Understanding H1's O3 B Electronics Compensation Systematic Error

J. Kissel for the Calibration Group

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. Review of the Circuit
2. Fit Results for each channel
3. Converting fit results in to systematic error in C
4. Converting sys error in C to sys error in R

But -- your time is valuable

This is 126 page slide show. Here are the answers, in case you don't have time to be educated as to how I came to each conclusion, with the confusing details and the lessons learned that got me to it. I hope at least some folks read it.

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

- **Executive summary:** non-Jeff's everywhere whom guessed the answer ahead of time are vindicated in that **the UIM electronics error** -- either from differences in compensation between states, or poor compensation in general – **doesn't substantially contribute to the response function systematic error. (See slide 70 for quantitative answer)**
- We may safely proceed with O3B chunk 1 uncertainty budget development without including this systematic error.
  - Note that this would have *not* been "covered" by the GPR even it it were non-negligible.

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

- **Executive summary:** While I can predict the systematic error from the configuration switch, **it also doesn't substantially contribute to the response function systematic error (see slide 124 for quantitative answer)**.
- We can probably proceed with O3B chunk 2 uncertainty development without including this systematic error.
- We need to remeasure and recompensate the OMC Whitening Chassis.
- We need to find out what happened on / around 2020-03-23 instead.
- We need to use different measurements we have to make the best guess for the systematic error…

PART I:
The ETMX UIM Driver,
from Nov 27 to Dec 03 2019

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. **Why do you care about the UIM?**
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

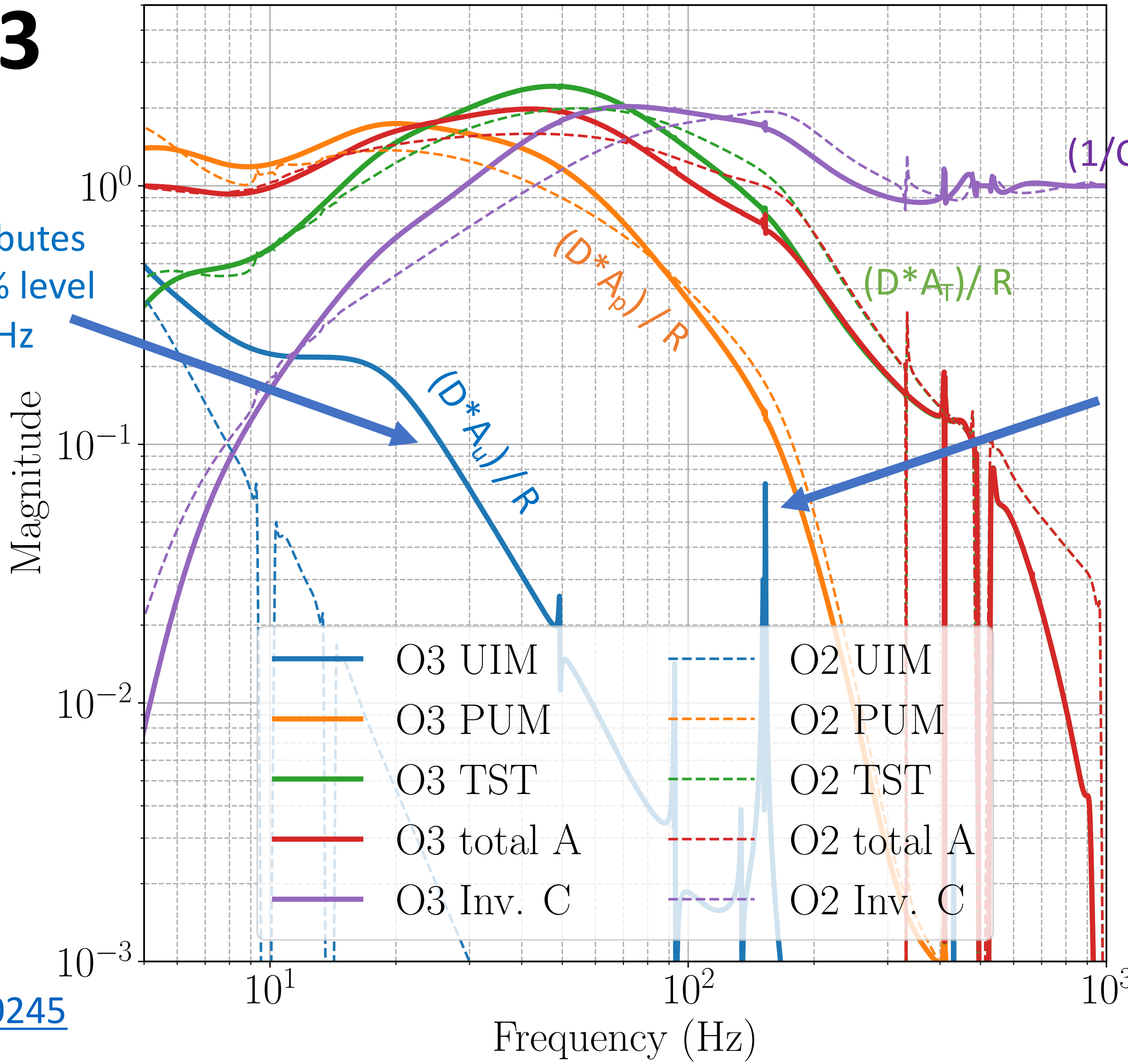
I.1 Why do you care about the UIM?

- The UIM always gets pushed to ”low priority” because “we should be rolling off its authority fast enough that it doesn’t matter in the detection band.”
- That means: we ignore it, assuming anything we do above 10 Hz to the UIM doesn’t matter, and don’t stress about the consequences when we change something until it’s too late.
- We’ve already identified one systematic error in the UIM that has bit us ‘cause we ignored it: the nasty bending response of the UIM Blade + Non-magnetic Blade Dampers.
- This amplifies the contribution of the UIM to the response function at 150 Hz, making *all* UIM systematic error important, right in the bucket. (This is true only for H1, which doesn’t roll off their UIM fast enough. L1 should be safe.)
- **But also: this is the era of the 1%. Even when we fix the UIM contribution by rolling it off faster, this study emphasizes that we must question everything and *confirm* *quantitatively* that something is “negligible.”**
- **This didactic presentation is good practice, and by presenting in great detail, I aim to train the next generation, lest the art of understanding analog electronics analysis dies.**

I.1 Why do you care about the UIM?

H1 O3

UIM Contributes at the ~10% level out to ~25 Hz



Vertical Blade Spring Twisting / Bending in L direction causes UIM contribution to spike back in to play at 150 Hz

Figure 4 from P1900245

Outline

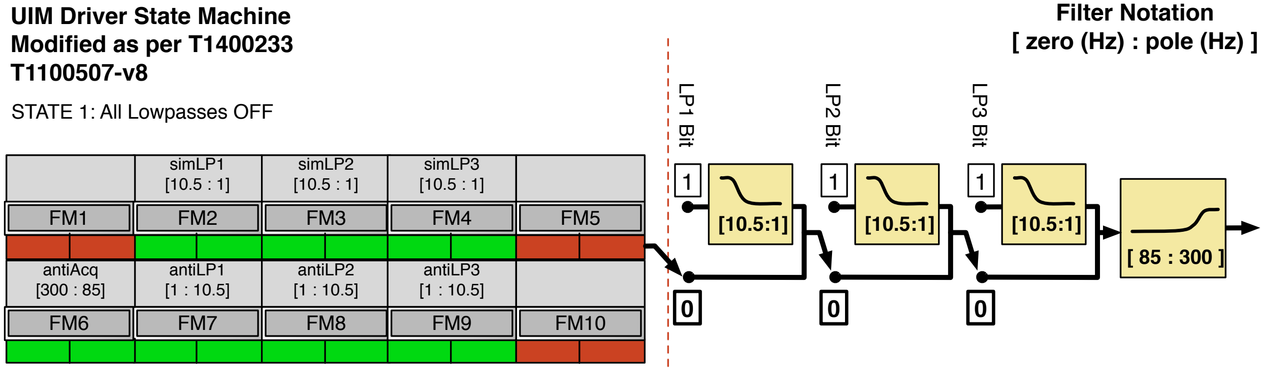
Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. **Review where we were before we started**
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.2 Review of where we were before starting



