies, and how the resonance peak has shifted thanks to the new legs. For this purpose, we used a seismometer (a PZT accelerometer coupled with a voltage preamplifier (SR560)).

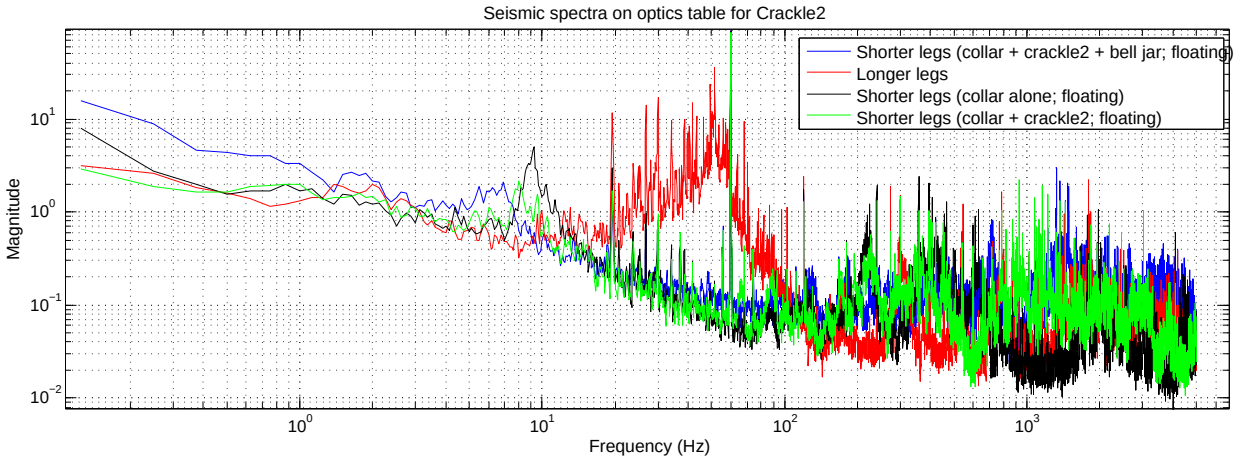


Figure 11: Seismic spectra for vertical displacement on horizontal table for Crackle2. Of primary importance here are the traces in red and blue, where the former case corresponds to the older legs, and the latter corresponds to the case with new legs and the vacuum chamber. Notice the difference in noise levels in the 10s of Hz region in both the cases.

Clearly, the huge bump in the 20-100Hz region is now gone, upon changing the legs. The new table, which can also be "floated" in air, has certainly improved seismic isolation for the whole setup. It is now much more likely to get less noisy transfer function measurements. With this motivation, I started another round of transfer function measurements.

4.4.4 Automation of (multiple) simultaneous transfer function measurements

All of the initial trails of transfer function measurements involved running measurements in series: that is, one coil/d.o.f. excitation after another. Each measurement took about 6-8hrs, and 6 such measurements meant a week's time. It was then required to bring this time down by adjusting the parameters of DTT measurements, such as number of averages, number of cycles of measurements and such.

One can then say, it would be even more efficient if all measurements could be taken in parallel in, say, a night.

Turns out that, in principle, it is possible with our system for the simple reason that it is a linear system. Why is it a linear system? Because Newton's laws – which govern its behavior – are linear!

As long as the system is not excited by the same frequency through different coils at the same instant of time, due to linearity, the measurements should be as good as they were taken independently. The difficult part, however, is to implement it.

It is certainly not possible for one to do it manually each time by opening 6 parallel sessions of DTT. Automation is needed. To **automate** parallel sessions of transfer functions, the following are

some possible methods:

1. Use the Diagnostic Tool (”diag”) through the workstation computer along with a python script
2. Use Python and MATLAB to run DTT measurements

Both of these methods have been tried out before at LLO/LHO. I have initially attempted to work on the second solution. However, I faced some issues with getting the Python-MATLAB interface working. (To be specific, the problem was with Python XML parsing and its interface with MATLAB.) The problem didn't even make much sense, and so I switched to the first method.

In this method, a DTT template for the measurement of interest is first prepared and saved as an XML file. The desired parameters are set; 'Start time' set as 'Now'.

A measurement based on this DTT template can be run by using a short 'diag' script. The script itself is simple, and goes as follows:

```
open
restore PATH–TO–DIRECTORY/IN_FILE_NAME.xml
run –w
save PATH–TO–DIRECTORY/OUT_FILE_NAME.xml
exit
```

Here, **path to directory and file names are to be set appropriately**.

The above script is for the Diagnostic Tools ”diag”. In order to automate this for all 6 coil excitations, I have written a short Python script based on the 'subprocess' module. In short, the whole process can be summarized as:

1. Input DTT templates are saved in directory
2. 'diag' scripts are prepared appropriately and saved in the same directory
3. A python script to execute all the 6 'diag' scripts in parallel terminal sessions is then executed

4.4.5 Transfer Function Measurements: Results

After having developed the whole system of actuating and sensing degrees of freedom that are of interest, automating measurements and making simultaneous measurements possible, and having gotten the experimental setup ready to an optimal working stage, transfer function measurements were performed again. Here, I show only the final results; the ones at various stages are not of much interest.

