

# Notes on LIGO detectors' calibration

Gabriela González  
Louisiana State University

September 8, 2002

## 1 Introduction

These notes are intended to help clarify some issues about using the calibration for LIGO data. These issues came up in conversations with different people, so I am addressing some particular concerns; however I am NOT intending to make these notes a comprehensive discussion of the topic.

## 2 The DARM loop

I will follow closely the nice presentation that Peter Shawhan made at the March'02 LSC meeting: *An Overview of LIGO Length Sensing and Control*, G020064 (the document can be looked up in the LIGO-LSC web site). In his notation, we have a sensing function  $C(f)$ , a servo gain function  $G(f)$  and an acuation function  $A(f)$ . Each of these functions is complex, and has units that depend on the details of the definitions; typically the sensing function is in counts/m (for AS\_Q response to a change in distance), the servo gain function is in counts/counts (DARM\_CTRL/AS\_Q), and the actuation function is in m/count. The frequency dependence of the sensing function is usually just a "cavity pole" from the Fabry-Perot cavity in the arms; the shape of the actuation function is also usually simple, like a pendulum transfer function. These functions are measured, and does not change often, except by overall gains. The shape of these functions is similar in all detectors, although the overall gain of the functions may be different. The servo gain function is usually complicated, and is different in every detector, but is well known, since it is a software code that produces the output (DARM\_CTRL) from the input (AS\_Q). The overall gain of the servo function is constant and the overall gain of the actuation function should be constant; however the gain of the sensing function is dependent of the power buildup in the FP cavities, which in turn depends on alignment: this gain drifts within a locked segment, and can change significantly from segment to segment.

The open loop gain, sometimes just called loop gain, is the product of all three functions,  $H=GAC$  (the open loop gain is more often called "G", but since I am following Peter's notation and he used G for the servo function,

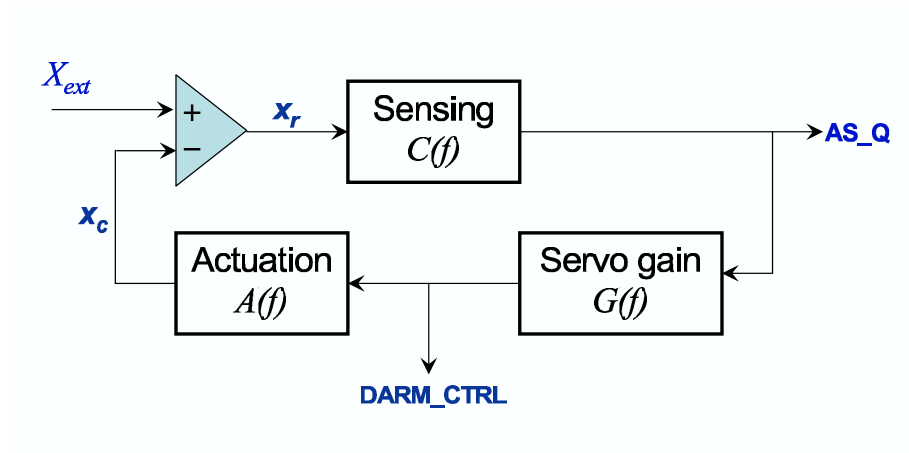


Figure 1: Length servo loop, from Peter Shawhan's presentation G020064

I used the next letter available). The open loop gain is dimensionless. The frequency at which the magnitude of the open loop gain is one is called the “unity gain frequency”, or ugf. The frequency band where the magnitude of  $H$  is high (typically low frequencies) is called the “servo band”. In LLO, the ugf is about 200 Hz, but of course it changes when the gain goes up or down with alignment.

The math of the response of different signals to excitations in different places is simple, and is always related to the open loop gain. For example, given an externally induced change in distance  $X_{ext}$  (by seismic noise or a gravity wave):

$$\begin{aligned}
 \text{control length:} & & x_c &= A(f)DARM\_CTRL \\
 \text{control signal:} & DARM\_CTRL & &= G(f)AS\_Q \\
 \text{error signal:} & AS\_Q & &= C(f)x_r \\
 \text{residual length:} & x_r & &= X_{ext} - x_c \\
 & & &= X_{ext} - A(f)DARM\_CTRL \\
 & & &= X_{ext} - A(f)G(f)AS\_Q \\
 & & &= X_{ext} - A(f)G(f)C(f)x_r
 \end{aligned}$$