- During O1 and O2, we were using all ETMY stages for the DARM actuators, the UIM included.
- We updated the low-pass compensation filters on ETMY based on fit to measurements [LHO:21283], but we only used the "DTT measurements with the coil driver monitor circuits" technique, which are insensitive to the 85:300 zero:pole pair which results from the output impedance network [LHO:21142], and we ran out of time (remember GW150914?), so we didn't update the antiAcq filter.
- We ran in ETMY, UIM, State 1 for all of O1 and O2, so the updates didn't actually matter (Sorry Darkhan!).
- In Jan 2019, 4 months before O3, we made the switch to using all ETMX stages for the DARM actuator.
- The UIM electronics were fully measured in analog on Feb 03 2019 (yes, a Sunday!), by Rich Abbott and Jeff Kissel. Rich was unconvinced that we needed the differential driver full setup as described in D1900027, so we did some sort of single-ended, direct via clip-leads measurement [LHO:46927] (this becomes important later).
- Lilli tried to fit the State 1 data, but they didn't make any sense to us at the time [LHO:47195].
- We did take the DTT data on Feb 07 2019 to update the low pass compensation, but never got to processing it.
- Because of confusion about the results in the State 1 measurements, and because the UIM was low priority, we just chose not to update anything: [LHO:47167]. (Remember ER14 and how there was systematic error everywhere [LHO:47378] ?)
- Flash-forward to Nov 27 2019, we got suspicious of DAC quantization noise [LHO:53376], and switched the ETMX UIM driver to State 2 [LHO:53528], forgetting the terrible state of the compensation, and assuming "the UIM doesn't matter."
- Only 6 days later on Dec 03 2019 (and thus in between regular calibration sweeps), we reverted back to State 1 [LHO:53652].
- **The switch happened between two regular actuator sweeps (taken on 2019-11-11 and 2019-12-04), so there for we must model what the systematic error with the measurements we have (namely, the Feb 03 2019 data) for this 6 day period, in which -- of course – there lies GW191129.**

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. **Review of the Circuit**
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

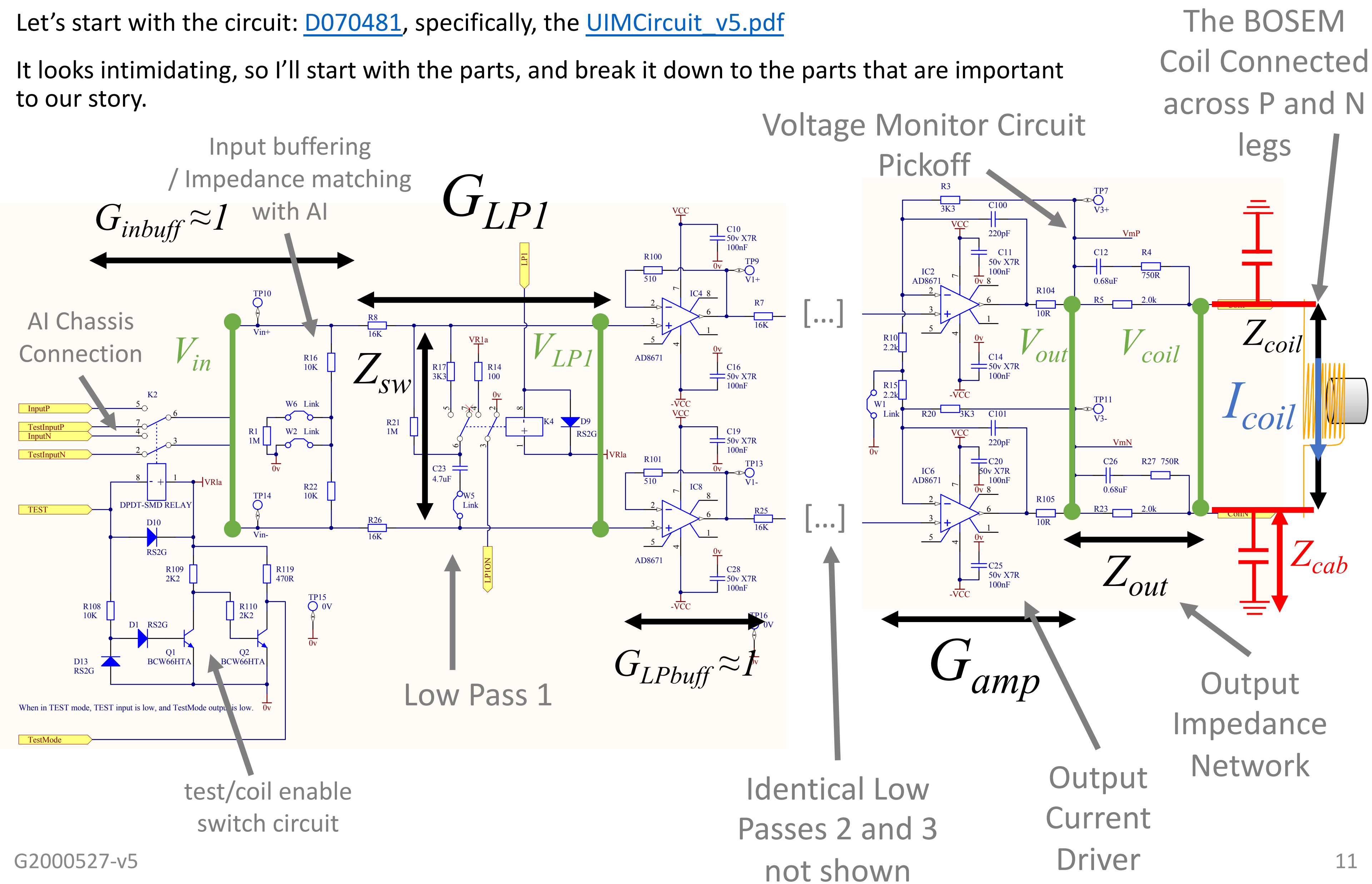
PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.3 Review of the Circuit: Forest through the Trees

To understand the 2019-02-03 data, we need to understand the circuit and the measurement.

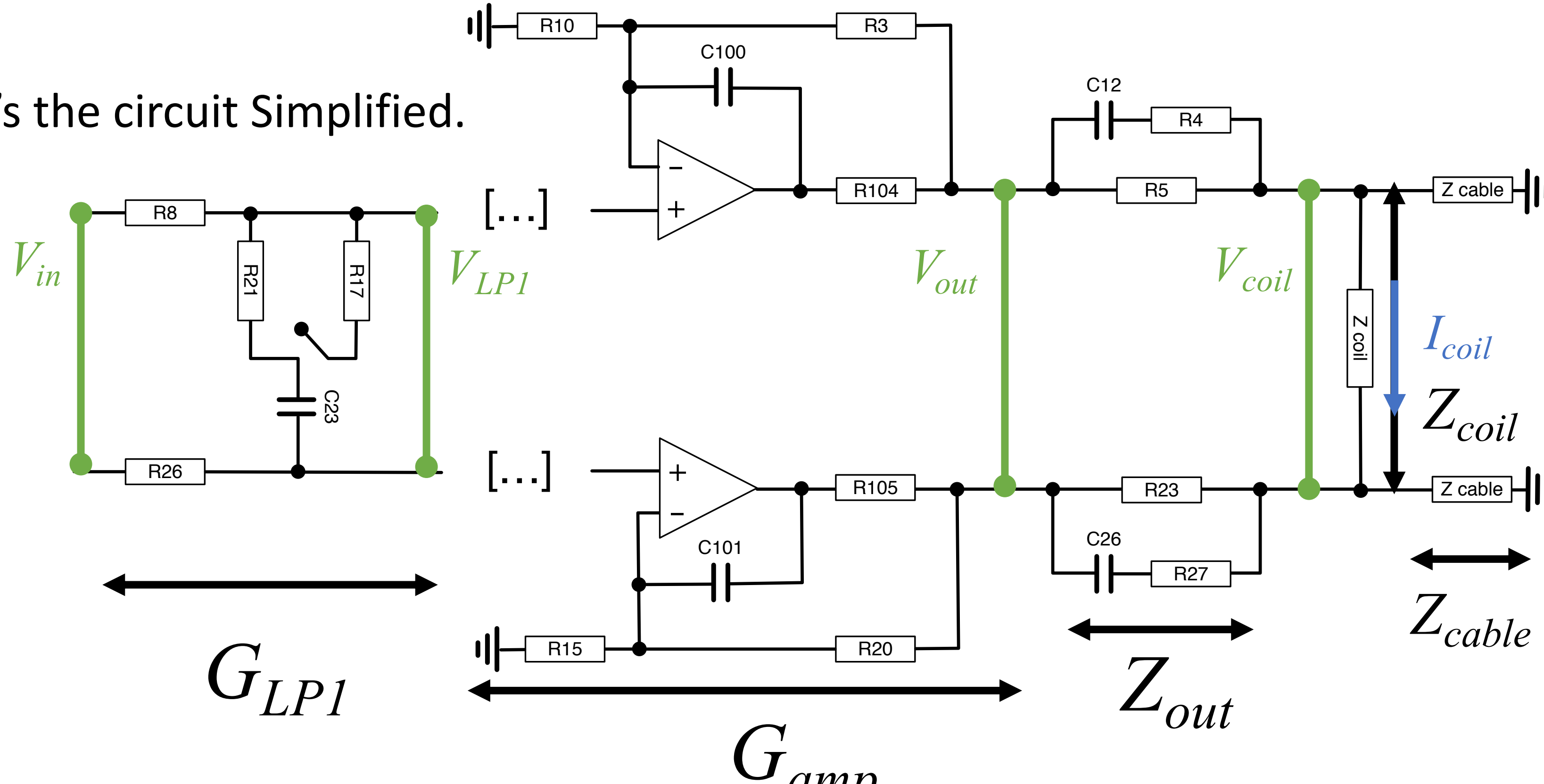
Let's start with the circuit: [D070481](D070481), specifically, the UIMCircuit_v5.pdf

It looks intimidating, so I'll start with the parts, and break it down to the parts that are important to our story.