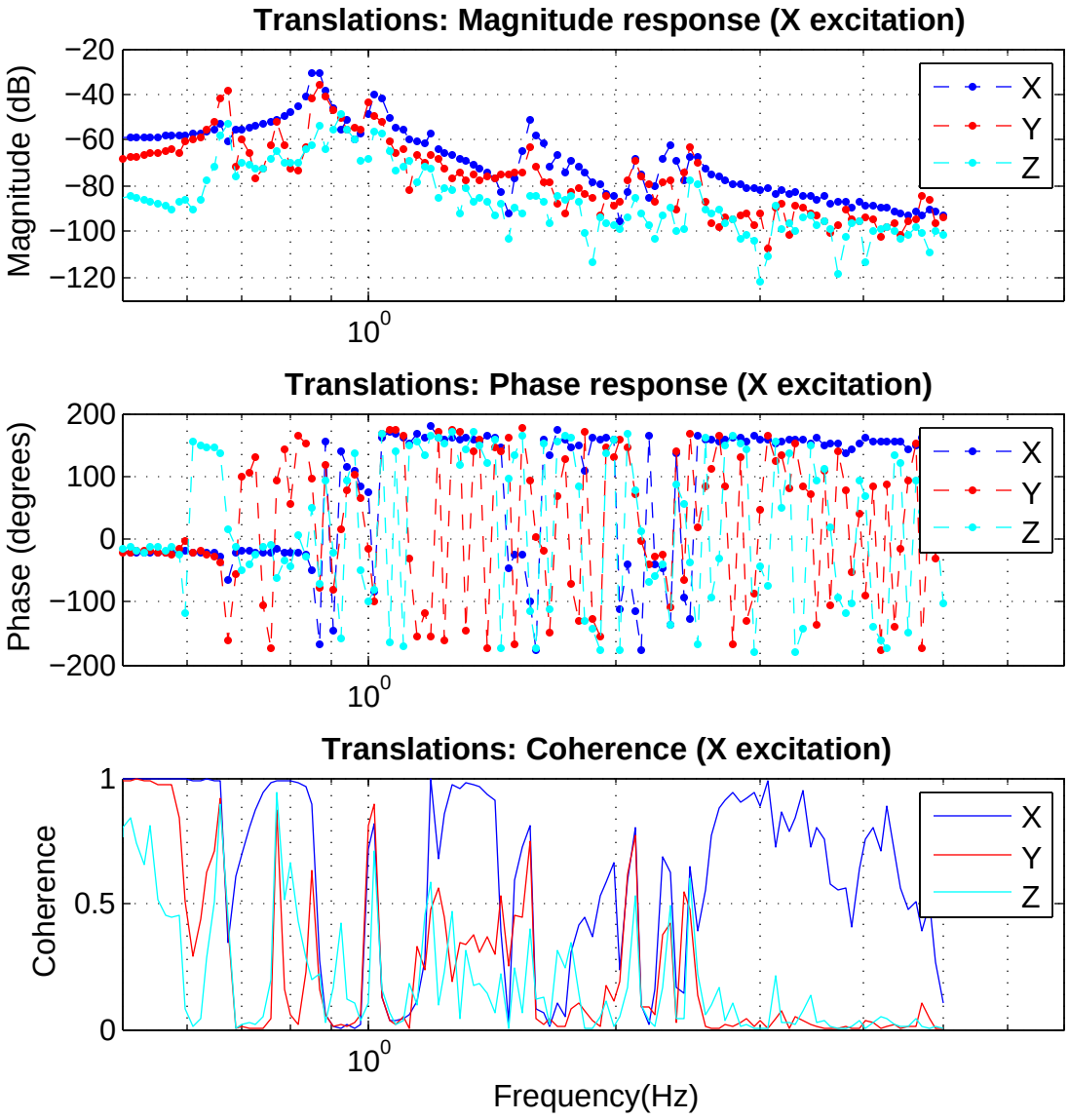


Figure 12: Bode plots for translational d.o.f. for X excitation

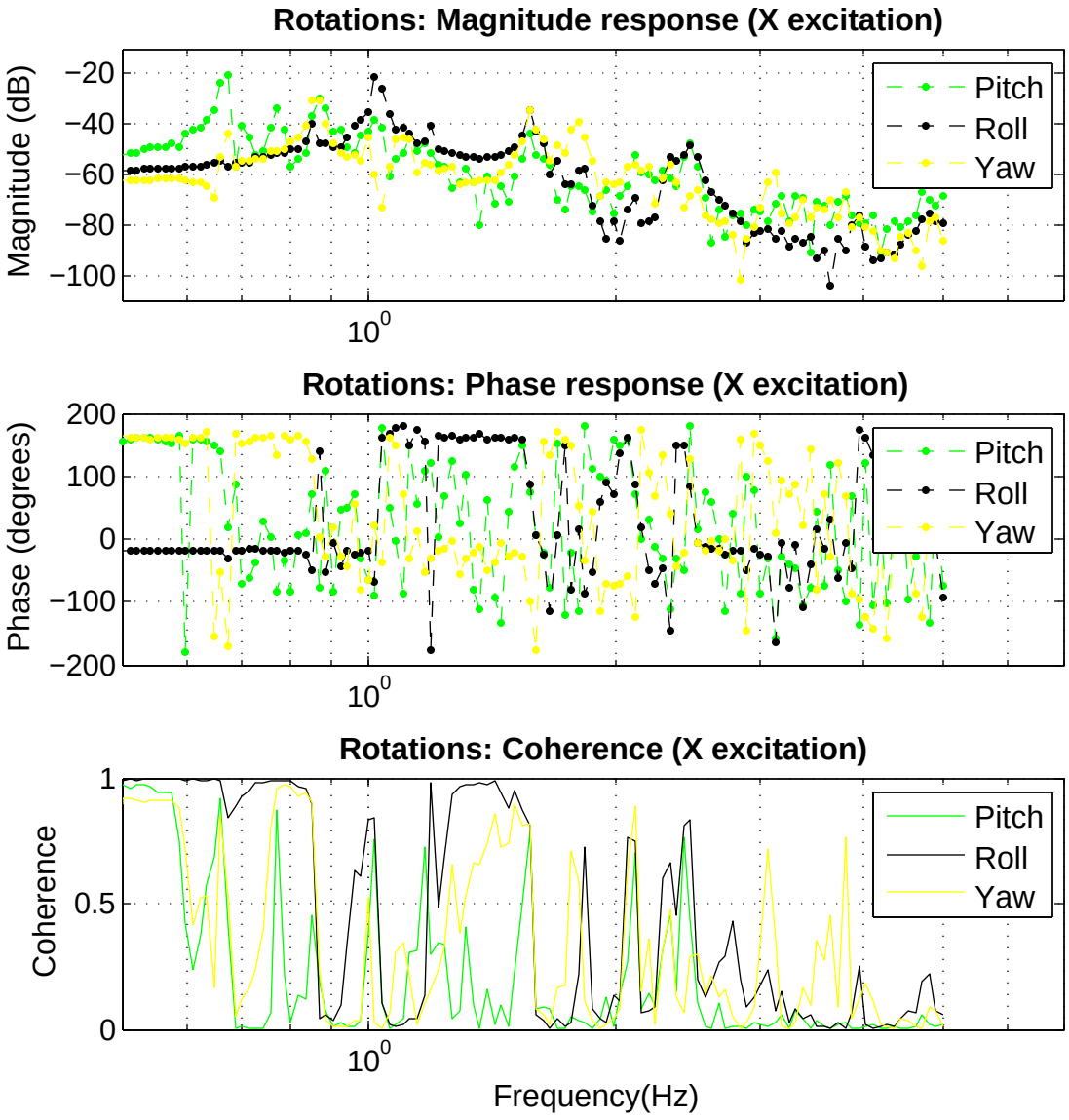


Figure 13: Bode plots for rotational d.o.f. for X excitation

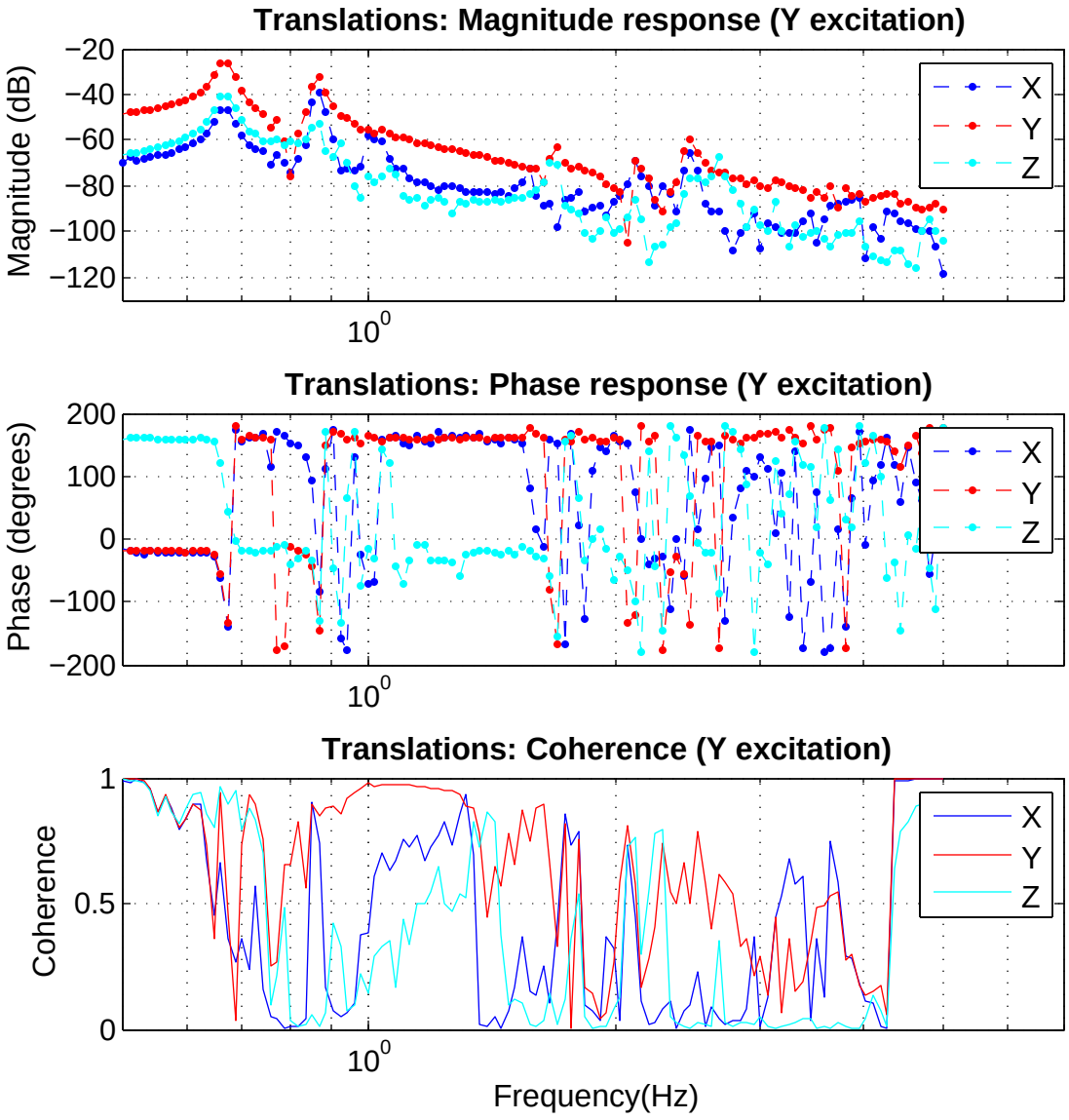


Figure 14: Bode plots for translational d.o.f. for Y excitation

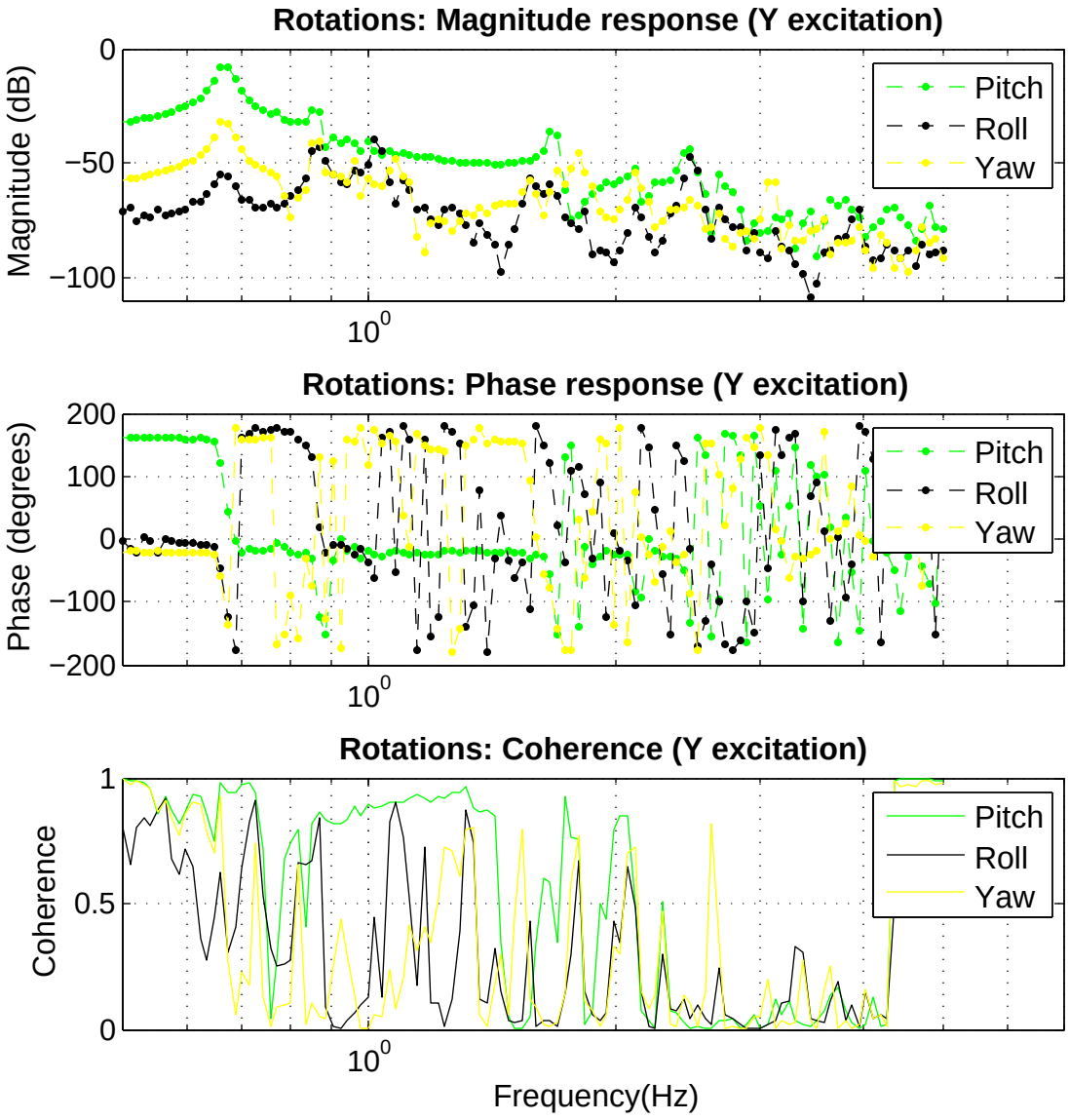


Figure 15: Bode plots for rotational d.o.f. for Y excitation

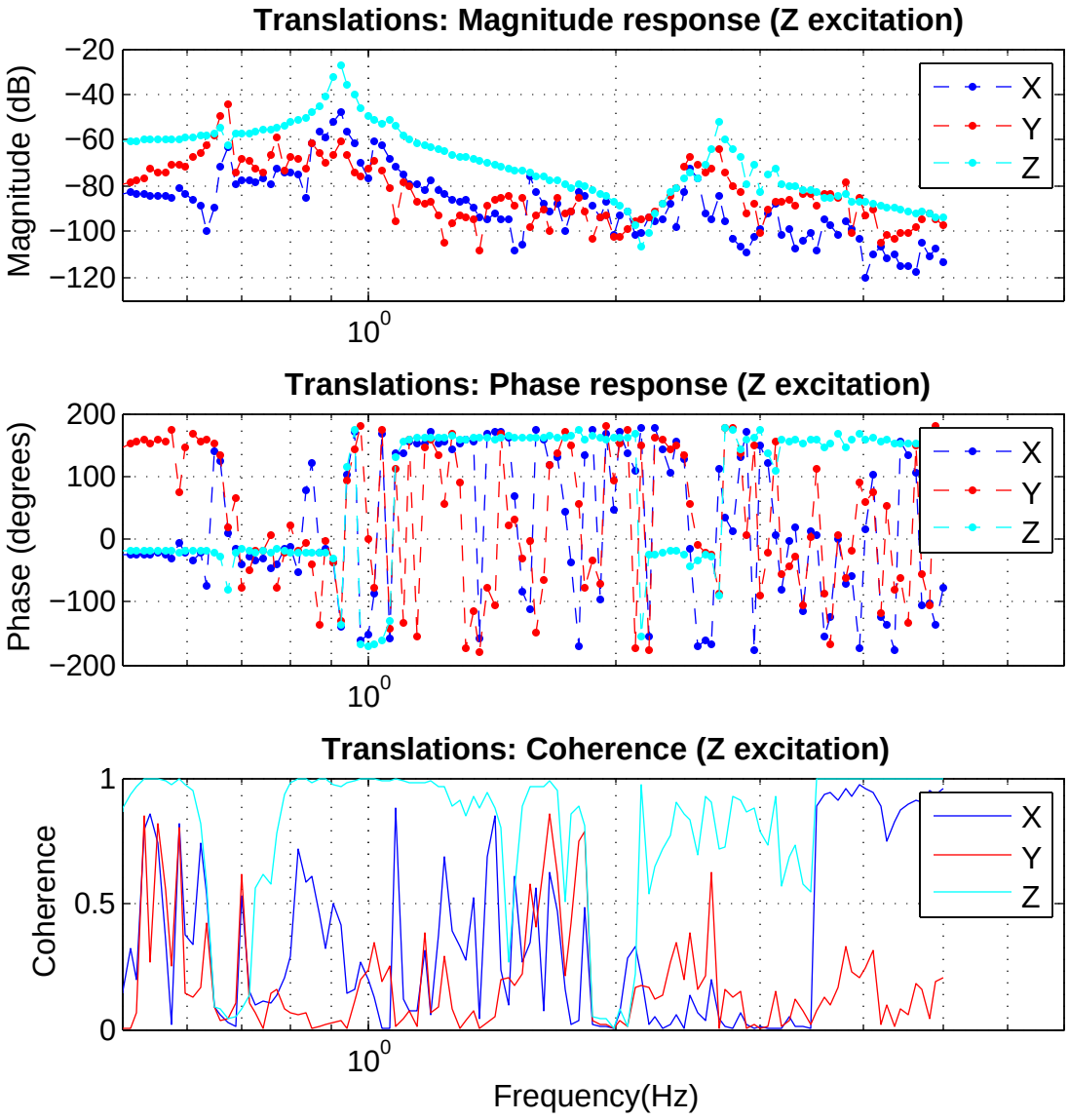


Figure 16: Bode plots for translational d.o.f. for Z excitation

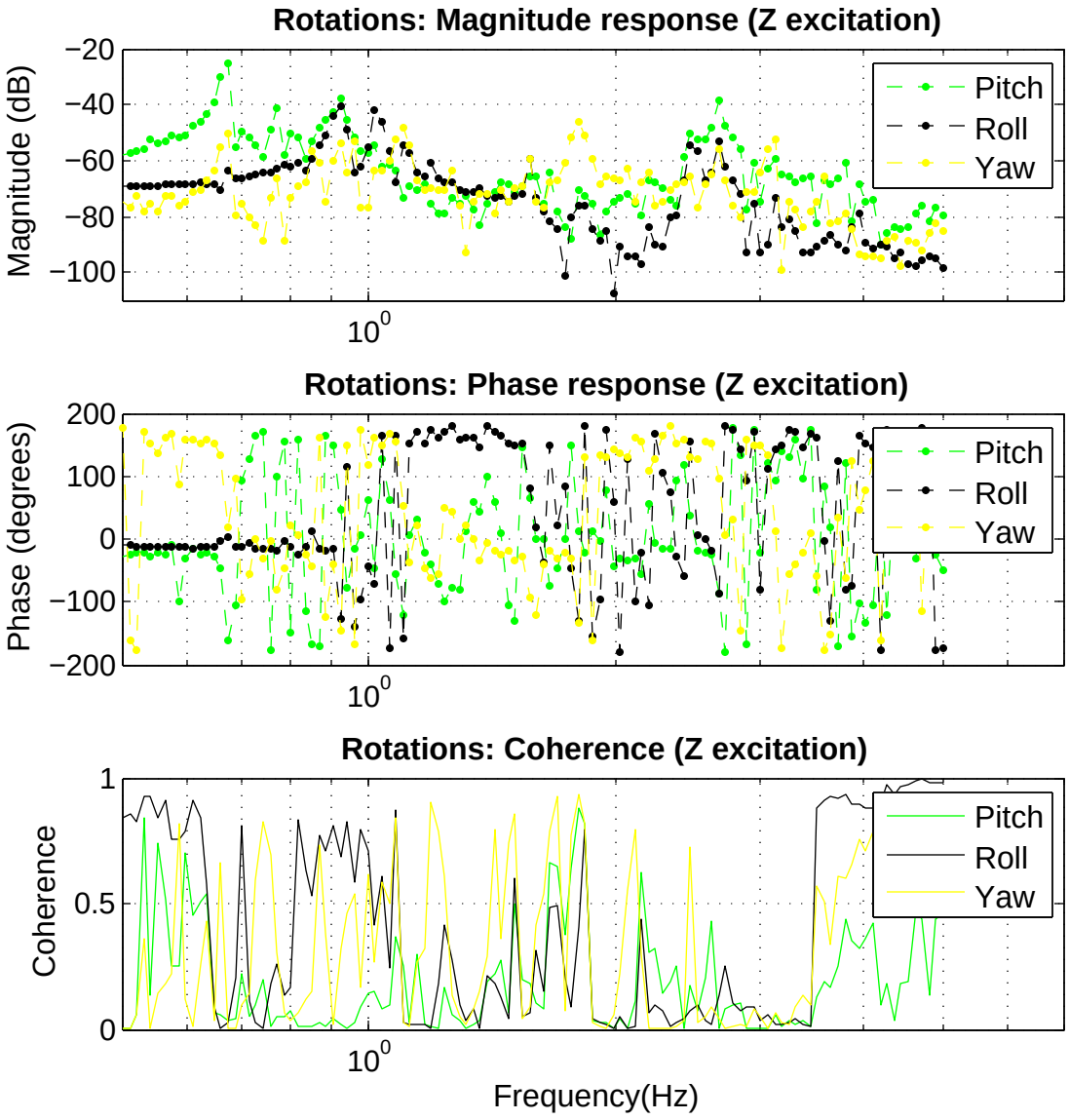


Figure 17: Bode plots for rotational d.o.f. for Z excitation

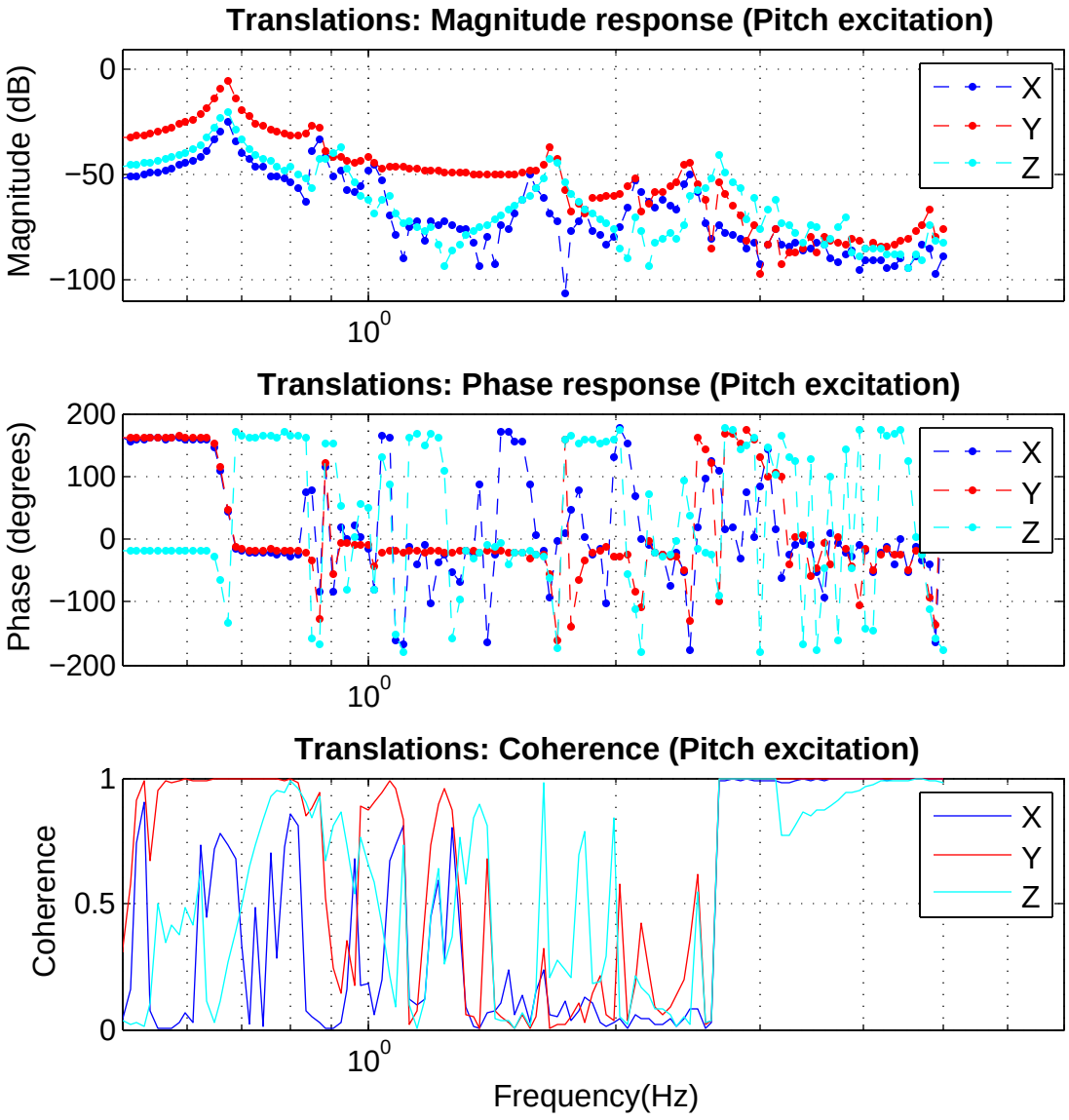


Figure 18: Bode plots for translational d.o.f. for Pitch excitation

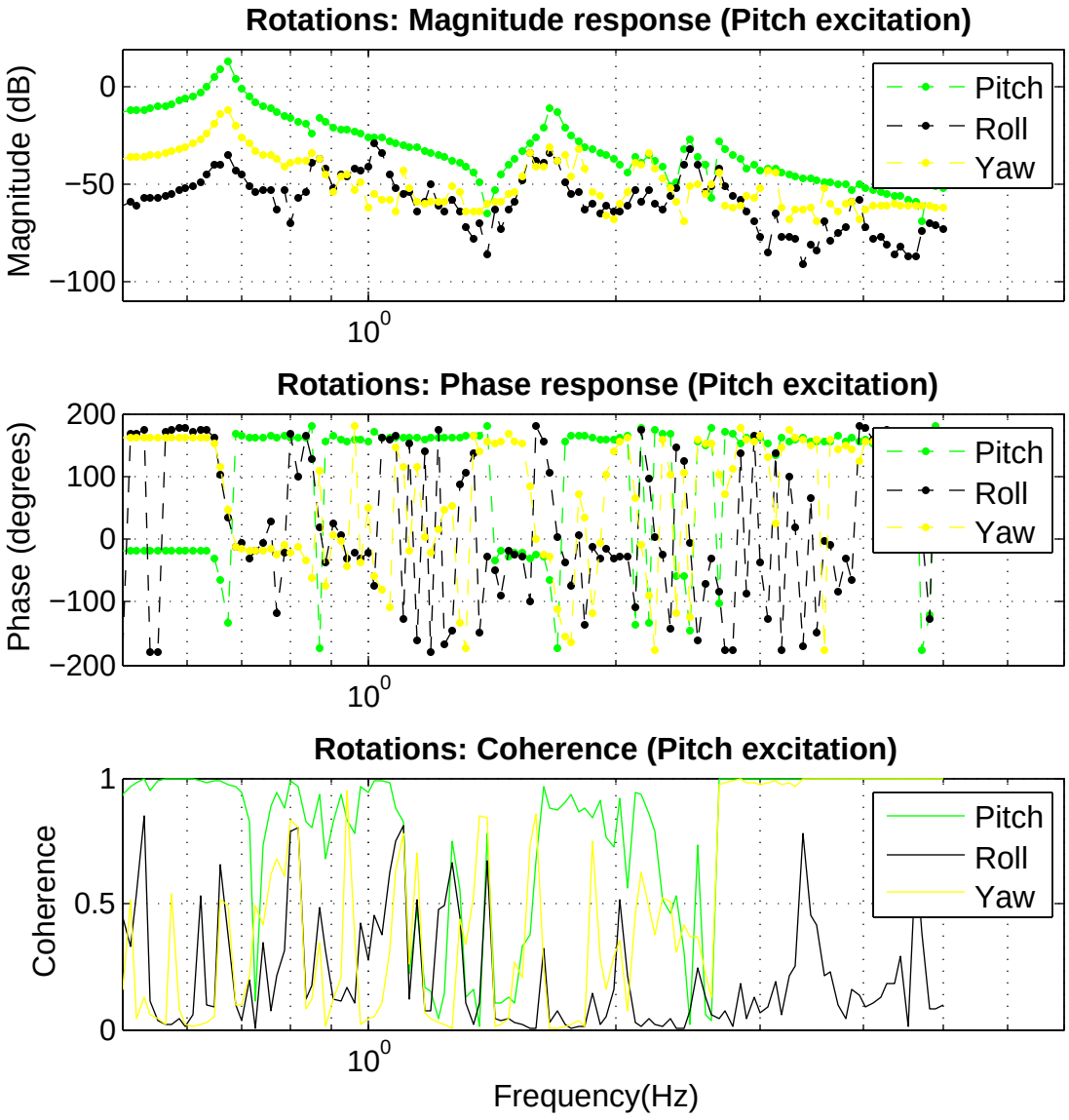


Figure 19: Bode plots for rotational d.o.f. for Pitch excitation

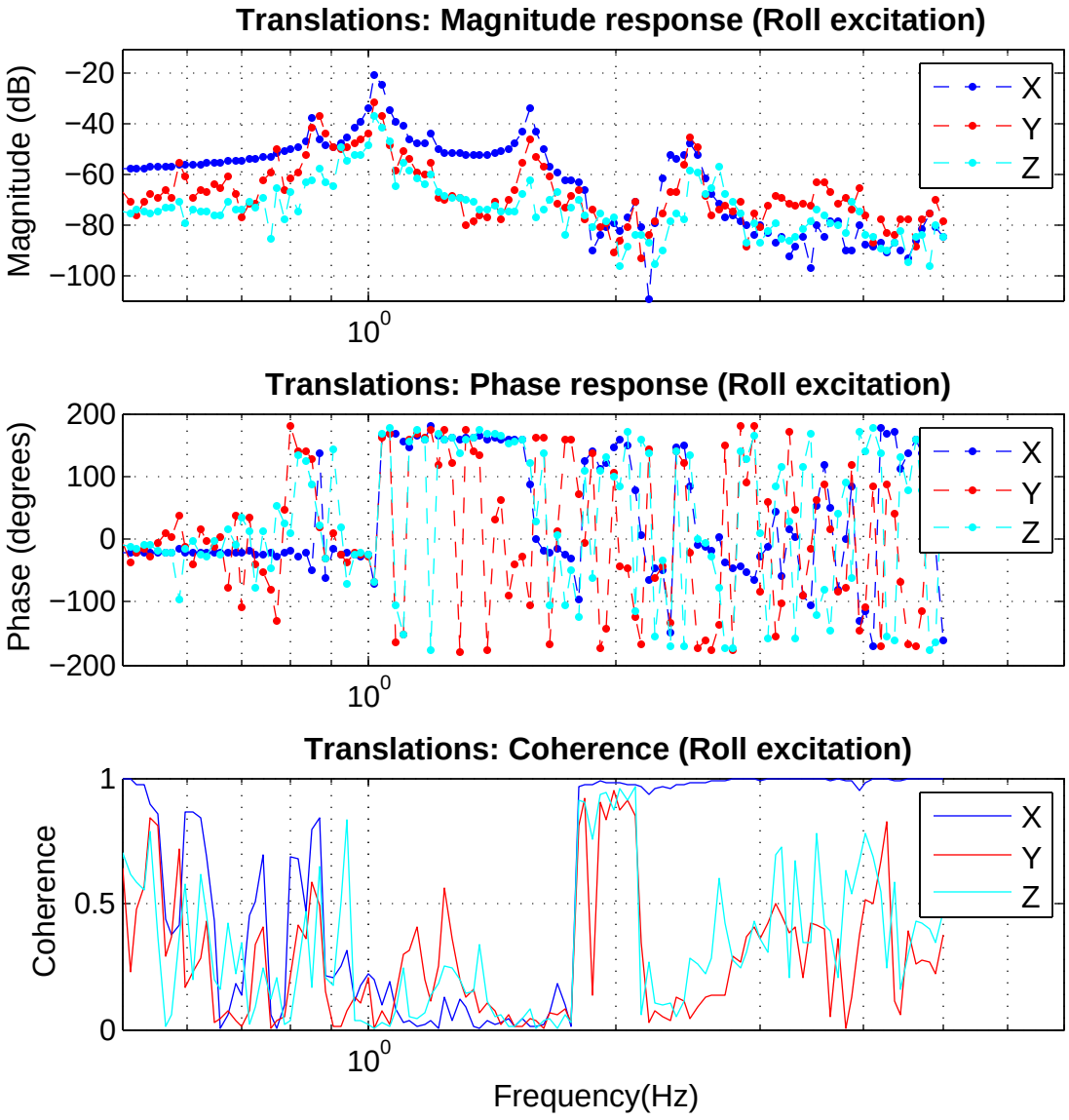


Figure 20: Bode plots for translational d.o.f. for Roll excitation

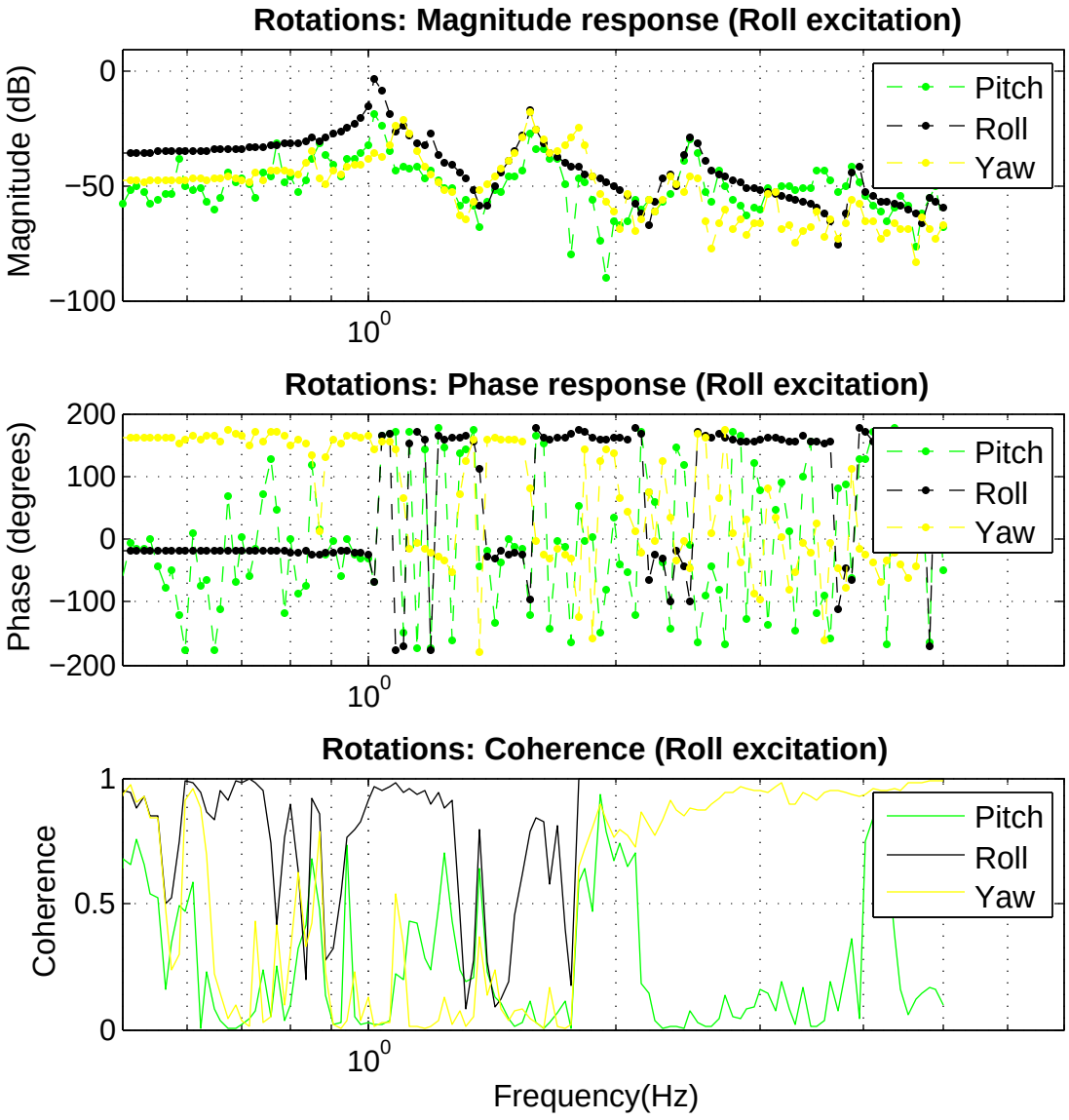


Figure 21: Bode plots for rotational d.o.f. for Roll excitation

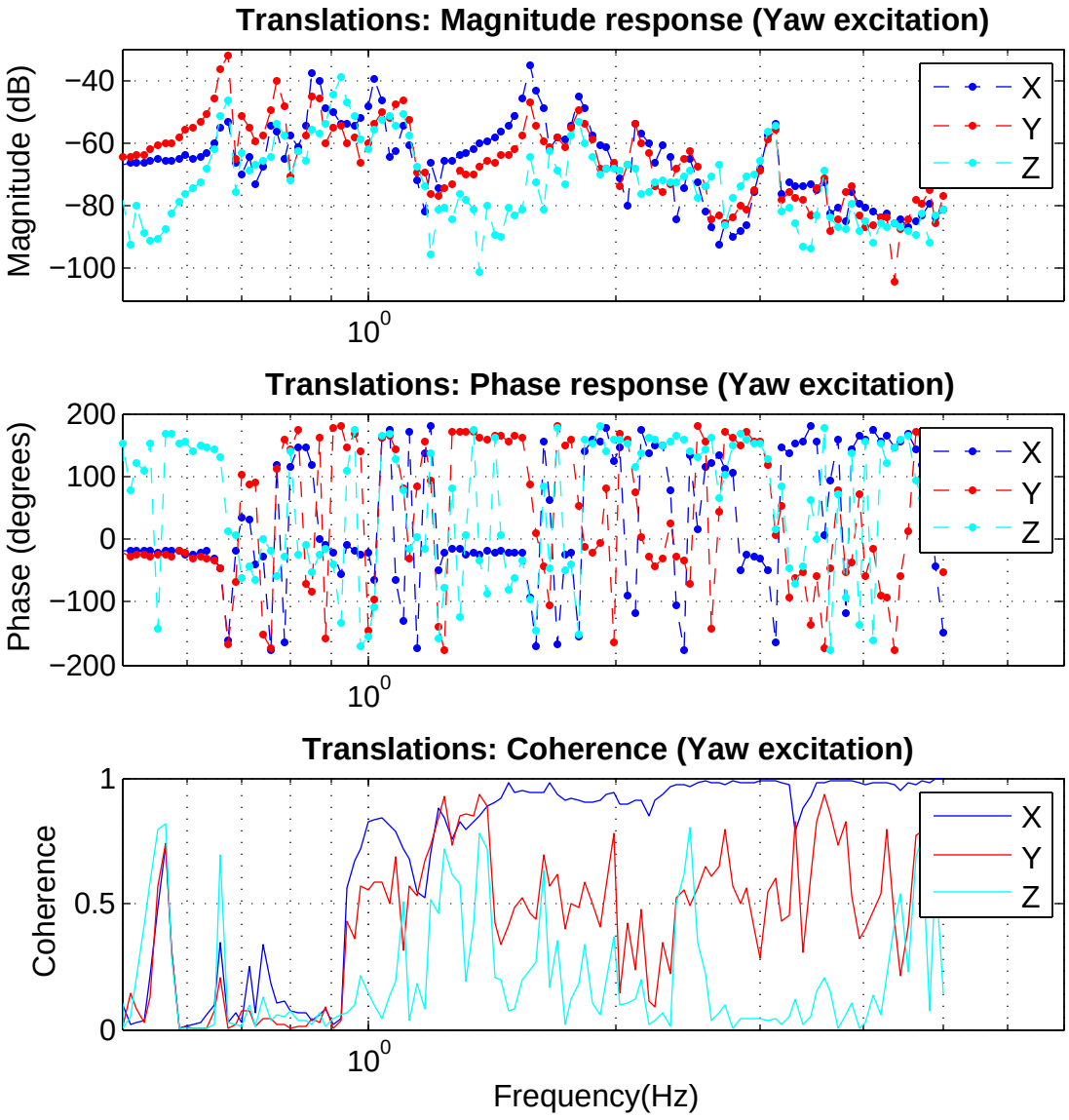


Figure 22: Bode plots for translational d.o.f. for Yaw excitation

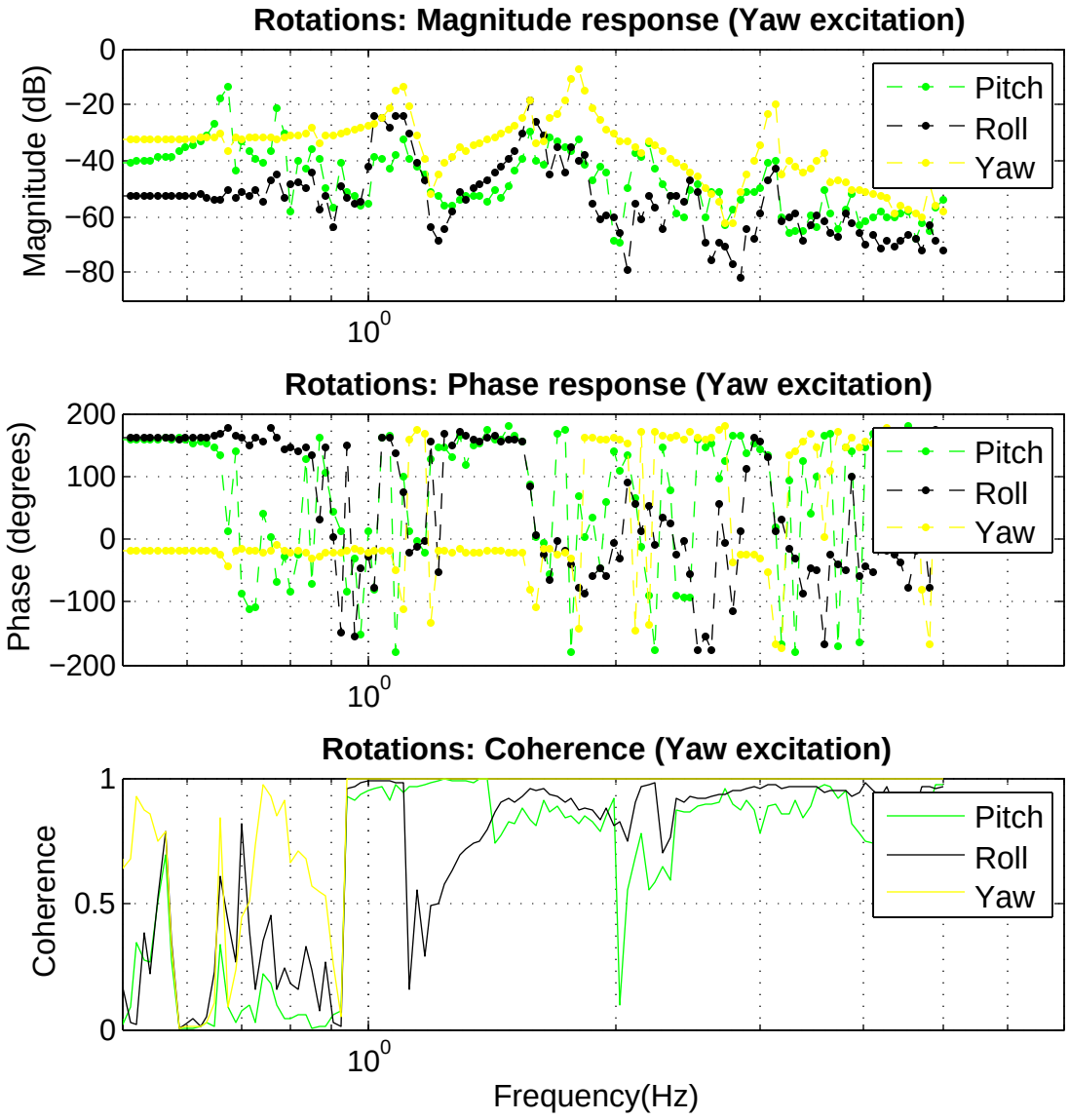


Figure 23: Bode plots for rotational d.o.f. for Yaw excitation

Figures 12 through 23 show bode plots of transfer functions for sensing and actuation all six degrees of freedom. One can easily notice two peaks in most of the magnitude plots, and this is characteristic of a double spring-mass or double pendulum system. It can be a tedious job to compute the location of each and every pole and zero by hand and compare with the experimental result. A smarter thing to do would be to simulate the mechanical system and compare results to see if the results make sense overall.