From going around the loop, we deduce

$$\begin{aligned}
 x_r &= X_{ext} \frac{1}{1 + H(f)} \\
 x_c &= X_{ext} \frac{H(f)}{1 + H(f)} \\
 AS\_Q &= X_{ext} \frac{C(f)}{1 + H(f)}
 \end{aligned}$$

$$DARM\_CTRL = X_{ext} \frac{G(f)C(f)}{1 + H(f)} = X_{exc} \frac{1}{A(f)} \frac{H(f)}{1 + H(f)}$$

At low frequencies where the gain is very high,  $H(f) \gg 1$ , we see that the residual length is much smaller than the exciting motion  $x_r \ll X_{exc}$ , while the control length is very similar to the exciting motion,  $x_c \sim X_{ext}$  (that's how the motion is cancelled, of course). At high frequencies, where the gain is low,  $H \ll 1$ , the opposite is true: the residual length is close to the exciting length,  $x_r \sim X_{exc}$ , while the control length is much smaller,  $x_c \ll X_{exc}$ .

### 3 Notation for poles and zeros

A detail worth mentioning, since it is cause of confusion, is that often we think of these transfer functions as functions of frequency, as representing the response of one point to another driven with a sine wave. However, when designing servo filters or when plotting them in Matlab and other control system tools, it is more useful to work in the Laplace domain, where the frequency in the Fourier domain is related to the Laplace variable through  $s = i\omega = 2i\pi f$ . Yet a different convention is the one used in DTT... all this means that when giving a list of poles, zeros, and gains for a transfer function, we need know exactly what space we are working in.

Given  $x(t)$ , the formulas for Fourier and Laplace transform are:

$$x(\omega) = \int_0^{+\infty} x(t)e^{-i\omega t} dt$$

$$x(s) = \int_0^{+\infty} x(t)e^{-st} dt$$

In general, we think of the Fourier frequency  $\omega = 2\pi f$  as a real variable (since we assume it is the setting in a signal generator), while the Laplace variable  $s$  is complex, and often written as  $s = \sigma + i\omega$ , so that  $x(\omega) = x(s)|_{\sigma=0}$ .

We usually deal with transfer functions, so that if  $y(t)$  is obtained by a linear, causal operator on  $x(t)$ , then  $y(f) = T(f)x(f)$ , and  $y(s) = T(s)x(s)$ . Notice that while the Fourier transforms of  $x(t)$  or  $y(t)$  change in amplitude if you want to use  $\omega$  in radians/sec or  $f = \omega/2\pi$  in Hertz, the transform function does not:  $T(\omega) = T(f)$  (this is not true for the signals themselves:  $x(\omega) \neq x(f)$ ). In general, the Fourier transform of a time series is a complicated function (not always well defined), but the Fourier representation of transfer functions are usually (but not always) simple polynomials. This is where the "zero,pole,gain" notation becomes very handy. In general, if we describe a transfer function in the Laplace domain with poles  $p_i, i = 1..N_p$ , zeros  $z_i = i = 1..N_z$  and gain  $k$ , then we understand that the polynomial is

$$T(s) = K \frac{\prod_1^{N_z} (s - z_i)}{\prod_1^{N_p} (s - p_i)}$$

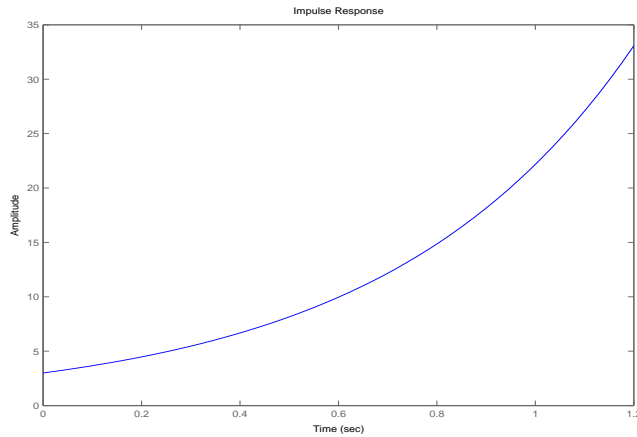


Figure 2: Impulse response obtained in matlab for `zpk(1,2,3)`

If the transfer function applies to signals that have a Laplace transform (that is, causal and integrable), it means the poles  $p_i$  are all in the left hand plane: they have a negative real part. It also requires that if poles are complex, they appear in conjugate pairs. In the Fourier domain, the “left hand side” rule means, roughly, that exponentials (described as complex poles  $\omega_p = \omega_0(1+i/Q)$ ), decay exponentially instead of growing unbounded.