I.3 Review of the Circuit: Simplified, Differential

Here's the circuit Simplified.



- All parts of the circuit that have gain, but no frequency dependence, we just ignore. We'll scale the gain of all models to the measurement in the end. We're looking for poles and zeros
- The low pass, the output impedance network, and the coil will define the "important" poles and zeros (below 1 kHz). (I wonder if the output current amplifier is important later)
- In the end, the "transfer function" we want the *transconductance* of the driver / coil system: I_{coil} / V_{in}

I.3 Review of the Circuit: Trust the Basics

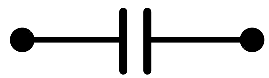
Here's a friendly reminder of the tools in the circuit analysis toolbox:

Converting to Impedance:

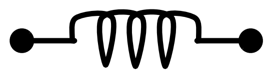
$$Z_R = R$$

$$Z_C = 1/i\omega C$$

$(\omega = 2\pi f)$

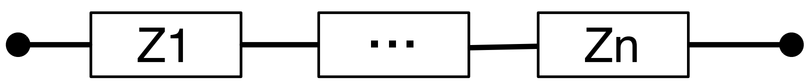


$$Z_L = i\omega L$$



Series Impedance:

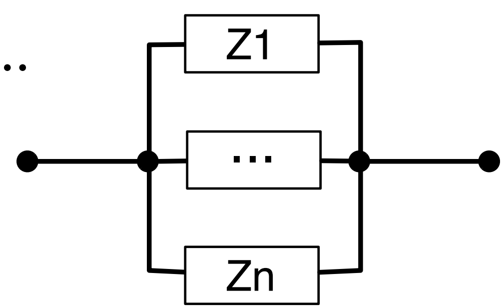
$$Z_{tot}^{S} = Z_1 + Z_2 + \ldots$$



Parallel Impedance:

$$\frac{1}{Z_{tot}^{P}} = \frac{1}{Z_1} + \frac{1}{Z_2} + \ldots$$

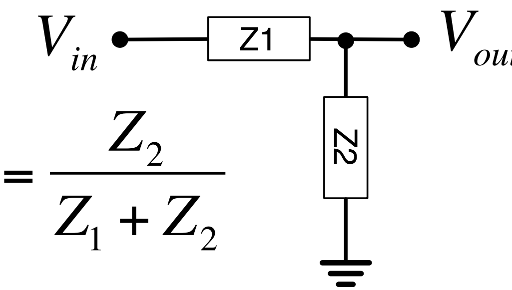
$$Z_{tot}^{P(2)} = \frac{Z_1 Z_2}{Z_1 + Z_2}$$



Ohm's Law:

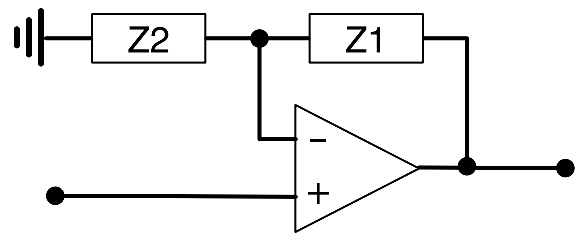
$$V = I Z$$

Voltage Divider:



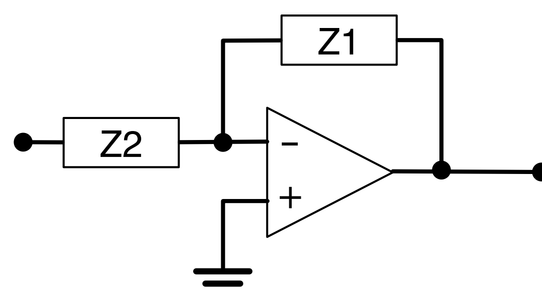
$$\frac{V_{out}}{V_{in}} = \frac{Z_2}{Z_1 + Z_2}$$

Non-inverting Op-Amp



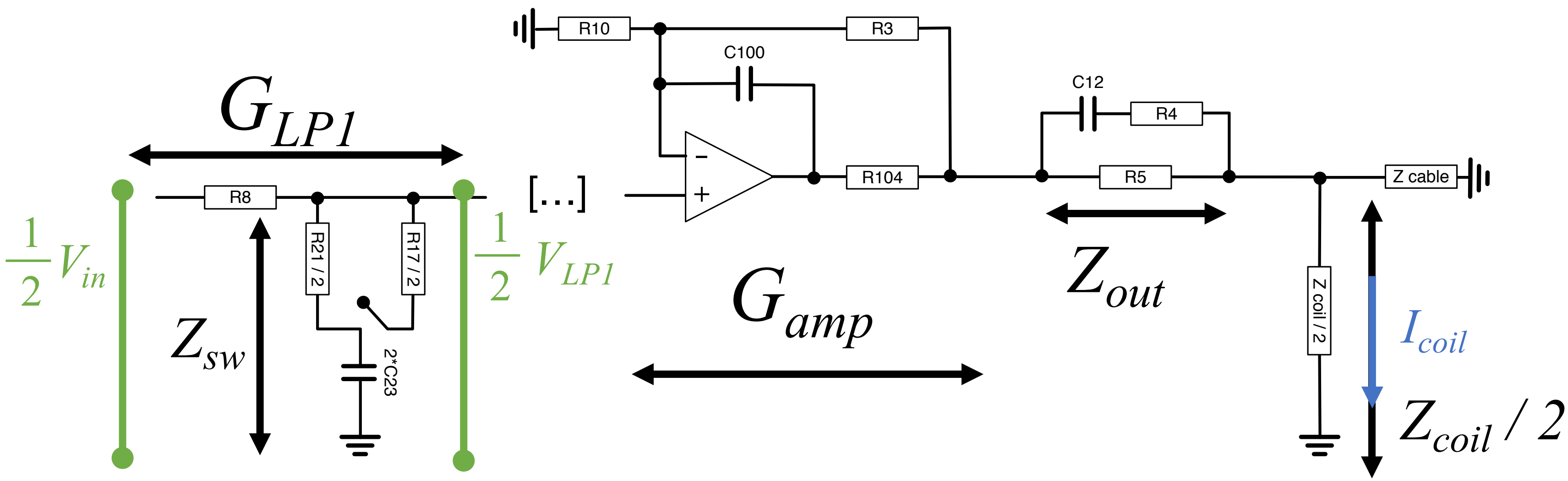
$$\frac{V_{out}}{V_{in}} = 1 + \left(\frac{Z_1}{Z_2}\right)$$

Inverting Op-Amp



$$\frac{V_{out}}{V_{in}} = -\left(\frac{Z_1}{Z_2}\right)$$

I.3 Review of the Circuit: Simplified, Single-Ended



- **One trick for differential circuit analysis:** consider only one leg, and divide everything that "crosses between legs" by two -- voltage, impedance, etc, -- and reference everything to ground (0V). The transfer functions are the same, and the analysis is equivalent.

$$Z_{sw}^{Open}(\omega) = \frac{1}{2}\left(R_{21} + \frac{1}{i\omega C_{23}}\right)$$

$$Z_{sw}^{Closed}(\omega) = \frac{1}{2}\left(\left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right] + \frac{1}{i\omega C_{23}}\right)$$

$$\frac{(1/2)\,V_{LP1}}{(1/2)\,V_{in}} = \frac{V_{LP1}}{V_{in}} = G_{LP1} = \frac{Z_{sw}}{R_8 + Z_{sw}}$$