4.4.6 Analytical Modeling of Crackle2 Setup

After spending some time on a Mathematica model that was built earlier for Crackle2 and not getting satisfactory results, I moved on to a Mathematica-based SUspension Model CONstructor (SUMCON) built by folks at KAGRA. The functionalities of the platform are the following. (For brevity, I will not describe the first model I worked on.)

- Define bodies/stages and associate one of them as attached to ground: In our case, the cage is attached to ground, and there's an intermediate suspension stage and the payload (optics breadboard).
- Give shape information, mass, moment of inertia, initial position values for each of the bodies.
- Setup wires for suspending one stage from another, and also input properties such as thickness, length, material.
- Define springs (in our case blade springs) and their properties (Q, spring constant, etc.)
- Setup other things, which are not relevant to our setup, such as inverted pendulum, heat links, dampers etc.
- The results include transfer function plots, and information about eigenmodes.
- 3D plots showing eigenmode motion of the system are also included in the results.

The parameters I used for simulating Crackle2 are described in Table 4.

The platform itself is quite intuitive, I do not delve into explaining how to use it in this report. I directly jump to the results.

Figure 24 shows Crackle2, as modeled on the platform.

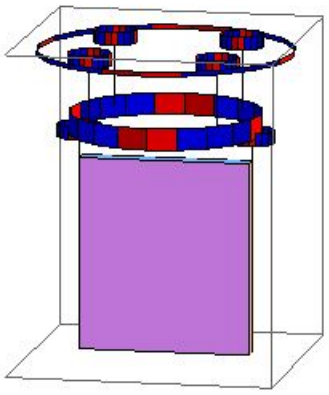


Figure 24: Crackle2, as built on the SUMCON platform

Table 4: Parameters used in Analytical Modeling on SUMCON

Parameter	**Description**	**Value**
M1	Mass of intermediate mass stage	20 kg
M2	Mass of lower mass stage	20kg
$I_{1,xx}$	Moment of Inertia of intermediate mass stage about x-axis	0.406 kgm^2
$I_{1,yy}$	Moment of Inertia of intermediate mass stage about y-axis	0.399 kgm^2
$I_{1,zz}$	Moment of Inertia of intermediate mass stage about z-axis	0.643 kgm^2
$I_{2,xx}$	Moment of Inertia of lower mass stage about x-axis	0.338 kgm^2
$I_{2,yy}$	Moment of Inertia of lower mass stage about y-axis	0.604 kgm^2
$I_{2,zz}$	Moment of Inertia of lower mass stage about y-axis	0.267 kgm^2
L_1, L_4	Lengths of suspension wires joining the intermediate mass stage to the cage	0.16 m
L_2, L_3	Lengths of suspension wires joining the intermediate mass stage to the cage	0.16 m
L_5, L_6	Lengths of suspension wires joining the lower mass stage to the intermediate mass stage	0.20 m
k_1, k_2, k_3, k_4, k_5, k_6	Spring constants corresponding to all suspension blades	1500 Nm^{-1}
$R_{1,x}$, $R_{1,y}$		0.34 m
d		0.07 m
g	Acceleration due to gravity	9.8 ms^{-2}

The reader can compare the shape of the transfer function and locations of poles and zeros from these plots to the experimental results. Since we are interested only in the "diagonal" transfer functions (i.e. X-sensing/X-excitation etc.), I have included analytical results of those alone.