Most control systems design work in the Laplace domain, because intuition is developed for these rules, and make trouble-spotting easier. For example, in Matlab, if you type

```
>>zpk(1,2,3)
```

you get

```
Zero/pole/gain:
3 (s-1)
-----
(s-2)
```

and if you type

```
impulse(zpk(1,2,3))
```

you get the plot in Figure 2, showing an unstable filter, since we have a pole in the “right hand side”.

When one converts from Laplace domain to Fourier domain, we just remember  $s = i\omega$  and convert poles and zeros:  $p_i$ (in FD) =  $-ip_i$  (in LD), so:

$$T(\omega) = K \frac{\prod_1^{N_z} (\omega - z_i)}{\prod_1^{N_p} (\omega - p_i)}$$

We also usually list poles and zeros in Hertz, and then write the above polynomial for  $T(f)$  using  $\omega = 2\pi f$ , of course.

A yet different notation that makes easier to get the plot of the transfer function in your head without using Matlab or anything else, is to use scaled variables, so that the “gain” for the polynomial is in fact the “DC gain”<sup>1</sup>. This is the notation used by Daniel Sigg in his documents about the plant transfer functions, and by DTT, and it is a mixed notation with Laplace variables in Hertz, with a change in sign (sigh!):

$$T(s) = K_{DC} \frac{\prod_1^{N_z} (1 + s/2\pi z_i)}{\prod_1^{N_p} (1 + s/2\pi p_i)}$$

In this notation, for example, a pendulum pole pair at 0.76Hz with a Q of 10 would be  $.076 \pm i0.76$  (and confuse everybody who doesn’t know the notation because it is on the right hand side...).

## 4 A real example

To show how the previous concepts work together, I put here how we use AS\_Q and DARM\_CTRL to get residual and control lengths using simple DTT calibrations.

Often, we plot the residual length and the control length, using AS\_Q and DARM\_CTRL with a simple cavity pole calibration for AS\_Q and a pendulum calibration for DARM\_CTRL: we are plotting  $x_r = AS_Q/C = 1/(1 + H)$  and  $x_c = ADARM_CTRL = H/1(1 + H)$ . This is shown in Fig3, taken at LLO during S1.

The calibration used in Fig3 for AS\_Q is  $1/C(f) = 3 \times 10^{-15} \text{m/ct}/(1 + i88\text{Hz}/f)$  (the cavity pole is at 88 Hz), and for DARM\_CTRL is  $A(f) = 1.72 \times 10^{-9} \text{m/ct}/((1 - 0.76\text{Hz}/f)^2 + 0.1^2)$  (the pendulum is at 0.76 Hz, and we assume a Q=10). This calibration would be entered in DTT as (k=3e-15, z=88) to convert AS\_Q into residual length, and (k=1.7e-9, p=0.076 0.76) to convert DARM\_CTRL into control length.

Notice the high gain at low frequencies, where the input motion is suppressed by 10 orders of magnitude, and the unity gain frequency around 200 Hz where the two curves (residual and control lengths) cross each other.

The envelope of the residual and control lengths (the larger of the two) is an approximate measure of the exciting motion, *except* near the unity gain frequency. If  $|H| \sim 1$ , then the magnitude of  $1/(1 + H)$  depends critically on the phase of  $H$ . In particular, if the phase is close to 180 degrees,  $1/(1 + H)$  gets very

---

<sup>1</sup>This only works if we don’t have any pole at zero frequency (which would have infinite gain at zero frequency). We do sometimes use poles at zero, these are simple integrators and are useful. I think there are no poles at zero in any of the transfer functions needed for the DARM loop.