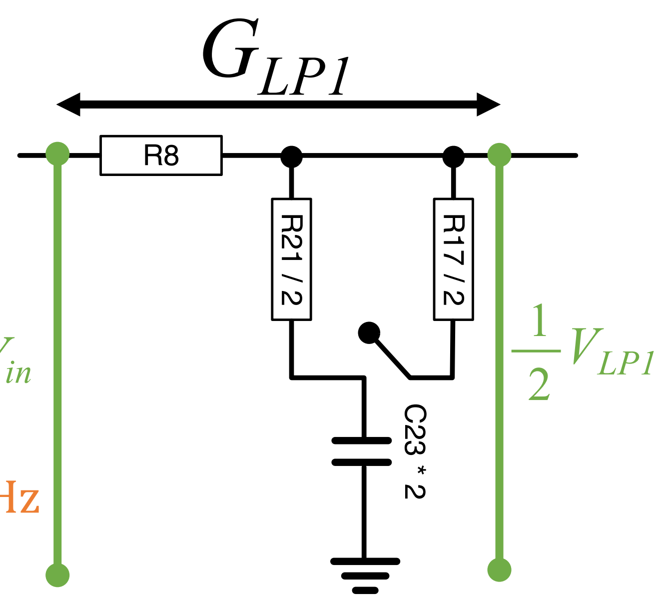
With the switch closed, that means,

$$G_{LP1}\Big|_{closed} = \frac{\frac{1}{2}\left(\left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right] + \frac{1}{i\omega C_{23}}\right)}{R_8 + \frac{1}{2}\left(\left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right] + \frac{1}{i\omega C_{23}}\right)}$$

which is enough to plot the transfer function, but we can re-arrange to show the analytic computation of the poles and zeros of this TF…

I.3 Review of the Circuit: The Low Pass

$$G_{LP1}\Big|_{closed} = \frac{\left(1 + i\omega \left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right] C_{23}\right)}{\left(1 + i\omega \left(R_8 + \frac{1}{2}\left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right]\right)(2C_{23})\right)}$$



That means

$$f_z^{LP1}\Big|_{closed} = 1/\left(2\pi \left[\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right] C_{23}\right) = 10.2953 \text{ Hz}$$

$$f_p^{LP1}\Big|_{closed} = 1/\left(2\pi \left[R_8 + \frac{1}{2}\frac{R_{17}R_{21}}{R_{17}+R_{21}}\right](2C_{23})\right) = 0.9596 \text{ Hz}$$

Consistent with the expected low pass z:p = (10.5 : 1.0) Hz.

With the switch open, Rpara reduces to R17, leaving,

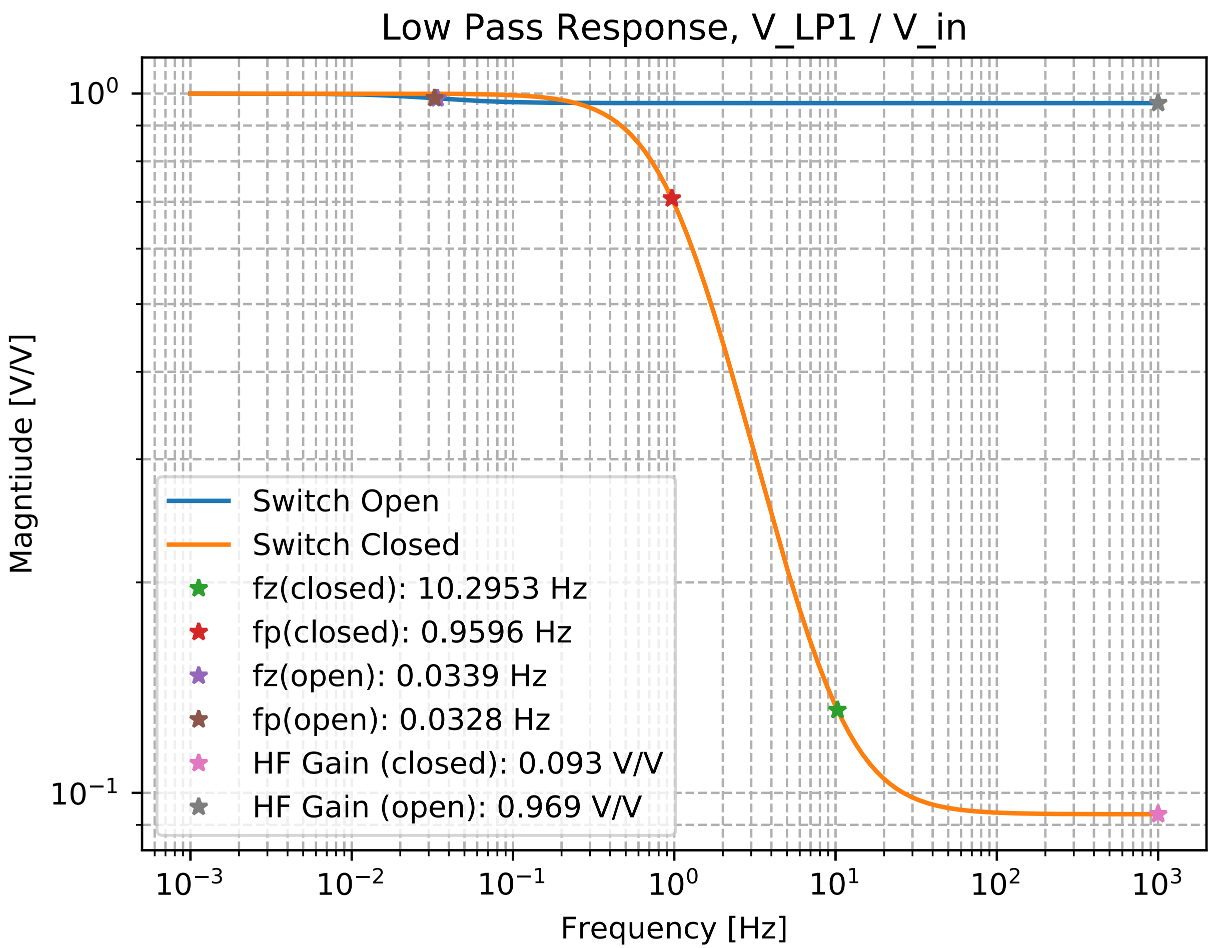
$$G_{LP1}\Big|_{open} = \frac{1 + i\omega R_{21}C_{23}}{1 + i\omega\left(R_8 + \frac{1}{2}R_{21}\right)(2C_{23})}$$

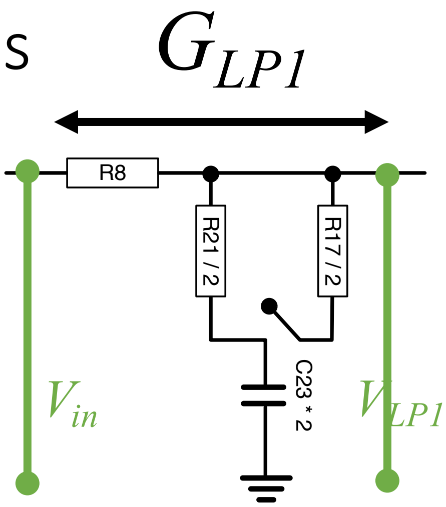
$$f_z^{LP1}\Big|_{open} = 1/(2\pi R_{21}C_{23}) = 0.0339 \text{ Hz}$$

$$f_p^{LP1}\Big|_{closed} = 1/\left(2\pi\left[R_8 + \frac{1}{2}R_{21}\right](2C_{23})\right) = 0.0328 \text{ Hz}$$

R8 = 16e3 # Ohms
R17 = 3.3e3 # Ohms
R21 = 1e6 # Ohms
C23 = 4.7e-6 # Farads

I.3 Review of the Circuit: The Low Pass





Another analysis trick: the suppression of the low pass (i.e. the asymptotic gain at high frequency) is the ratio of f_p / f_z