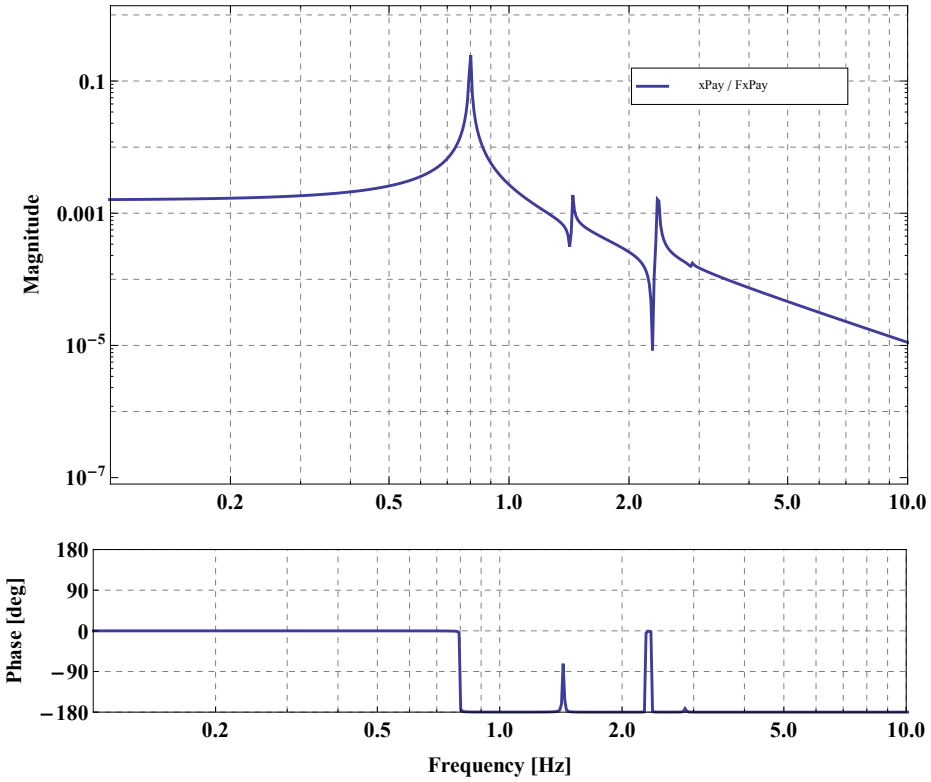


Figure 25: Analytical result for X-X transfer function

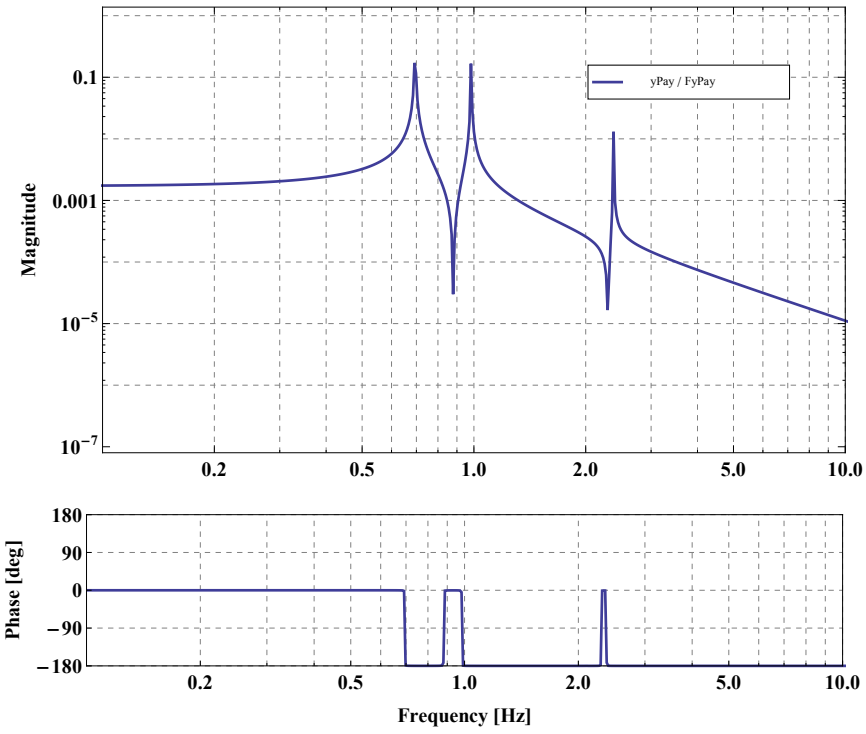


Figure 26: Analytical result for Y-Y transfer function

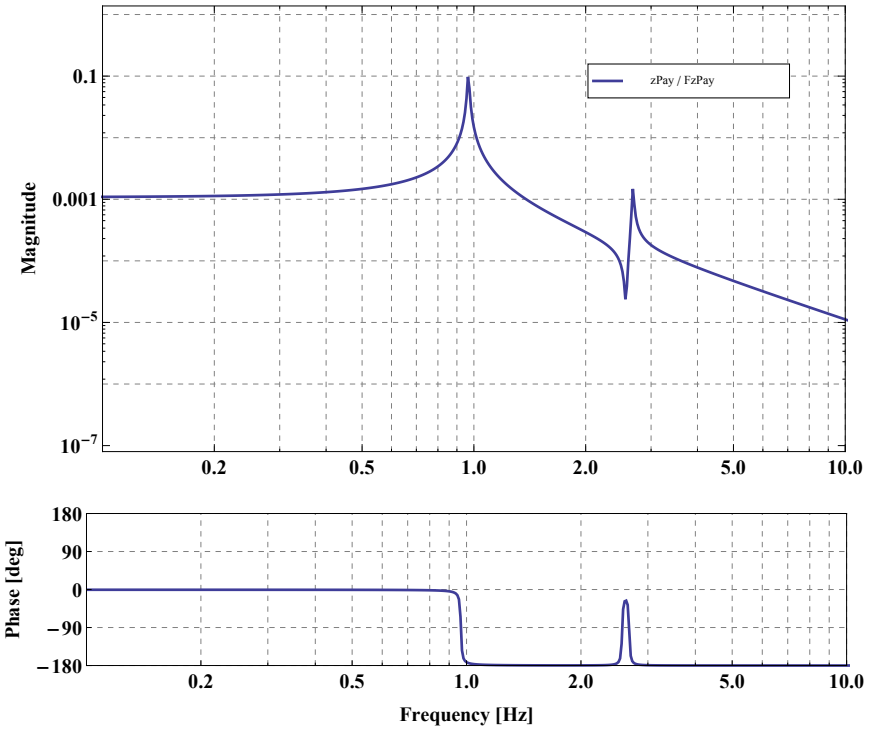


Figure 27: Analytical result for Z-Z transfer function

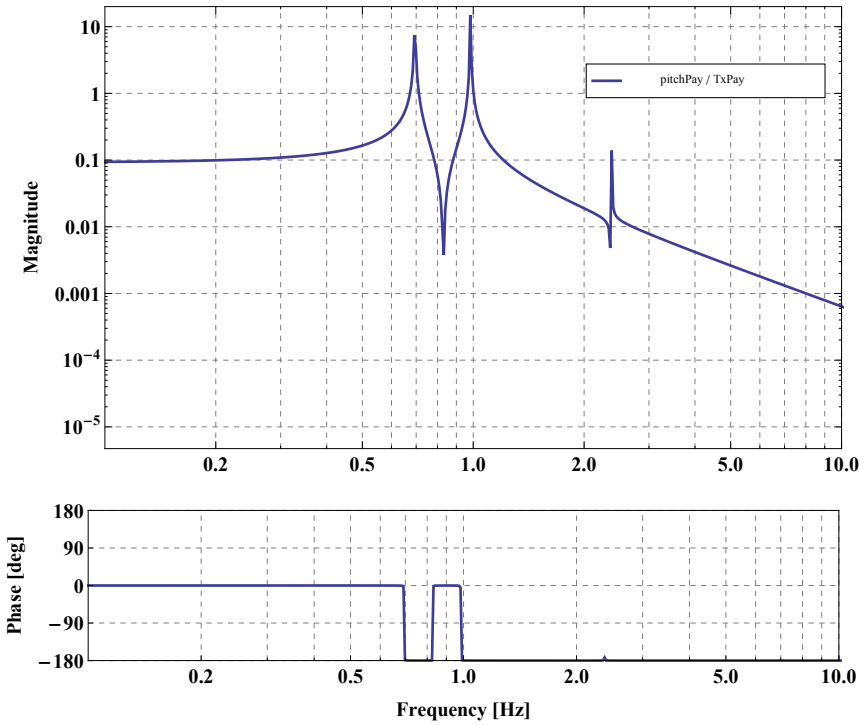


Figure 28: Analytical result for Pitch-Pitch transfer function

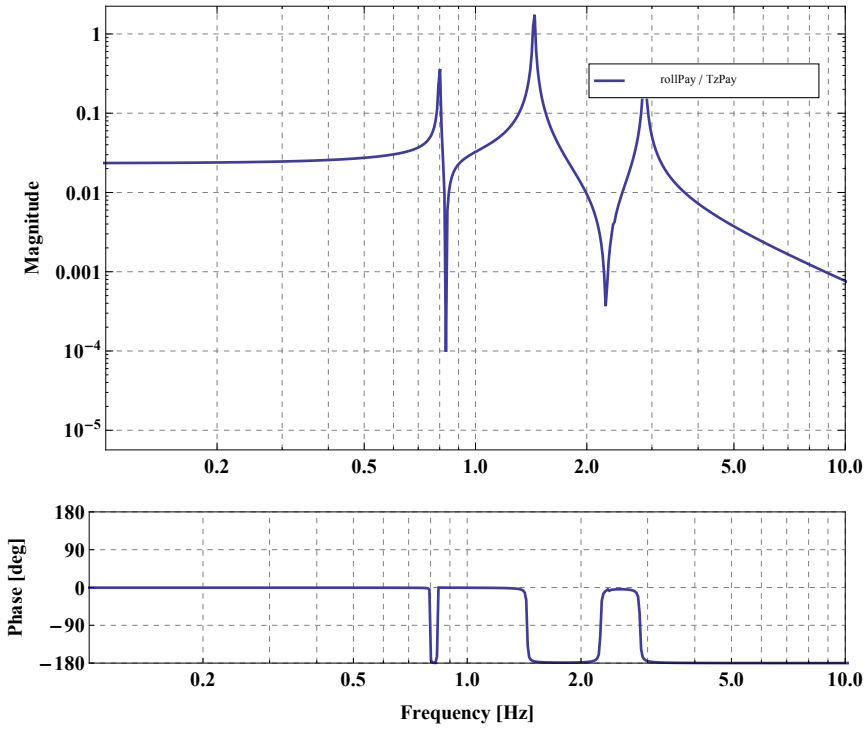


Figure 29: Analytical result for Roll-Roll transfer function

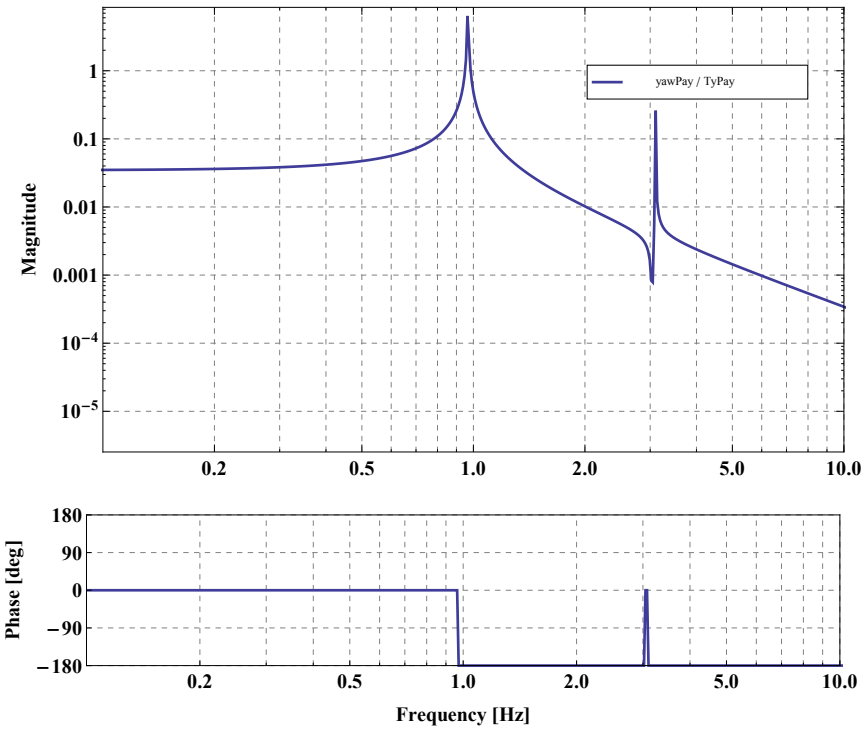


Figure 30: Analytical result for Yaw-Yaw transfer function

It is evident, upon comparing these plots with experimental results, that they are consistent. Even with no fine tuning of parameters in the analytical model, the results are quite similar in the pole-zero locations with the experimental results. A match to even this degree confirms to us that our measurements are valid, and that is all that we are interested in.

The next thing, therefore, was to fit these transfer functions into zpk models and use them to design feedback damping filters.

4.4.7 Vector-Fitting of Transfer Functions

Transfer function data is useful only upon vector-fitting to a zpk format. For this, I have used the **vectfit3** function in MATLAB. [4][5][6] The function works for both single transfer function as well as a vector of transfer functions.

The MATLAB code for fitting multiple TF's can be a bit messy, because it has a number of parameters to be set. I have written a wrapper function for convenient use.

For the purpose of illustration, I have included below in figure 30 the fit for just one transfer function: Z-Z transfer function.

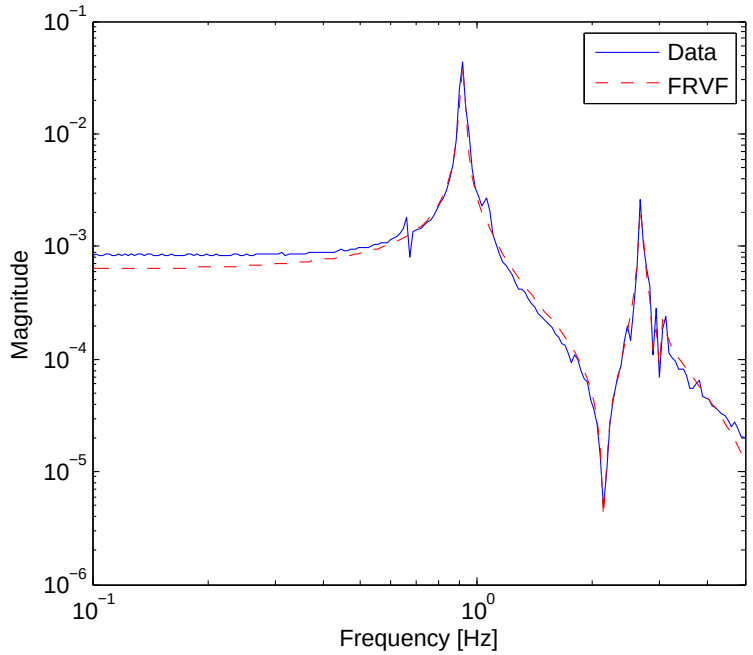


Figure 31(a): Vector fitting of Z-Z transfer function: Magnitude plot

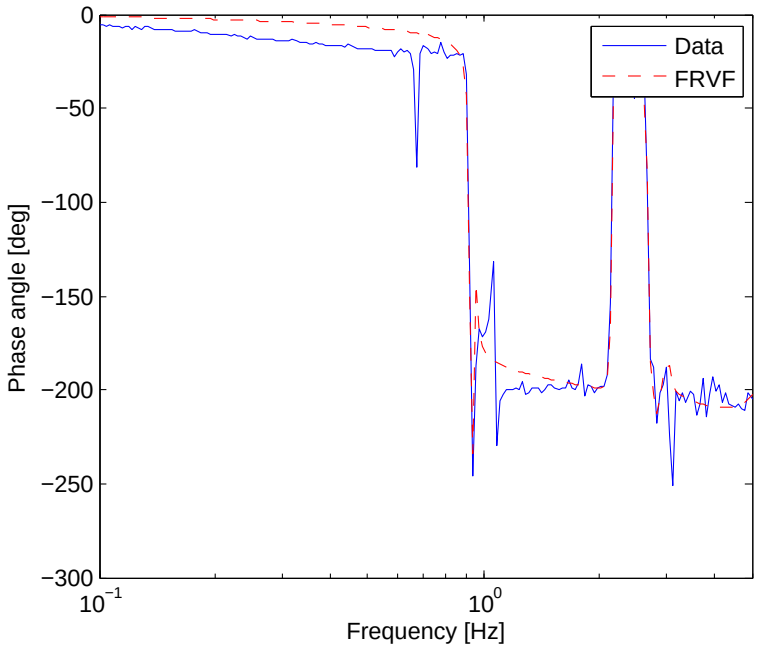


Figure 31(b): Vector fitting of Z-Z transfer function: Phase plot

The exact locations of the poles and other details are not of much interest to us here. These plots are for illustrative purposes alone.

4.5 Damping Filter Design

Having successfully measured transfer functions of the suspended optics breadboard, the next step was to design damping filters based on the measurements, and close the feedback loop by implementing the filters. In this subsection, I describe the logic and approach, and then list all of the filter designs.

4.5.1 The Approach

The problem statement is as follows: the transfer function consists of two poles (in most cases; around 1 Hz), and motion at these frequencies is to be reduced. That is, the Q of these poles is to be reduced. Also, the noise injected due to the compensator (filter) must be as less as possible at higher frequencies. More specifically, the damping noise in the 10's of Hz regime must be low as possible, since this is the range of interest in detection of Crackling noise.

The approach for filter design is as follows:

1. Since what we want to do is damping, the additional term that we must be an $\dot{x}$ term; where x stands for displacement. In the Laplace domain, this is an s: a Differentiator.

2. A differentiator alone would inject a lot of noise at higher frequencies, due to its monotonic nature. Therefore, it is needed to add one or two complex pole pairs, in order to roll off the compensator for frequencies well above the region with high motion. In this case, this means that the roll off must begin after a few Hz of frequencies.

One can always keep adding complex pole pairs to get better roll-off, but with faster roll-off, the unity gain frequency changes such that the Phase Margin drops. Having good Phase and Gain Margin is essential for the loop to remain stable, and therefore, one must also keep in mind this roll-off - stability trade-off while designing these filter.

4.5.2 The Design

I have used MATLAB's SISO (Single Input Single Output) Tool for designing these filters. I have assumed that all of the 6 degrees of freedom are decoupled – only then can I simply use the SISO tool – while in fact that is not the case! Because the suspended breadboard is "suspended" and not "free", there will be coupling in some of the degrees of freedom. A multiple-input-multiple-output (MIMO) control system is much more difficult to handle, and this single-input-single-output approximation is expected to hold in this case.

Also, it is interesting to note that the transformation to physical degrees of freedom is what made this possible: X and Y are (quite reasonably) decoupled, but OSEM A and OSEM B, for instance, are not decoupled! Remarkably, **if one were to use the OSEM signals as inputs and outputs, a rigorous MIMO control filter design would be necessary, but transforming to the physical d.o.f. makes the SISO approximation possible**!

Figures 32 show an example of the Open Loop Editor and Analysis plots of the MATLAB SISO Tool, for the Z-Z transfer function.

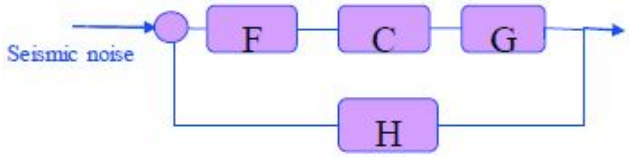


Figure 32(a): Control Architecture

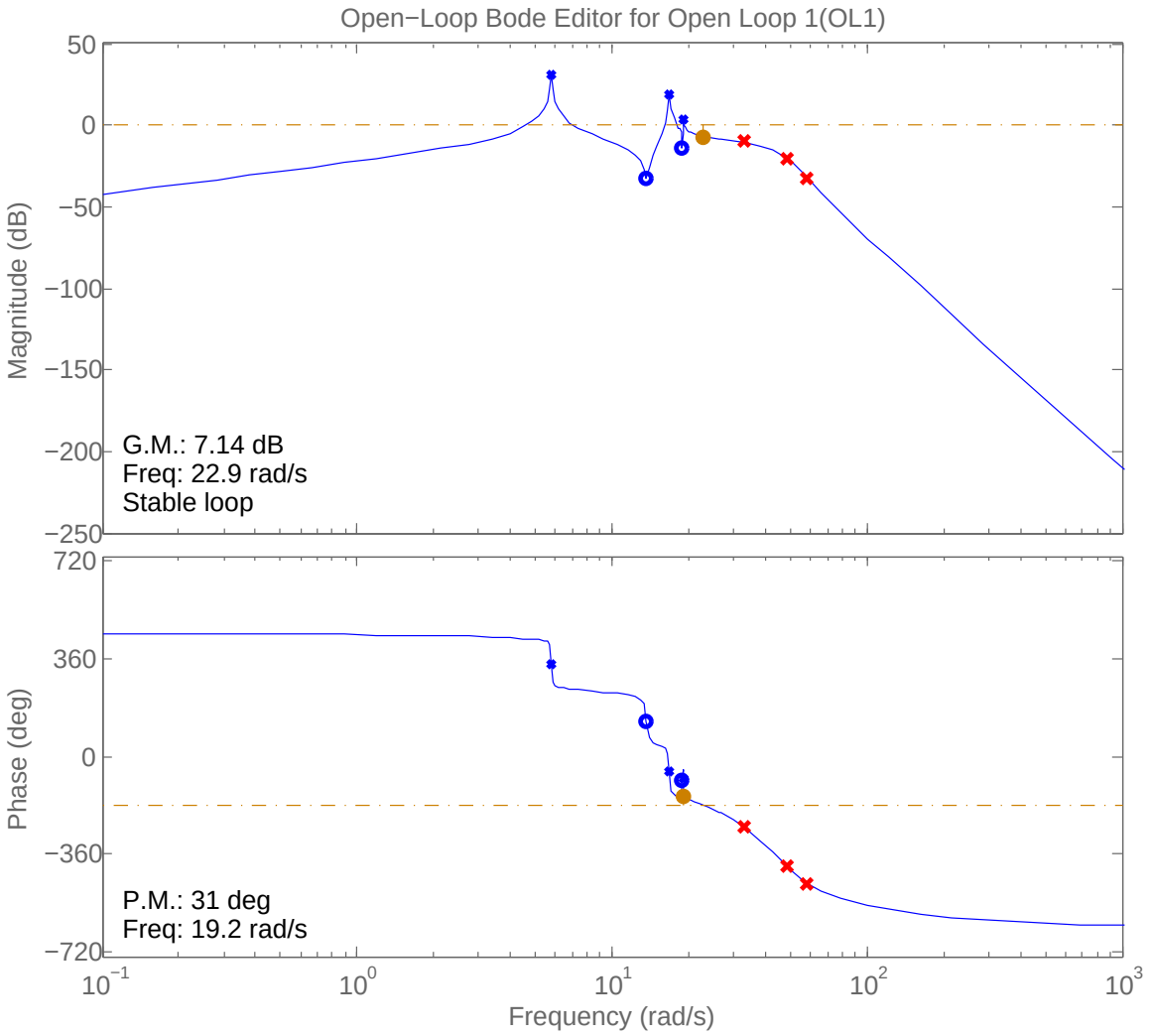


Figure 32(b): Open Loop Editor for Z-Z Open Loop Transfer Function (OLTF)
The plot in blue is the OLTF. In the magnitude and phase plots, one can notice red crosses depicting complex poles added for roll off. The Gain Margin and Phase Margin can be seen as well, and both the values are good enough for sustaining a stable loop.