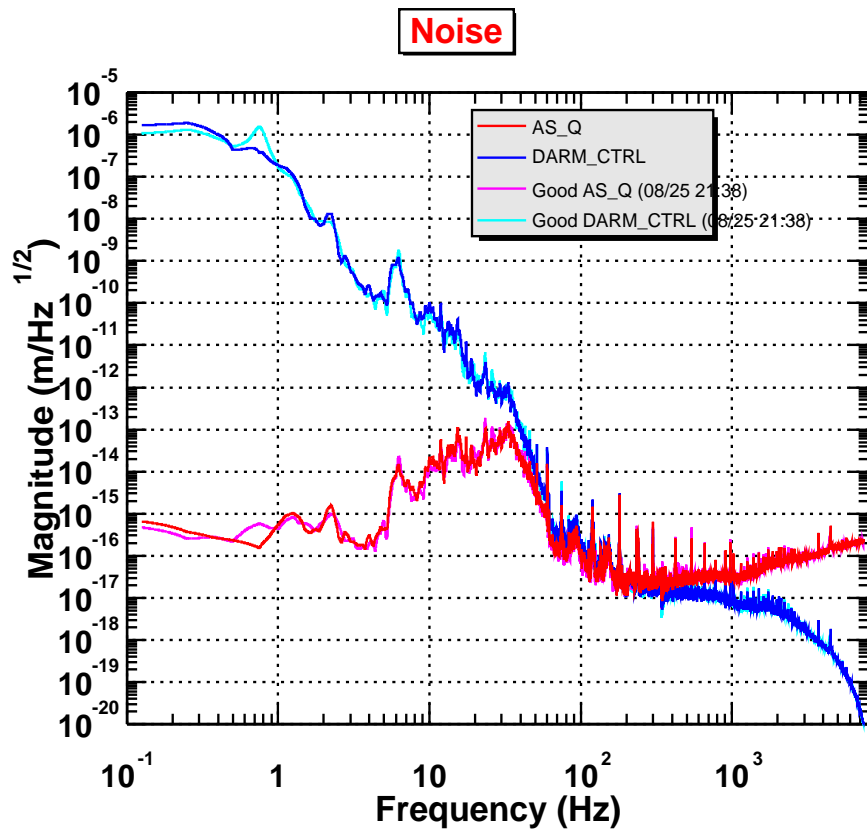


Figure 3: A measurement of input length noise, using AS\_Q and DARM\_CTRL.

large: this is called a *gain peaking*. Both AS\_Q and DARM\_CTRL will get have “bumps” near the unity gain, but the actual exciting motion is not as large as these plots would indicate. The difference in degrees between the phase of  $H(f)$  at the ugf and 180 degrees is called *phase margin*, and a loathy goal is to have it near 45 degrees (the best phase margin is 90 degrees). If the phase margin is 45 degrees at the ugf, then  $|1/(1+H)| = 1.3$ , so both A DARM\_CTRL and AS\_Q/C (which have equal magnitude at the ugf) overestimate the exciting motion by 30% (that is, the noise is even better than the one plotted in the figure). If the phase margin is 90 degrees, the exciting motion is underestimated by 30%.

In general, loop shaping used for  $G(f)$  uses real poles and zeros for simplicity and stability. However, when using sharper low or high pass filters (Butterworth, Chebyshev, etc), they have complex poles and zeros. Notches (like the ones used for violin modes), and high Q pendulums, are also necessarily complex, as are often used “resonant gain stages”. If we don’t care much about phase, we sometimes replace a pendulum pole pair (complex) with two real poles at the pendulum frequency: this trick allows for simplicity in reporting transfer functions, and is reasonable faithful to magnitude and phase as long as we are far from the pendulum frequency itself.

## 5 Calibration

As explained in “The Mechanics of an Engineering Run Amplitude Calibration”, LIGO-G020067, the calibration procedure involves two steps: first, calibrating the response of a drive signal on the ETM mirrors; and second, a swept sine measurement of the response of AS\_Q to that calibrated drive signal. In the language of this note, the first step is measuring the actuation function  $A(f)$ , while the second step is measuring the function  $SS = AC/(1+H)$ . The response function we need is that of AS\_Q to a gravity wave excitation  $X_{exc} = Lh$ , or  $T = C/(1 + H) = SS/A$ , so we can get it from the two measured functions. In order to get a formula for the transfer function, however, rather than a table of measured points, two different methods are often used:

- A fit is done to the transfer function  $SS$ , in some chosen frequency domain (within the measured range, of course), assuming some maximum number of poles and zeros, and limiting the numbers to real poles and zeros for simplicity. The frequency dependence of  $A$  is known, and its gain is measured, so a functional form for  $T = SS/A$  is obtained.
- Since the frequency-dependence of the functions  $A, C$  and  $G$  is well known, the gain of  $A$  is measured, and the gain of  $G$  is exactly known, the swept sine is compared to the the expected function, and used to get the single missing parameter, the gain of  $C$ . We than have a functional expression valid at all frequencies for each of  $A, C$  and  $G$ , and therefore for  $T = C/(1 + H)$ . This function is necessarily complicated, since it has all the well known details of  $G$ , but it can be simplified in a given frequency range if details outside the range are not needed.