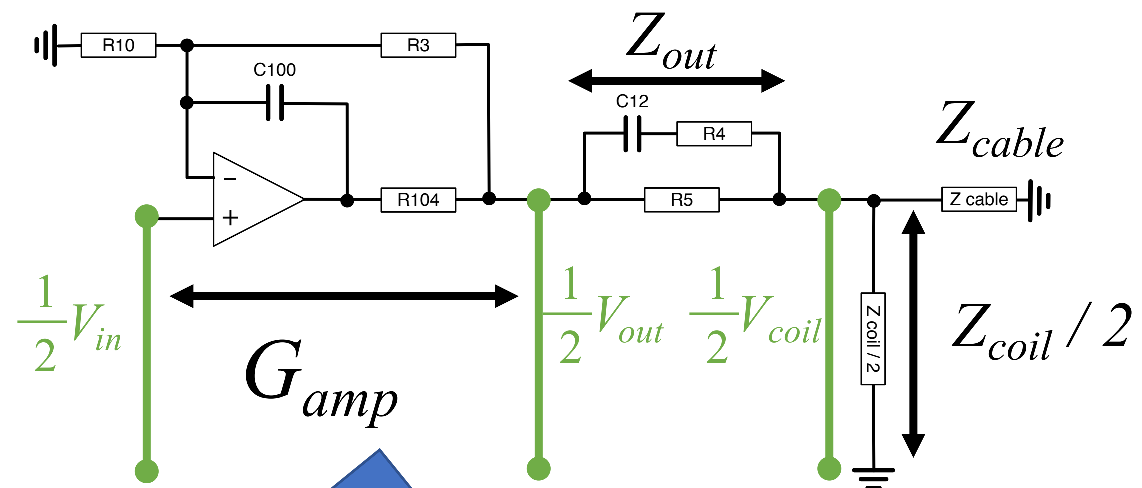
Also, with the switch open, the pole and zero nearly cancel.

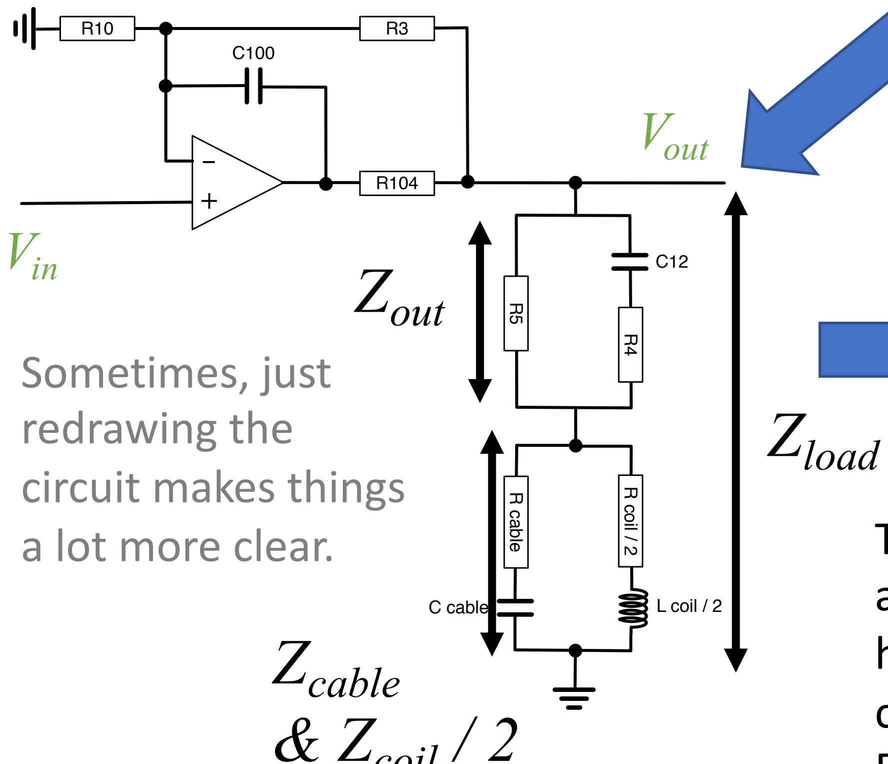
- When the switch is open, each LP stage – above 0.1 Hz – has gain of 0.969 V/V ≈ 1 V/V.
- Where we're concerned – above 1 Hz – we can treat this as "just" a part of the overall gain to be measured later, and uninteresting in terms of the frequency response
- **Thus: For State 1 (with no low passes on, all switches open), we can ignore the response of all three low passes.**

I.3 Review of the Circuit: The Output

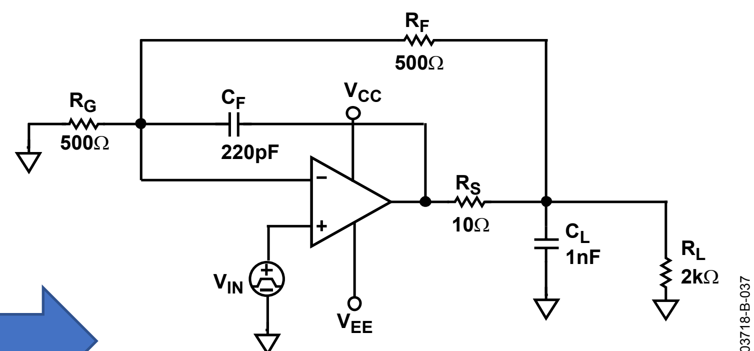
On to the response of the amplifier gain and impedance network.

These are important for State 1.





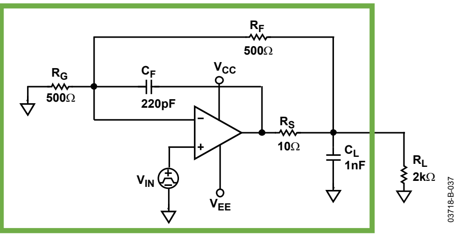
Sometimes, just redrawing the circuit makes things a lot more clear.



From the AD8671 Data Sheet!

This complicated network can be treated as “just” a non-inverting amplifier, with capacitive load, that has been “in-loop compensated.” More in-loop compensation here.
But, with a “duct tape and bubble gum” story …

I.3 Review of the Circuit:

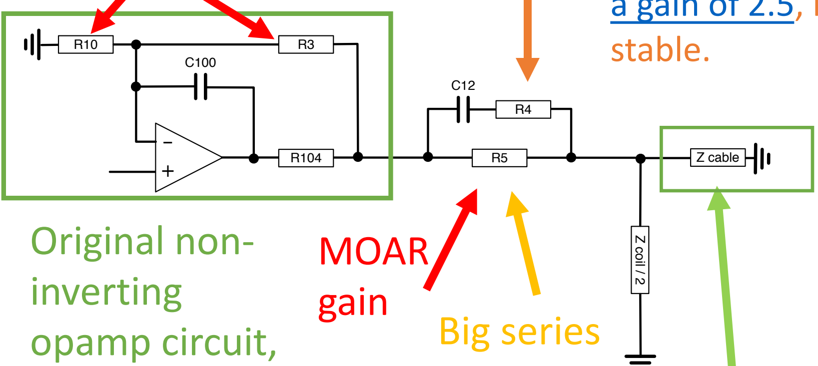


R_S = R104 = 10 # Ohm
R_F = R3 = 3.3e3 # Ohm
R_G = R10 = 2.2e3 # Ohm
C_F = C100 = 0.22e-9 # Farad
C_L = C_{cable} = 1.0e-9 # Farad
R4 = 750.0 # Ohm
"R_L" = R5 = 2.0e3 # Ohm
C12 = 0.68e-6 # Farad

RC bypass

$Im\{\frac{1}{i\omega C_{12}}\} \ll \frac{R_4 R_5}{(R_4+R_5)}$ by 1 kHz and above, so AD8671, whose UGF is ~2-4 MHz with a gain of 2.5, is quite stable.