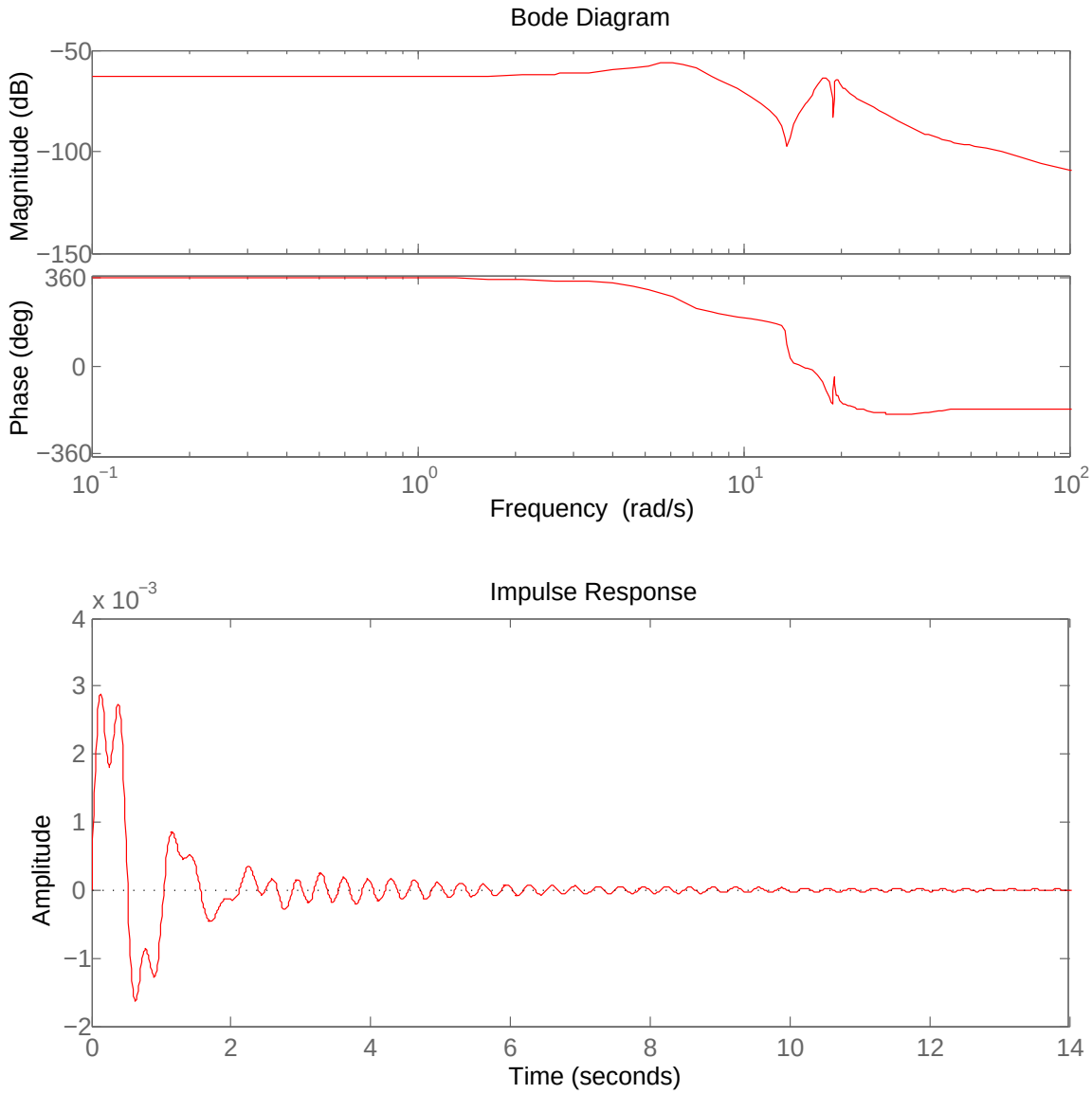


Figure 32(c): Analysis Plot showing Bode diagram of Closed Loop Transfer Function (CLTF) and Impulse Response of the closed loop

Notice in the Bode Diagram of the CLTF the reduced Q of the poles. Also, the impulse response of the closed loop system shows that it does not take very long for the system to settle back to stable position after an impulse.

Figures 33 through 38 show the filter designs for all six degrees of freedom. Each plot shows the Plant (breadboard) Transfer Function, the Compensator, and the Closed Loop Transfer Function. Notice the reduction in Q of the poles. In each of these designs, though not explicitly shown as in figures 32, I have taken care to have enough Gain and Phase Margins to ensure stability of the loops.