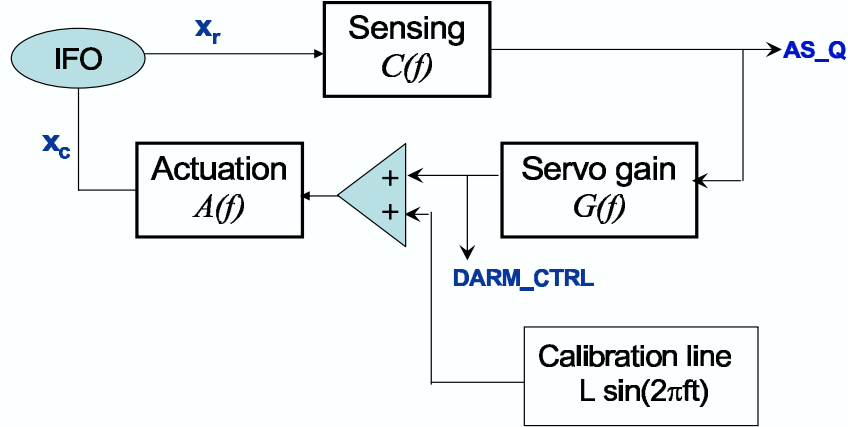


Figure 4: Calibration lines inserted in the DARM loop.

In both cases, the calibration is valid for the time at which it was done. The gain of  $A$  is not expected (or observed) to change, but the gain of  $C$  changes often (and continuously) due to alignment in the cavities. In order to use the calibration done and apply the change in gain, we need to do two things: have calibration lines in the spectrum; and do the math to get a new transfer function.

Two calibration lines are inserted, added to DARM\_CTRL, one at a low frequency ( $\sim 50$  Hz), in the servo band, and another at a high frequency ( $\sim 900$  Hz), outside the servo band. Let us call the added signals, in counts,  $L_i$ . I have drawn the servo diagram again in Fig4. For both lines (or any other signal or noise added to DARM\_CTRL), the observed amplitude in DARM\_CTRL is  $L * H / (1 + H)$ , and in AS\_Q is  $L * AC / (1 + H)$ . If the gain is high (low frequency line), then DARM\_CTRL is approx  $L$ , and AS\_Q is approx  $L/G$ . If the gain is low (high freq line), then DARM\_CTRL is  $LACG$ , and AS\_Q is  $LAC$ . So, from the two lines we get enough information to get the overall gain of the functions  $AC$  and  $G$ . In practice, we use the change in the high frequency line to scale  $C$  with it.

The lines give us information on the overall gain change of the *open loop gain*  $H=ACG$ . With this, we can scale the response function  $T = C/(1 + H)$ , but it is **not** an overall gain change in the transfer function. If a transfer function  $T_0$  was obtained at some time when the calibration was known, and then the gain in the cavity is observed to change by a factor  $\alpha$ , then  $H = \alpha H_0$  but  $T = \alpha C_0 / (1 + \alpha H_0)$ , so there isn't a simple relationship between  $T$  and  $T_0$ . Within the servo band,  $T \sim T_0$ , and outside the servo band,  $T \sim \alpha T_0$ . However,



the “servo bands” have also changed, since a change in the overall magnitude of  $H$  changes the position of the unity gain frequency.

## 6 LLO servo

Rana Adhikari has created a nice Matlab/Simulink model for the LLO DARM loop, collecting all known, measured and inferred information about the system, shown in Fig.5. Notice the presence of a known time delay in the ADC/DAC (analog-to-digital and digital-to-analog conversion) system, of about 50  $\mu$ seconds each. This is not a linear system, but can be linearized in the frequency range used when plotting transfer functions.

The open loop gain obtained from a linearized version of this model between 0.1Hz and 10kHz is shown in Fig.6. A linearization in the Laplace domain (Matlab domain, we should probably call) results in a zero-pole-gain model for the open loop gain of the system with 29 poles and 24 zeros, shown in the table below. Within the frequency band of interest, say 50 Hz to 2kHz (?), the functions are simpler, as shown in the zoomed graphics of Fig.6.