MOAR gain



Original non-inverting opamp circuit, in-loop compensated with R104

MOAR gain

Big series resistor

Original cable load that motivated the in-loop compensation design

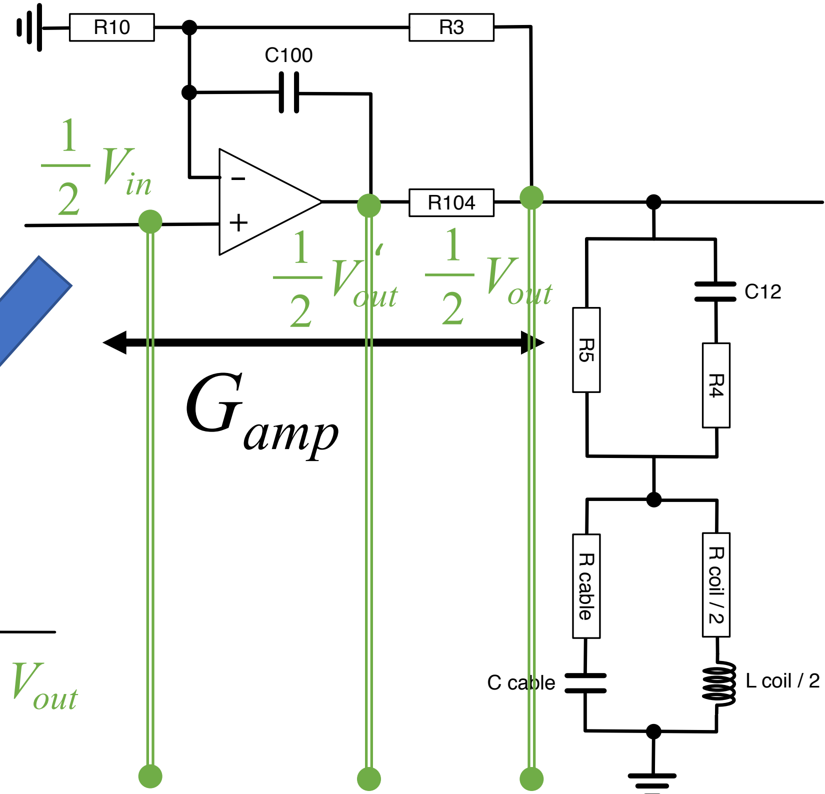
- Use the AD8671, it's a nice, low noise op amp.
- "OK, I'll use the data-sheet-recommended configuration, because the cable run will be pretty long – probably 3 nF of parasitic capacitance. Same component values should be fine."
- Right, but be conscious of the current noise, so make sure R_L stays big (the BOSEM, Z_{coil} , should be connected after it), and DAC noise from upstream.
- "OK, cool, a big series resistor *in the driver circuit*, prior to the cable load with R5 = 4k and increase R_F/R_G from 1 to 0.33."
- Mmm... but that reduces the actuator range. Can you give me more gain?
- "Sure –let's put in an RC bypass around R_L to amplify the range at 100 Hz."
- That reduces the protection against current noise, but should still be OK. And also... sorry... we still need more range.
- "OK, dropping R_5 to 2k, and bumping R_F/R_G up to 0.5."
- **But wait... the circuit isn't really ever capacitively loaded any more... so the this design doesn't make sense with this silly R_S that makes the circuit confusing to analyze!**

I.3 Review of the Circuit: Op-Amp = Just a Gain

So can we just ignore R_S = R104 = 10 Ohms?

```
R3 = 3.3e3          # Ohm
R10 = 2.2e3         # Ohm
R104 = 10           # Ohm
C100 = 220e-12      # Farad
```

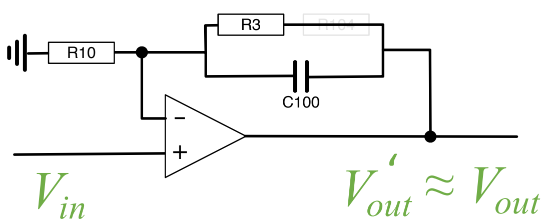
$$\frac{V_{out}}{V'_{out}} = \frac{R_3}{R_3 + R_{104}} = 0.997$$



YES. R10 is just a tiny voltage drop between Vout and Vout'.

Now, "just" a non-inverting amplifier.

What's the pole frequency?



$$Z_{RC} \approx \frac{R_3(1/i\omega C_{100})}{R_3 + 1/i\omega C_{100}} = R_3 \frac{1}{(1 + i\omega R_3 C_{100})}$$

$$G_{amp} \approx \frac{V_{out}}{V_{in}} = 1 + \left(\frac{Z_{RC}}{R_{10}}\right) = 1 + \frac{R_3}{R_{10}}\left(\frac{1}{1 + i\omega R_3 C_{100}}\right)$$

$$G_{amp}\Big|_{DC} = 1 + \frac{R_3}{R_{10}} = 2.5$$

$$f_p^{RC} = 1/(2\pi R_3 C_{100}) = 219.2\ kHz$$

219 kHz is sufficiently high frequency that we can ignore this pole too.

So, YES, ignore R104 *and* C100.
G_{amp} for us is just 2.5.
This won't be a part of the State 1 response either

I.3 Review of the Circuit: Output Zs: Rs, Ls, and Cs

Let's look at the load impedance.
We'll find out here's from were *all* the response from State 1 comes.

$$Z_{out} = \frac{R_5(R_4 + 1/i\omega C_{12})}{R_5 + (R_4 + 1/i\omega C_{12})} = R_5\left(\frac{1 + i\omega R_4 C_{12}}{1 + i\omega (R_4 + R_5) C_{12}}\right)$$

$$f_z^{out} = 1/(2\pi R_4 C_{12}) = 312.069\ Hz$$

$$f_p^{out} = 1/(2\pi (R_4+R_5) C_{12}) = 85.110\ Hz$$

$$Z_{coil} = \frac{1}{2}(R_{coil} + i\omega L_{coil}) = \frac{R_{coil}}{2}(1 + i\omega (L_{coil}/R_{coil}))$$

$$f_z^{coil} = 1/(2\pi L_{coil}/R_{coil}) = 571.085\ Hz$$

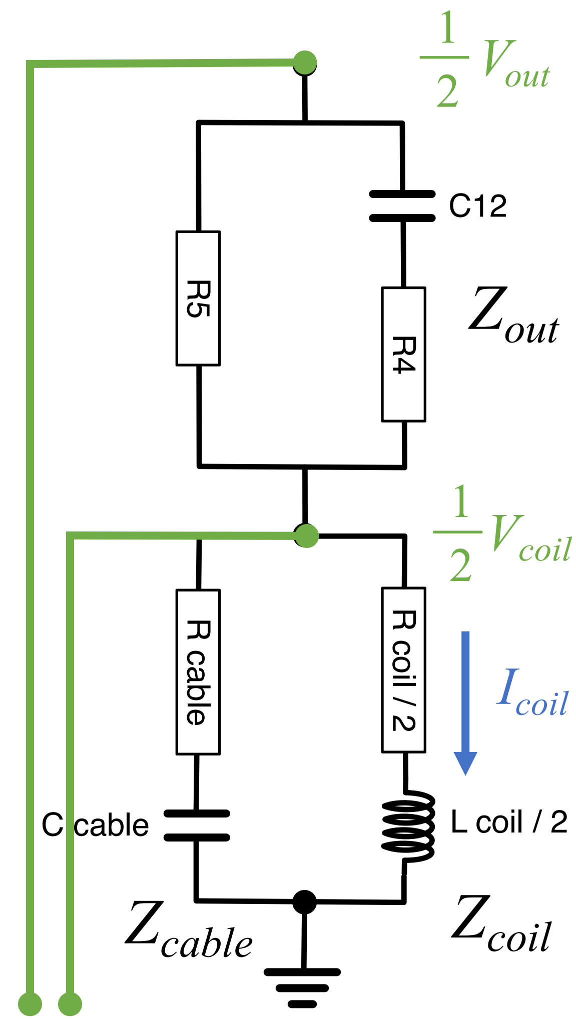
$$Z_{RLC} = Z_{coil} \parallel Z_{cable}$$

$$= \frac{1}{2} R_{coil} \frac{(1 + i\omega[L_{coil}/R_{coil}])(1 + i\omega R_{cable} C_{cable})}{(1 + i\omega(2R_{cable} + R_{coil})C_{cable} - (1/2)\omega^2 L_{coil} C_{cable})}$$