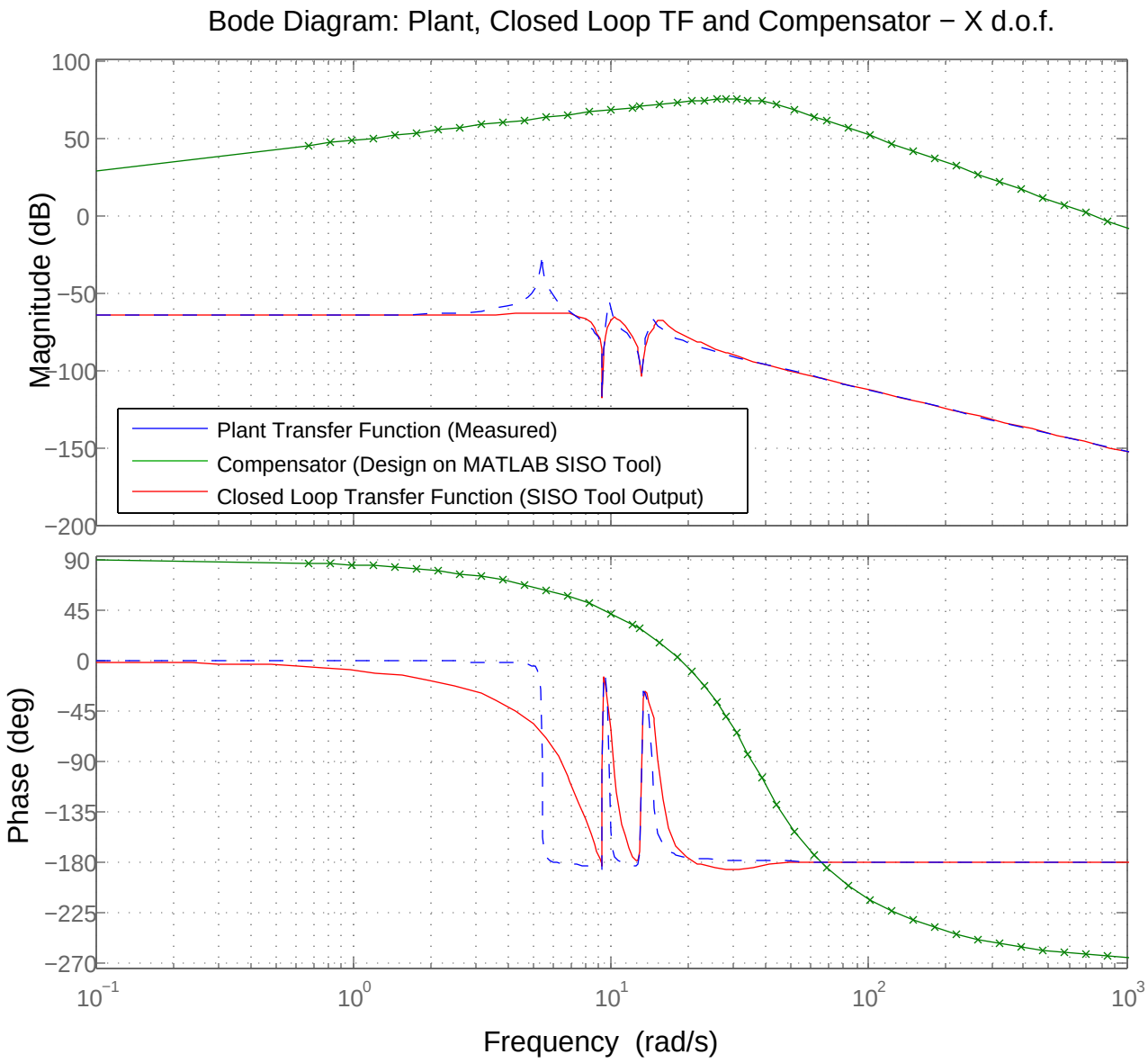


Figure 33: Filter design for X-X transfer function

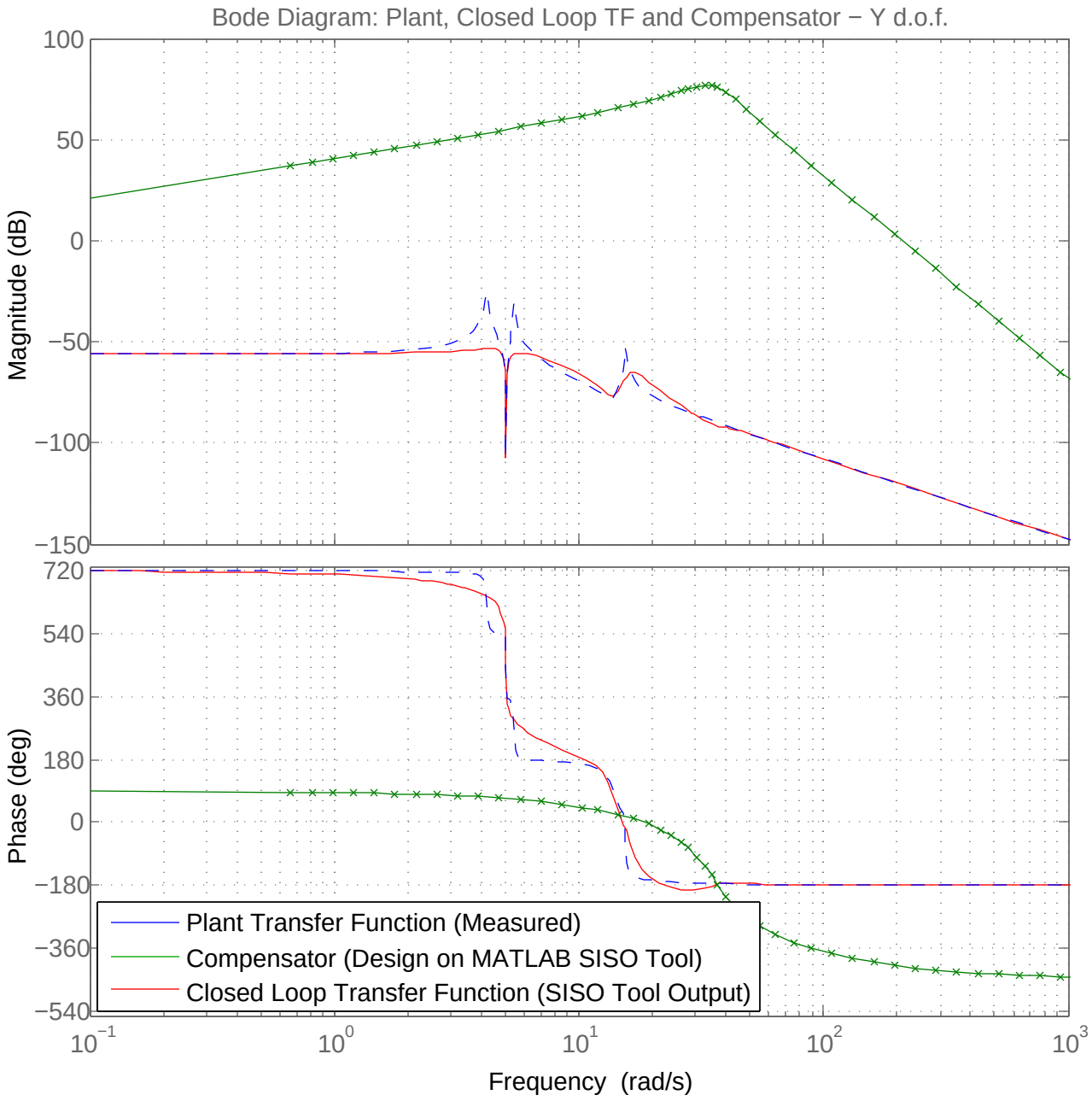


Figure 34: Filter design for Y-Y transfer function

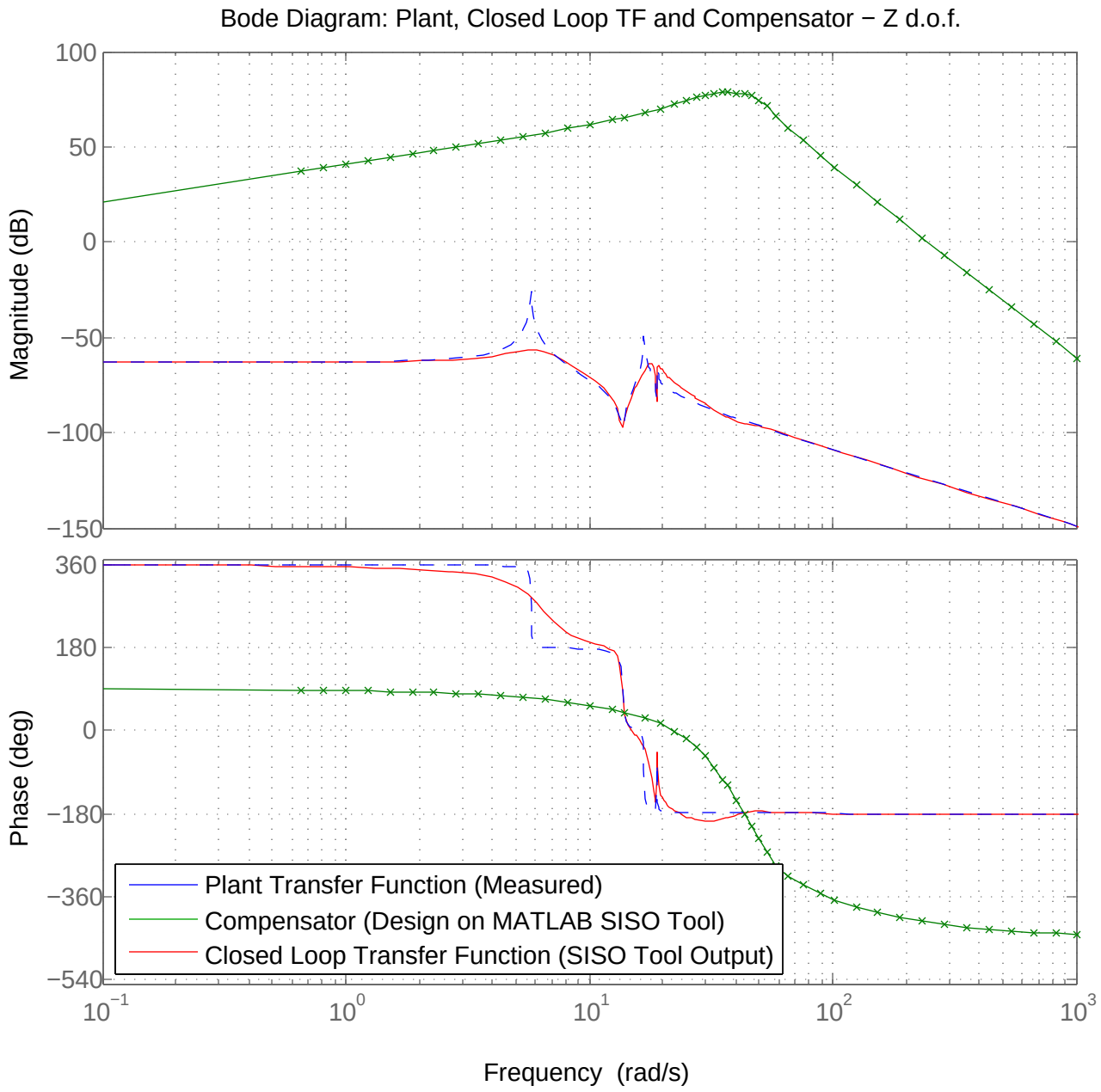


Figure 35: Filter design for Z-Z transfer function

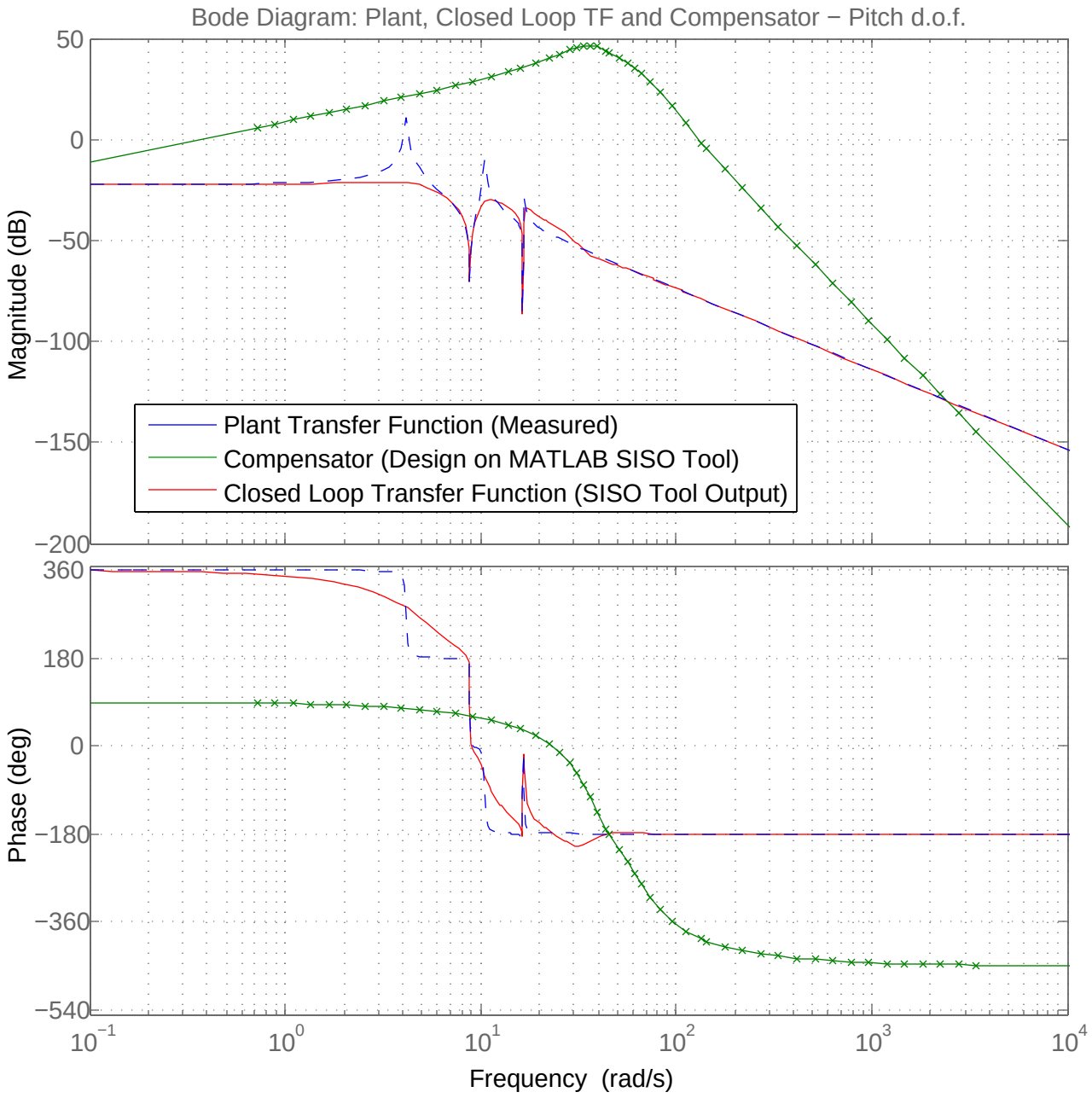


Figure 36: Filter design for Pitch-Pitch transfer function

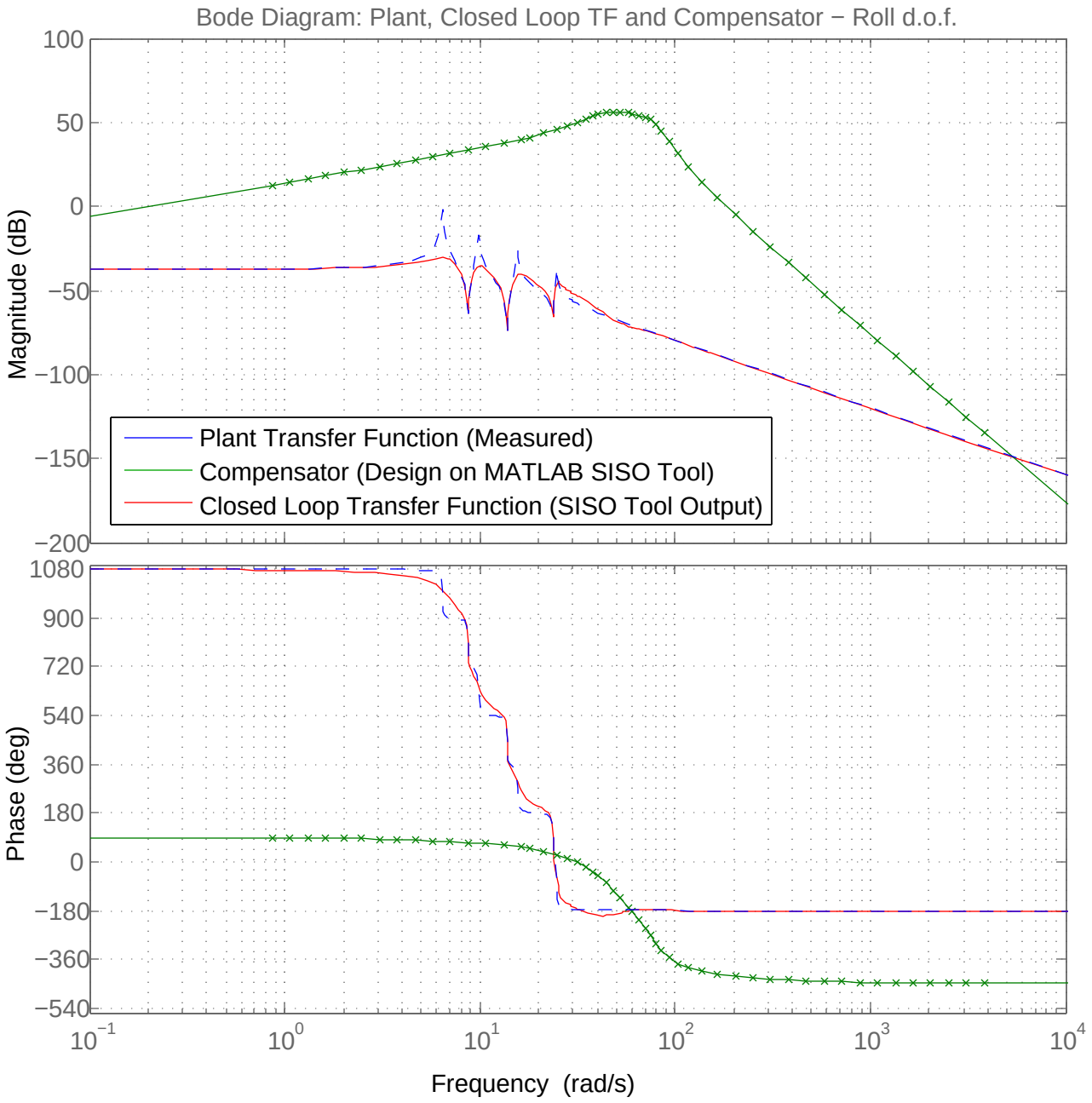


Figure 37: Filter design for Roll-Roll transfer function

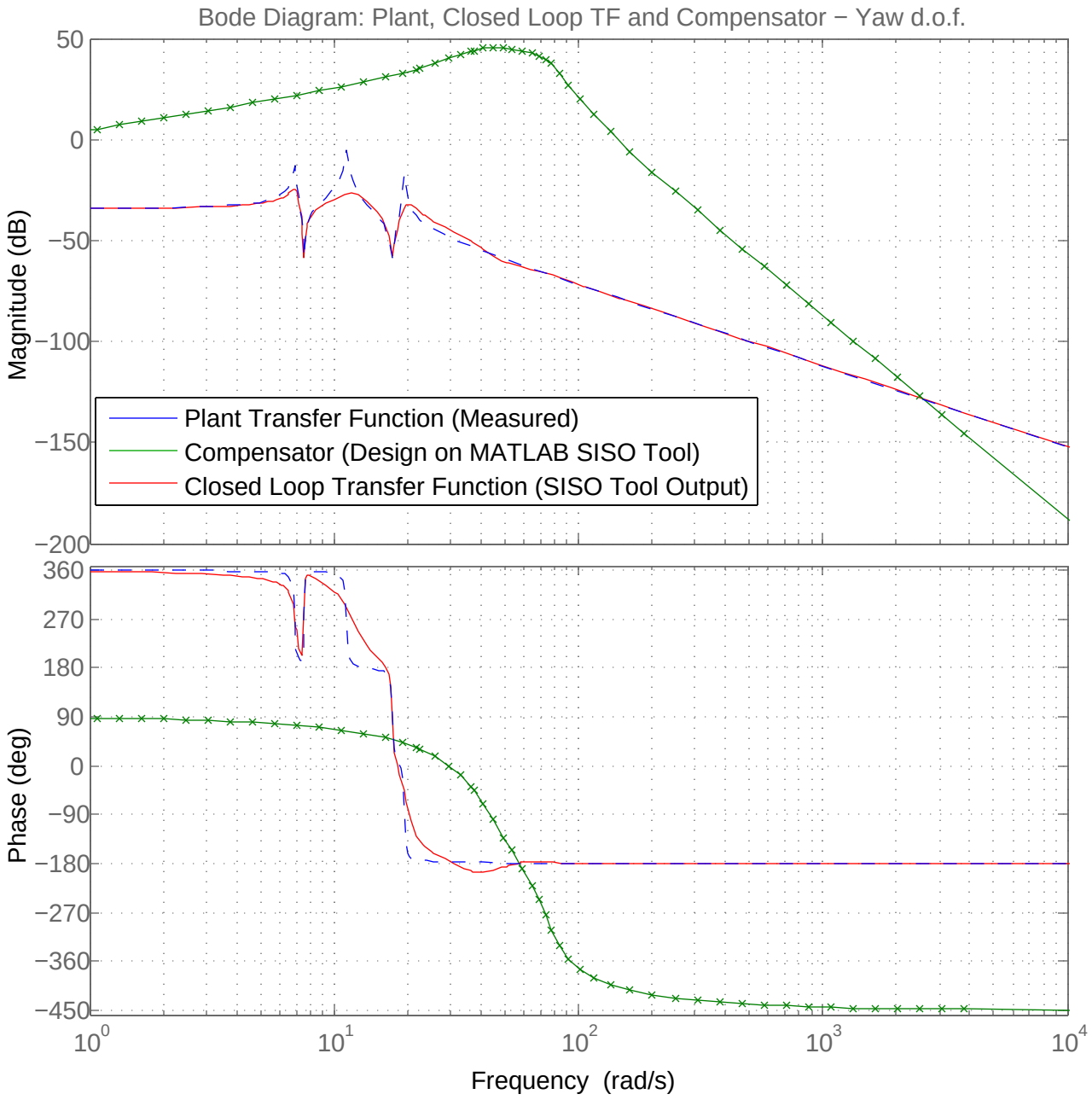


Figure 38: Filter design for Yaw-Yaw transfer function

4.6 Final Results

After designing the filters based on transfer function measurements, I then implemented the filters through the computer interface. Upon adjusting the magnitudes and signs of the gains in each loop and closing the loop, the feedback damping loops were in action and worked as expected.

Two plots summarize the results of the whole process: 1. Spectra of motion along all degrees

of freedom, and 2. Spectrum of the Michelson differential signal. The second of these is what ultimately matters, since that is what will be used to detect Crackling noise. Any excess noise introduced in the region of interest is bad. Let us take a look at both of the results.

4.6.1 Motion Spectra

Figures 39 through 44 show the spectra of motion along all physical degrees of freedom. In these figures, spectra for the damping OFF case are colored in Red, and for the damping ON case are colored in Blue.