Zeros (in Hertz)	Poles (in Hertz)
1.4154e-010 $\pm$ 4.9451e+004i	-4.0370e+002 $\pm$ 7.8819e+003i
-7.0590e-011 $\pm$ 3.7713e+004i	-4.8090e+002 $\pm$ 7.1772e+003i
1.1387e-010 $\pm$ 1.8119e+004i	-1.3308e+003 $\pm$ 7.0884e+003i
-1.3042e-010 $\pm$ 1.6402e+004i	-2.4748e+003 $\pm$ 5.2030e+003i
3.8341e-011 $\pm$ 1.2817e+004i	-5.0000e+003 $\pm$ 1.6461e-003i
-9.1882e-012 $\pm$ 1.1290e+004i	-1.3578e+003 $\pm$ 3.1431e+003i
3.1831e+003 $\pm$ 1.8378e+003i	-3.4361e+003 $\pm$ 1.9708e+003i
-1.0000e+003	-3.1831e+003 $\pm$ 1.8378e+003i
-999.9998	-1.4142e+003 $\pm$ 1.4142e+003i
-4.7801e-013 $\pm$ 3.4360e+002i	-7.0711e+002 $\pm$ 7.0711e+002i
-100.0000	-1.7180e+000 $\pm$ 3.4360e+002i
-20.0954 $\pm$ 40.6673i	-87.3000
-12.0572 $\pm$ 24.4004i	-2.0095 $\pm$ 4.0667i
-10.0000	-0.0375 $\pm$ 0.7491i
	-4.5418e-013 $\pm$ 3.5564e-006i

The DC gain is 2.3e19 counts/m. Notice that all the poles have a negative real part, as they should for a stable system. We can recognize the pendulum poles ( $-0.0375 \pm 0.7491i$  Hz) and the cavity pole at -87.3 Hz. Everything else would be included in the “servo gain” function  $G$ , and although complicated, is well known.

At any given frequency, one can approximate all poles and zeros *lower* than that frequency ( $|s| \gg |z_i|, |p_i|$ ) by a power law:  $\Pi_1^{N_z(s)}(s - z_i) / \Pi_1^{N_p(s)}(s - p_i) \sim s^{N_z(s) - N_p(s)}$ ; and all poles and zeros *higher* than that frequency ( $|s| \gg |z_i|, |p_i|$ )

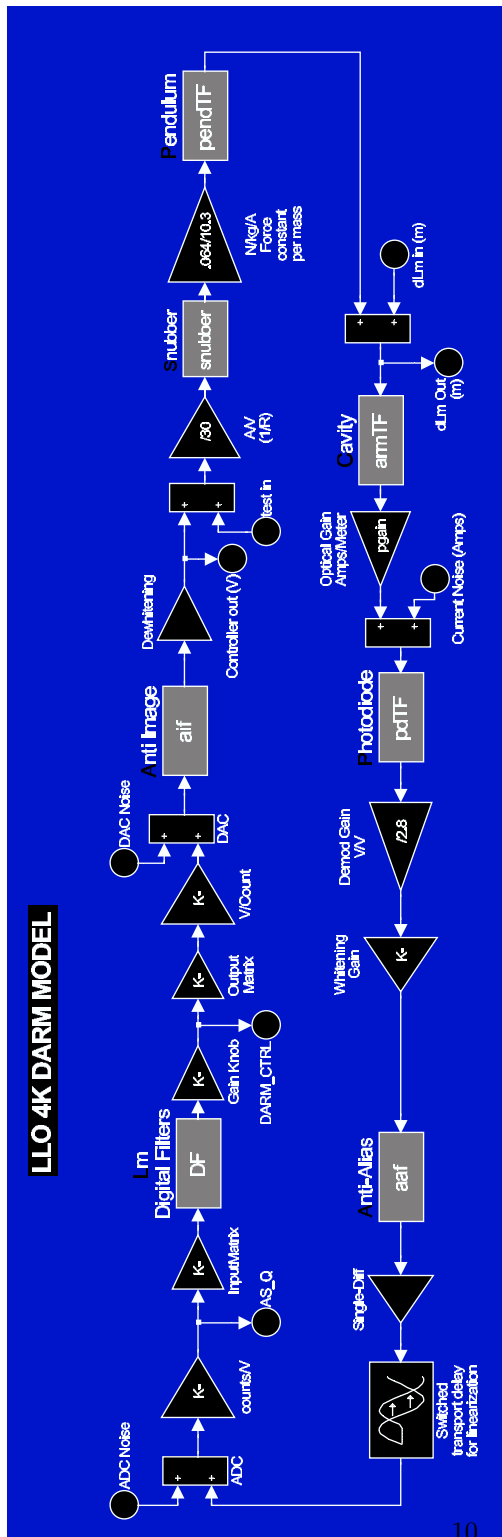


Figure 5: Rana Adhikari's Simulink model for the LLO DARM loop.