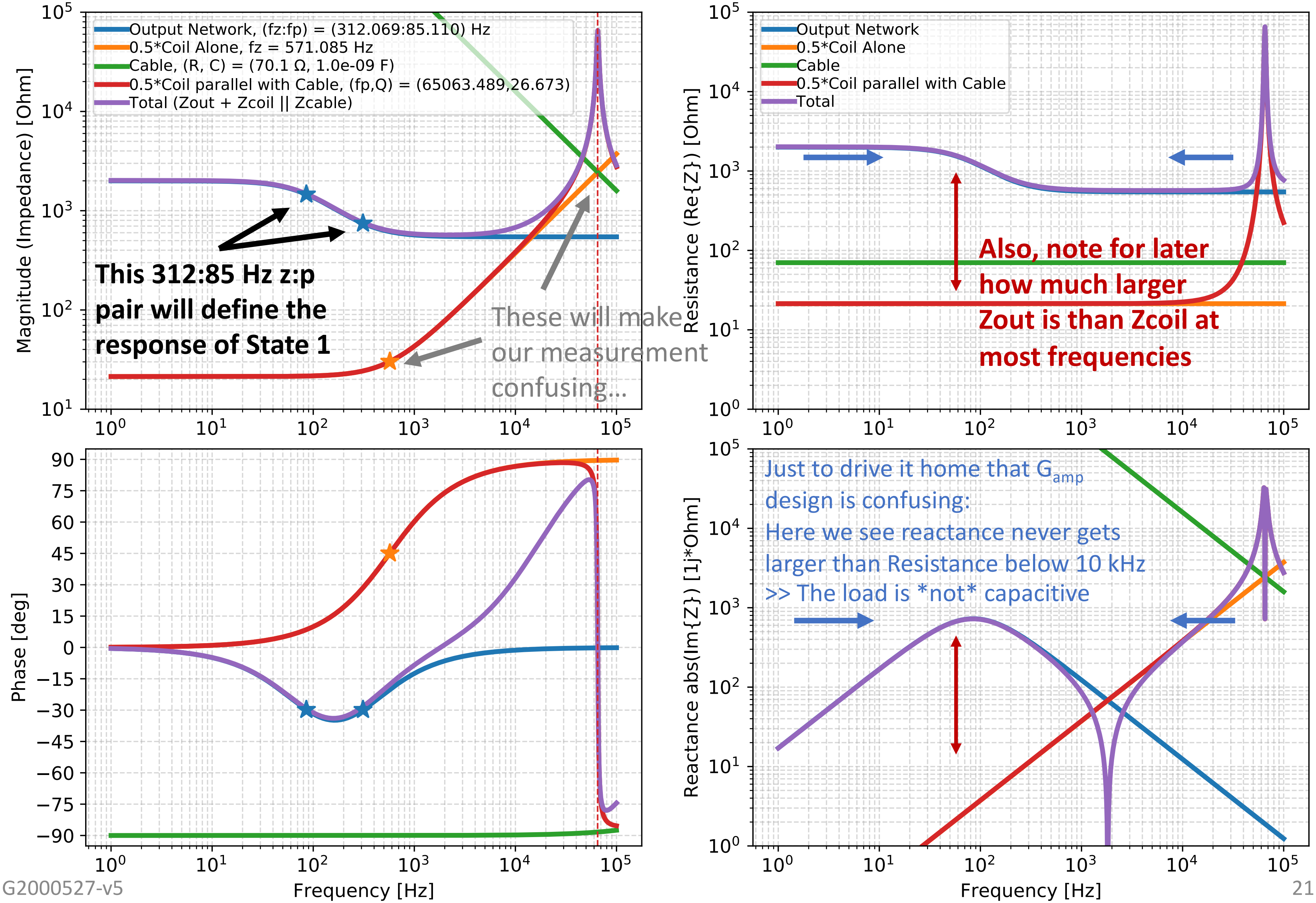
$$f_z^{coil} = 1/(2\pi L_{coil}/R_{coil}) = 571.085\ Hz$$

$$f_z^{cable} = 1/(2\pi R_{cable} C_{cable}) = 2.27\ MHz$$

$$f_p^{coil||cable} = 1/(2\pi)\sqrt{\frac{1}{(1/2)L_{coil}C_{cable}} - \left(\frac{2R_{cable} + R_{coil}}{(1/2)L_{coil}}\right)^2} = 65.063\ kHz$$

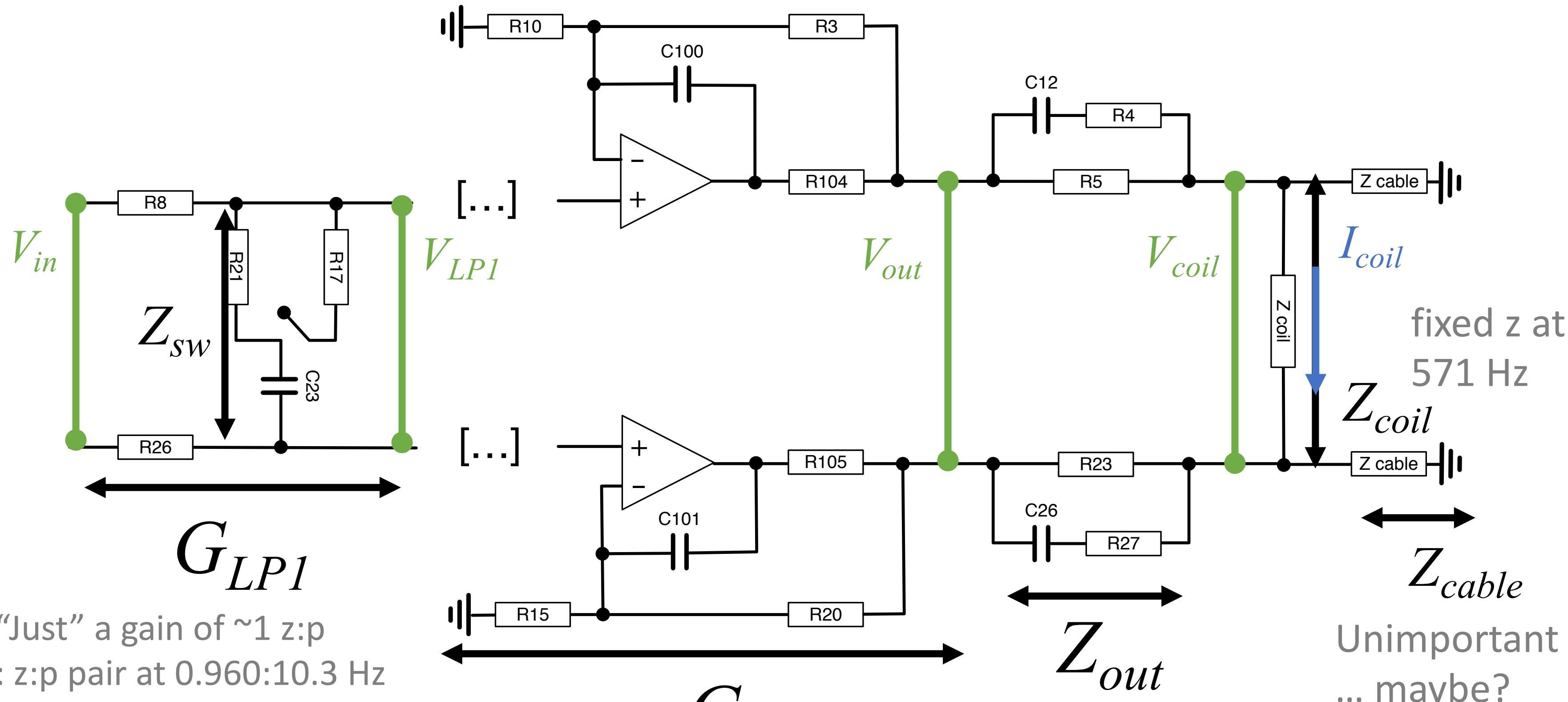


I.3 Review of the Circuit: Z total: poles and zeros



I.3 Review of the Circuit: OK, Let's Review

OK, now that we know what kind of response to expect from everything, we can head back to the differential picture and summarize.



Open: "Just" a gain of ~1 z:p
Closed: z:p pair at 0.960:10.3 Hz

fixed z:p pair at 312:85 Hz

"Just" a gain of 2.5.

Well, that was a fun exercise.
But what do we really want?
The response of the current created across the coil, I_{coil} to V_{in}.
So let's talk about how to measure it.

Outline

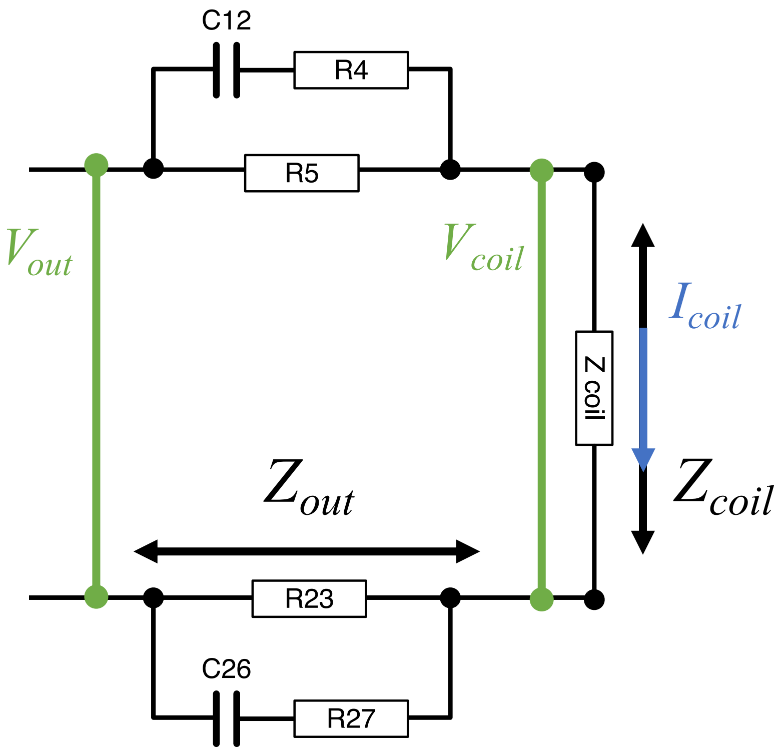
Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. **The Measurement**
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.4 The Measurement: Coil Current/Vout



We calculated that whatever the values of Rcable and Ccable are, they're not going to matter until several 10s of kHz. So let's make it easy to think about.