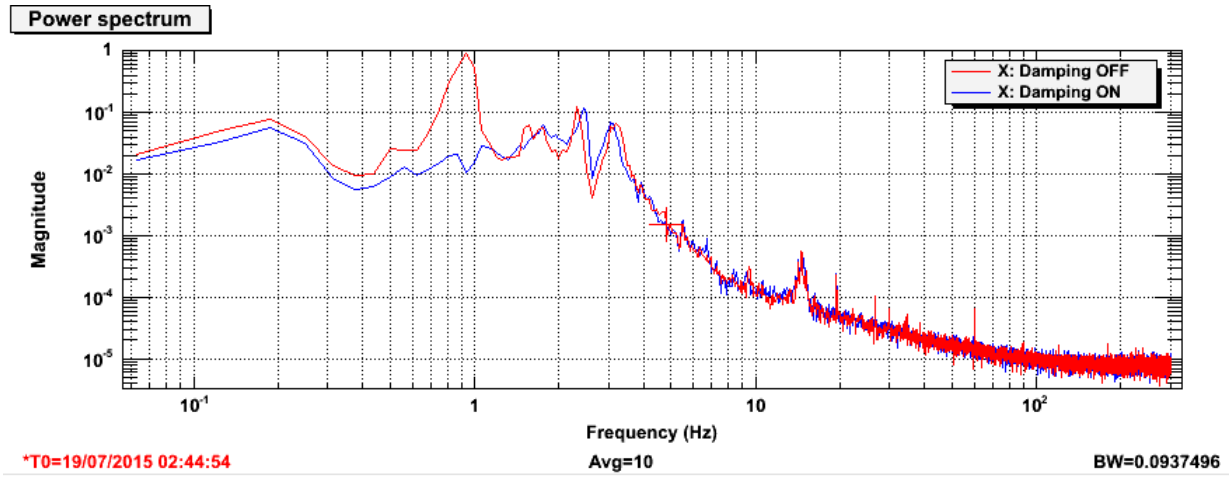


Figure 39: Spectrum of X motion: Before and after damping

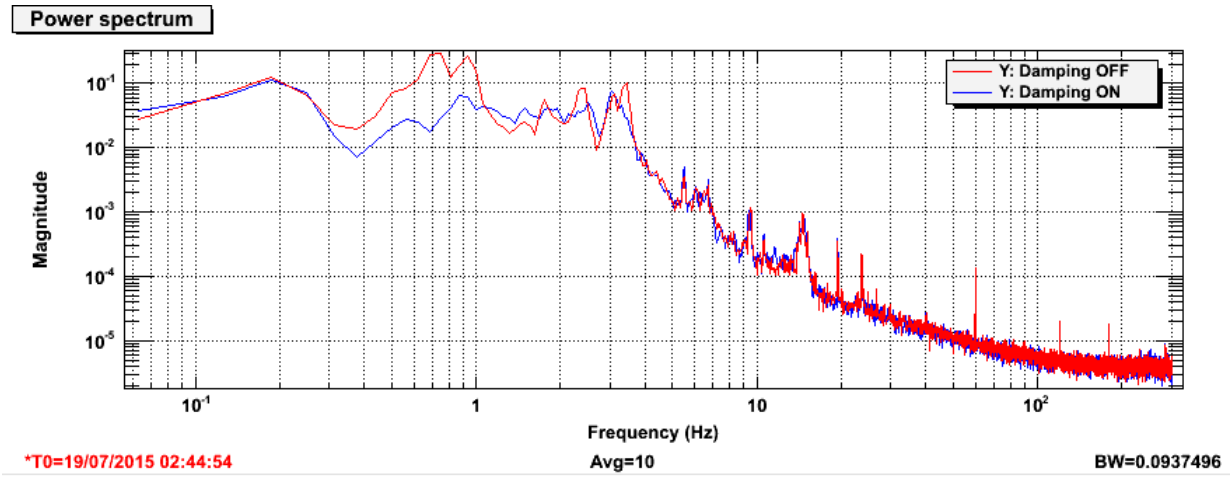


Figure 40: Spectrum of Y motion: Before and after damping

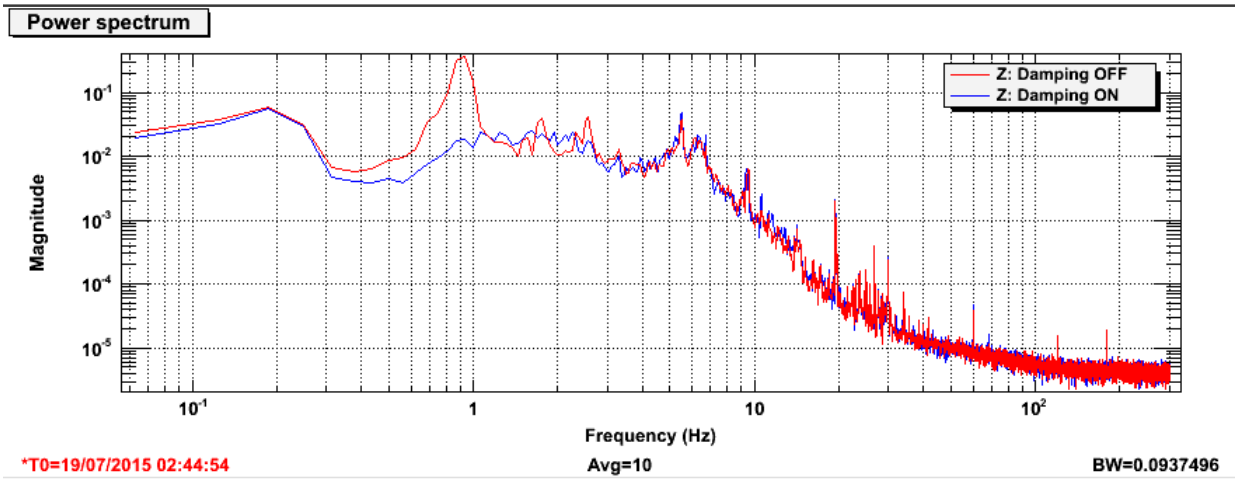


Figure 41: Spectrum of Z motion: Before and after damping

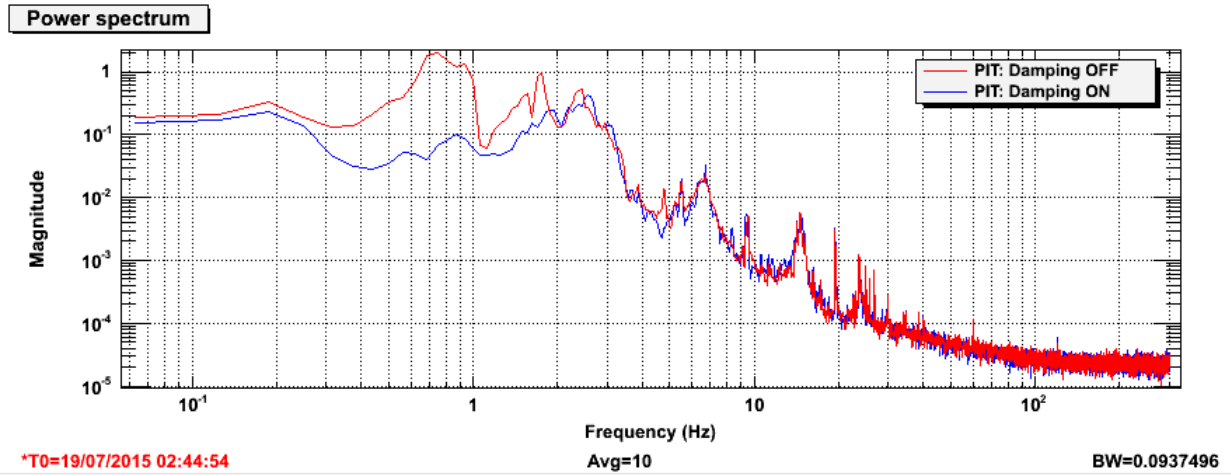


Figure 42: Spectrum of Pitch motion: Before and after damping

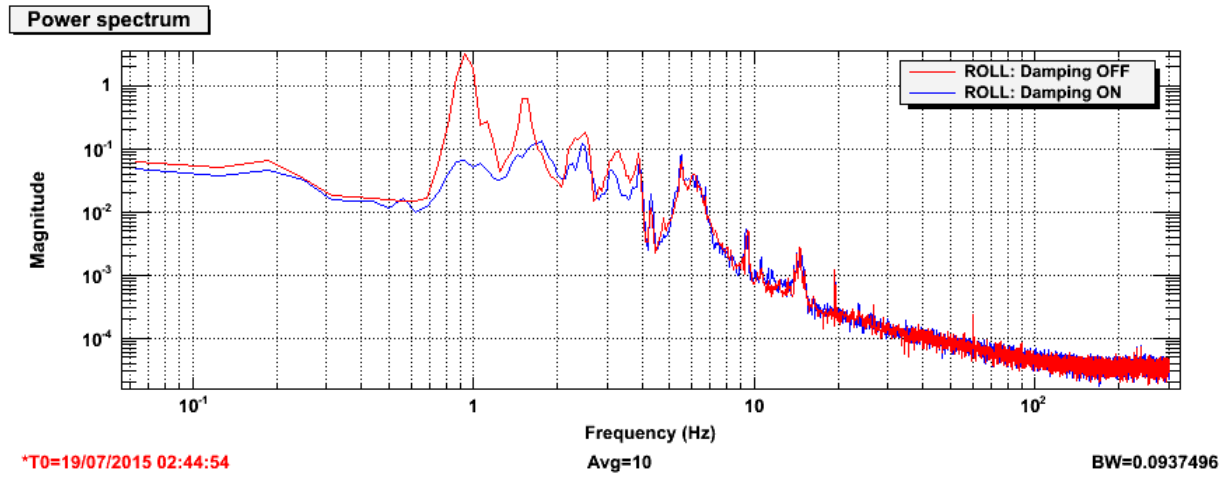


Figure 43: Spectrum of Roll motion: Before and after damping

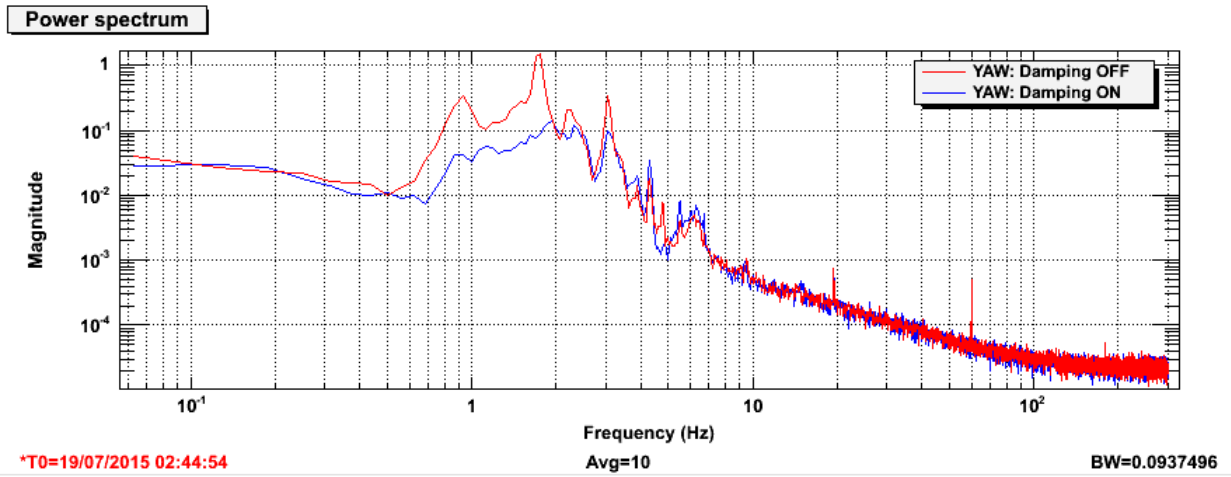


Figure 44: Spectrum of Yaw motion: Before and after damping

The striking feature of all of these plots is the reduction in Q of the peaks. This is exactly what the damping system was expected to do. Also, notice that at higher frequencies, as far as these motion spectra can show, there is no excess noise introduced by the damping. But, the resolution provided by the OSEM signals is only of the order of microns for a few Hz of frequencies. What matters is whether there is any new noise, in the region of interest for Crackle detection, injected in the Michelson differential signal.

4.6.2 Michelson Differential Spectrum

As discussed earlier, since Crackling noise occurs incoherently in each of the test blades, it is expected to show up in the differential signal of the Michelson interferometer. As a result, the spectrum of this signal is of interest to us.

Figure 45 shows the spectrum of the Michelson differential signal. In Red is the spectrum for the case with no damping, in Blue is for the case with the damping system I have described in this paper turned ON, and green is for the case with a preliminary damping system with ”local” sensor-to-coil feedback: i.e., sensor A to coil A feedback, and so on for all other OSEMs.

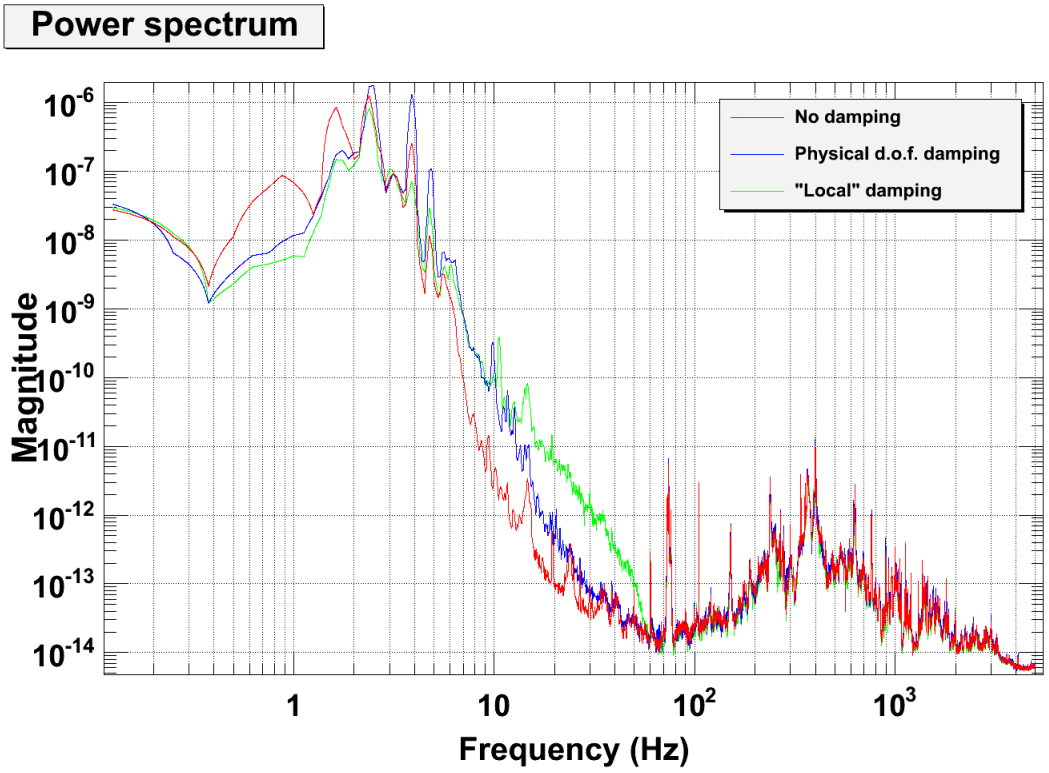


Figure 45: Spectrum of Michelson differential signal

Firstly, since the Michelson is locked, the breadboard motion peaks don't show up in this plot. What one must look for in this plot is the reduction or increase in sensitivity in the region of interest, i.e. 10's of Hz.

The "local" damping system clearly reduced sensitivity of the Michelson by about 2 orders of magnitude at some frequencies. That's too much noise when the ultimate target is a sensitivity of $10^{-15} m/\sqrt{Hz}$! The damping system I worked on and described in this paper gives much better results, improving the sensitivity by an order of magnitude. The reader must now be able to better see the importance of having a fast roll-off in the compensator: faster roll-off would mean lesser noise injection at higher frequencies.

In this case, the undamped case still looks better than the damped. However, it is important to realize the tricky trade-off: in the undamped case, there are high chances for the Michelson to unlock due to any spurious ground/horizontal table/setup motion! When damped, all of this is taken care of, but the sensitivity is compromised to some extent.

5 Conclusions

In this paper, I have described the control system I have designed to damp the motion of the suspended optics breadboard in Crackle2. I started off with sensing motion of the breadboard

using shadow sensor signals of OSEMs, reconstructed motion in physical degrees of freedom, and used the coils of OSEMs to drive the breadboard along various degrees of freedom; the stage was set for characterizing the mechanical response of the system by experimentally measuring transfer functions of actuation to motion. Results of the same were compared with analytical predictions and found to match.

Given the mechanical response, it was then possible to design filters for feedback, to damp motion of the breadboard at resonant frequencies. This was achieved by approximating the breadboard as a SISO system with six degrees of freedom. A filter was designed for each of the degrees of freedom and implemented. The resulting motion was characterized by observing the spectra of the six displacement signals. It was noted that the damping system satisfied the first of its requirements: reducing Q of the poles.

Finally, the spectrum of the Michelson signal indicated that there was improvement from the ”local” damping system, yet the undamped case showed better sensitivity. However, it is important to notice that in the undamped case, there are high chances for the Michelson to unlock due to any spurious ground/horizontal table/setup motion, while when damped, the breadboard is much more stable and lock is not lost.

It can be concluded that the control system developed as part of this project has successfully achieved the requirements set. However, there are a couple of minor improvements still possible.

6 Future Work

There are two subtle points that can be worked upon to improve the damping system.

1. In order to keep the sensitivity the same as well as prevent the breadboard from moving at lower frequencies, thereby ”protecting” the lock, a better filter design would be necessary. This filter must have a faster roll-off as well as good Phase and Gain Margins.

2. The second, though less feasible to spend time on than the first, is worth noticing. As mentioned before, treating our suspended breadboard as a SISO system is only an approximation. In fact, degrees of freedom are not completely decoupled. This can also be seen from the transfer function measurement results. Ideally, one must treat it as a MIMO system and design a MIMO feedback control system. This is a rather tedious and intimidating job, and one would choose this method only if absolutely necessary. Nevertheless, it is worth acknowledging this ”drawback”.

   Apart from these minor improvements, the OSEM damping system itself has successfully served its purpose.

References

[1] James P. Sethna, Karin A. Dahmen, and Christopher R. Myers, ”Crackling Noise”, *Nature*, 410, 242-250, 8 March 2001

[2] Xiaoyue Ni, E. Quintero, G. Vajente, ”Proposal for an upgrade of the Crackle experiment”, LIGO-T1400407-v1 (2015)

[3] E. Quintero, E. Gustafson, R. Adhikari, ”Experiment to investigate crackling noise in maraging steel blade springs”, LIGO-T1300465-v2 (2013)

[4] B. Gustavsen and A. Semlyen, ”Rational approximation of frequency domain response by Vector Fitting”, *IEEE Trans. Power Delivery*, vol. 14, no. 3, pp. 1052-1061, July 1999

[5] B. Gustavsen, ”Improving the pole relocating properties of vector fitting”, *IEEE Trans. Power Delivery*, vol. 21, no. 3, pp. 1587-1592, July 2006

[6] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, ”Macromodeling of Multiport Systems Using a Fast Implementation of the Vector Fitting Method”, *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 6, pp. 383-385, July 2008

[7] Bernard Friedland, ”Control System Design: An Introduction to State-Space Methods”, *Dover Publications Inc.*, 2005

7 Acknowledgements

SURF with the LIGO Laboratory group at Caltech has made for one of the best summers I ever had. I would like to take this opportunity to thank everyone who has contributed to making this happen.

Keeping with the Indian tradition, I would first like to thank my parents for having supported me throughout, for having staying with me during my ups and downs. It is their hard work and sacrifice that has made all of this possible for me. I will remain indebted to them forever.

My mentor, Dr. Gabriele Vajente, has been an amazing guide, mentor and friend. The interactions that I had with him have contributed greatly to my learning, and I hope to one day become as good a Controls Engineer as he is!

Xiaoyue Ni (whom we call the Glue Master!) and her help has been crucial to my project. It is her inputs that made it possible for me to get a good start to my work.

Prof. Rana Adhikari and his group have been a great source of inspiration for me throughout. I love experiments, and I got to experience with this group some of the finest cutting edge experiments around; the exposure I got through this has been immense. I am proud to have been part of this group for the summer.

I would also like to thank Prof. CV Krishnamurthy, Dr. Deepa Venkitesh, Dr. Manu Jaiswal, and Dr. L Sriramkumar, my faculty guides at IIT Madras, for having supported me through the internship selection process, and giving me valuable advice to make the best of my summer.

I have used platforms or codes developed by folks at the LIGO Observatories, KAGRA, and MIT. Their inputs have proved invaluable for this project.

My interactions with other SURF students at Caltech have given me a lot to learn. I carry with me memories of spending an incredible summer at Caltech with all of them.

This project has been made supported by and made possible by the following organizations. I would like to thank them for the support and facilities provided.

- National Science Foundation (NSF),
- California Institute of Technology (Caltech),
- Indian Institute of Technology Madras (IITM),
- Laser Interferometer Gravitational Wave Observatory (LIGO),
- Indian Initiative for Gravitational Wave Observations (IndIGO).