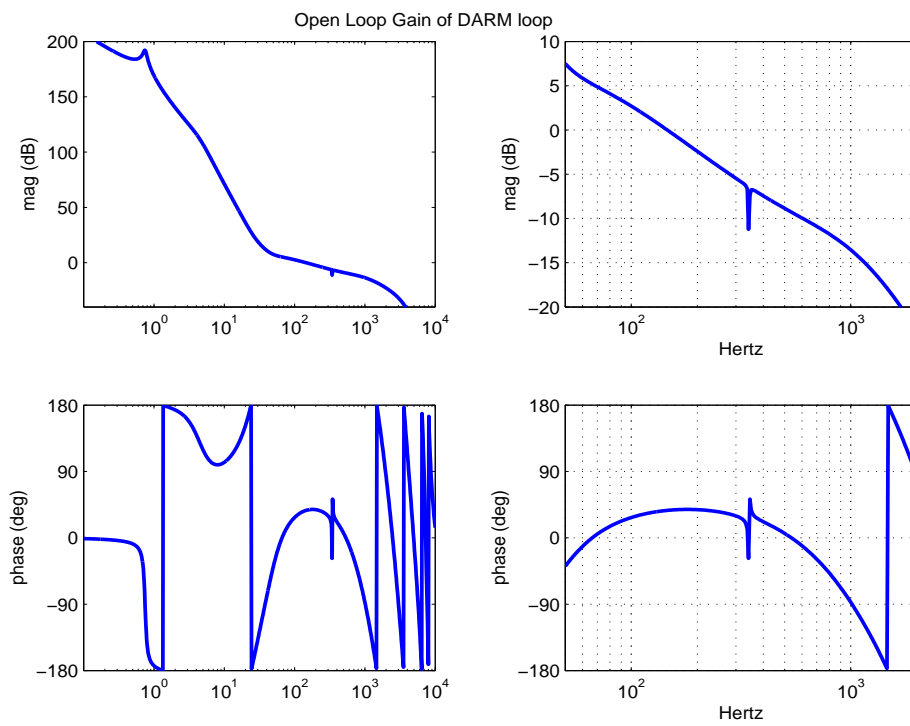


Figure 6: Open loop gain obtained from Rana's Simulink model.

by a constant:  $\Pi_1^{N_z(s)}(s - z_i)/\Pi_1^{N_p(s)}(s - p_i) \sim \Pi_1^{N_z(s)}(-z_i)/\Pi_1^{N_p(s)}(-p_i)$ . Thus, if we arbitrarily decide that we want a simpler expression for the open loop, valid for frequencies between 100 Hz and 2kHz, for example, then the list of poles and zeros can be shortened. Rounding up some numerics, a new approximate list of poles and zeros would be

Zeros (in Hertz)	Poles (in Hertz)
-1.0000e+003	-1.4142e+003 ± 1.4142e+003i
-1.0000e+003	-7.0711e+002 ± 7.0711e+002i
± 3.4360e+002i	-1.7180e+000 ± 3.4360e+002i
-100.0000	-87.3000
-20.0954 ± 40.6673i	0,0,0

The three poles at zero Hz (which could be at any frequency below the frequency range chosen) produce the appropriate power law due to the poles and zeros at low frequencies that were replaced by a power law. The fit to the magnitude is very reasonable, but the phase is not as close, as shown in comparing blue and green curves in Figure7. The fit to the phase can be improved at the low frequency end by changing the poles at zero Hz to higher frequencies but still below the range chosen, as shown in the red curve in Figure7, and at high frequencies by introducing zeros at frequencies above the frequency range. However, it is clear that the best fit would be obtained by using the known and modelled parameters, even if they are outside the frequency range of interest. The low frequency approximations are shown for the open loop gain, but they are easily translated and produce similar (and related) effects in the response function, of course.

Another simplification often tried for calibration purposes is using *real* poles and zeros. In our shortened list of poles and zeros, we see complex poles and zeros near 344 Hz, which are used to produce the notch at the violin modes. These cannot be approximated by real poles and zeros, but if we neglect the effect of the notch, and replace the in-band poles and zeros by their real ones, we get a short list: poles at -2000,-2000,-87.3,15,15,15Hz, and zeros at -100,-46,-46 Hz produce the cyan curve in Figure7. This is not a good approximation for the magnitude or phase, but of course can be improved fitting both to a larger number of poles and zeros and a more appropriate fitting routine.

With the model in hand, we can see the effect of different gains in the calibration function used for AS\_Q. Three different gains in the sensing function  $C(f)$  were used to generate the calibration functions in figure8. We can recognize the unity gain “bumps”, as well as an approximate linear change in the magnitude of the calibration at high frequencies. The change in phase is not so easily approximated, nor is the change in magnitude near the ugf.

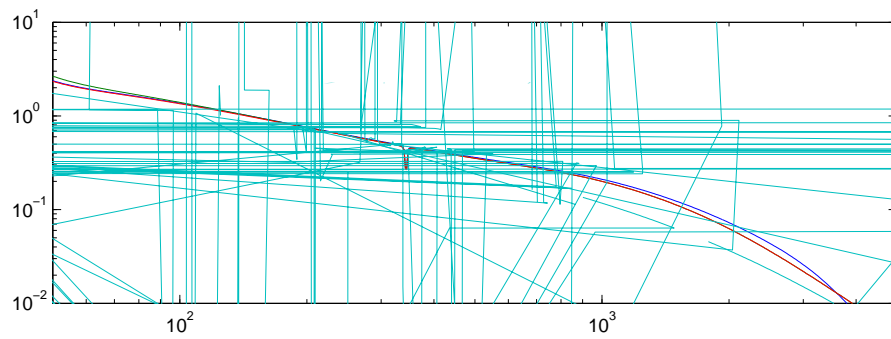


Figure 7: A linearized curve for the open loop gain of the DARM loop obtained from the Simulink model (blue curve), and different approximations to it.

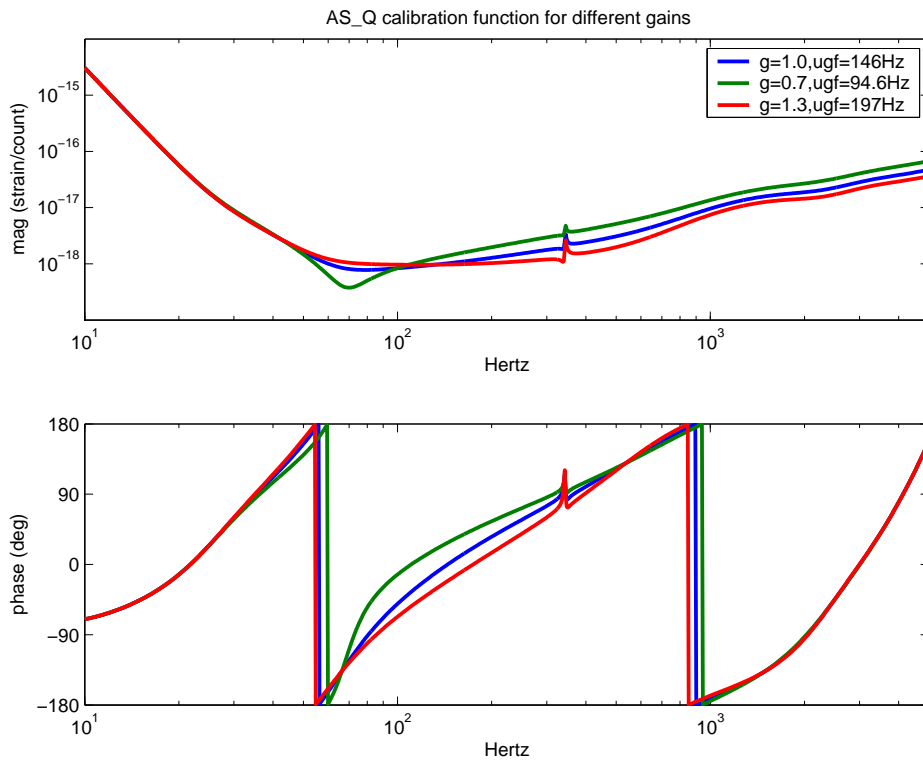


Figure 8: AS\_Q Calibration functions with different sensing gains.

## 7 Conclusions

I hope this note helps to clarify some initial confusion, probably due to just jargon in different fields. However, a few items should be mentioned:

- We expect the interferometer to change its open loop gain by a constant factor (due to alignment, for example). This effect can be measured by calibration lines. However, the response function will not change by a constant factor.
- A fit with real poles and zeros to the open loop gain, or to the response function, is no better or worse than an often longer list of complex poles or zeros obtained from a model of the interferometer and a known series of hardware and software modules. However, changes are more easily tracked, and the effect of different noise sources can be more easily understood with a model. Maybe more importantly, if the model is appropriate, only one parameter (or two, if we observe changes in actuation forces) is left to be fitted to the status of the interferometer at any given instant.
- Rana Adhikari, Mike Landry and others from the “Calibration Team” have both been dutifully producing response curves for the LIGO engineering runs and the first Science run. This note does not in any way replace their work, it only makes their product even more important. Hopefully, more people will now be better able to use their results.