In **State 1**, we already know, Vout/Vin is "just a gain" at ~2.5. So the current, held fixed by the G_{amp} opamps, will just obey Ohms Law as it heads out to the coil and back across the differential connection to the coil:

$$V_{out} = I_{coil}\, Z_{total} = I_{coil}(2Z_{out} + Z_{coil})$$

$$\frac{I_{coil}}{V_{out}} = \frac{1}{(2Z_{out} + Z_{coil})}$$

And, we know for State 1, that means,

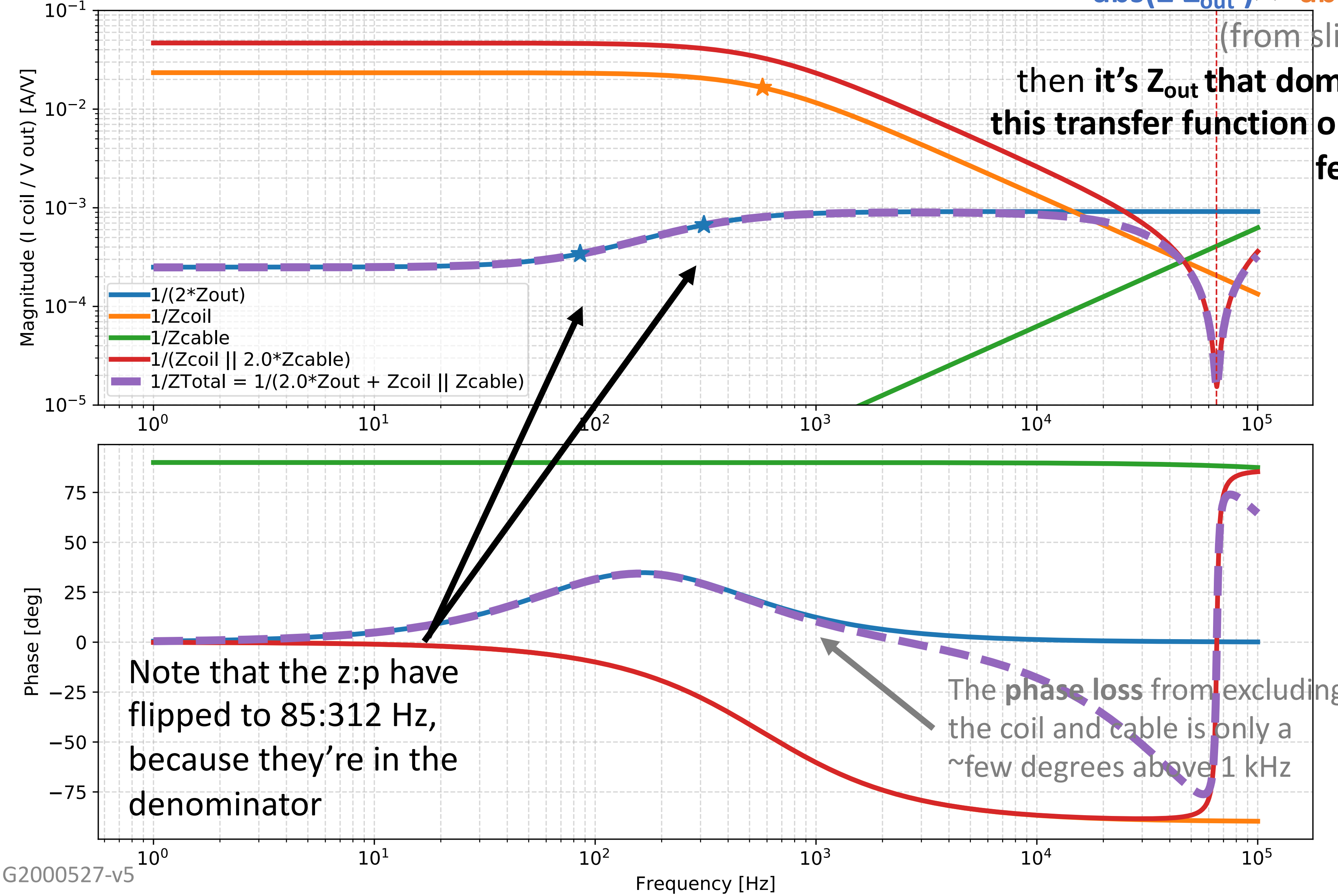
$$\left.\frac{I_{coil}}{V_{in}}\right|_{State\ 1} \approx \frac{2.5}{(2Z_{out} + Z_{coil})}$$

So let's look at that response, with our basic analytic model.

I.4 The Measurement: Coil Current / Vout

$$I_{coil} / V_{out} = 1/(2Z_{out} + Z_{coil})$$

Ah – OK, since
abs(2 Z_{out})>> abs(Z_{coil})
(from slide 17)
then **it's Z_{out} that dominates this transfer function out to a few kHz**



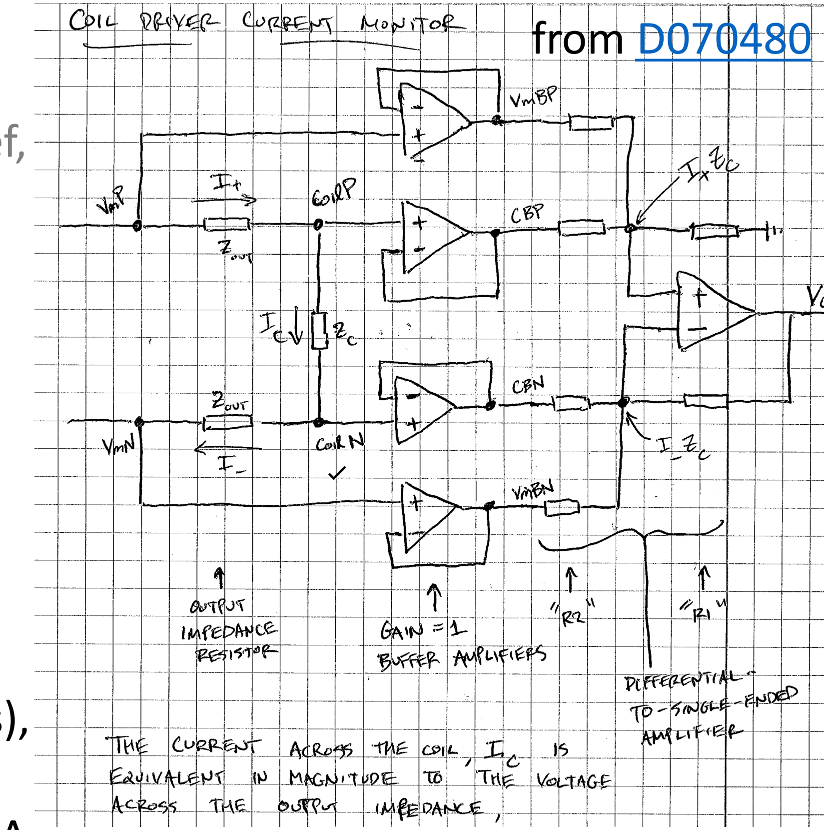
I.4 The Measurement: Fast Current Monitor?

Why do we have to physically measure the transfer function in analog? Why not use the fast current monitor?

- **The answer does include the output impedance network for this driver** (contrary to popular belief, started by 2014 Jeff)
- BUT -- the fast current monitor board itself may contribute some frequency dependence, and there's an AA chassis between the analog IMON signal and where it's read in by the DAQ. **These responses will confuse fitting routines and/or your interpretation of the results.**
- It works well for *ratios* of measurements, namely to get poles and zeros from things that *change* between states (i.e. the low pass filters), but it does not help you characterize State 1.
- We typically operate in state 1, and at least the AA chassis has appreciable response in frequency bands of interest to us, so …

- Analog measurement it is.

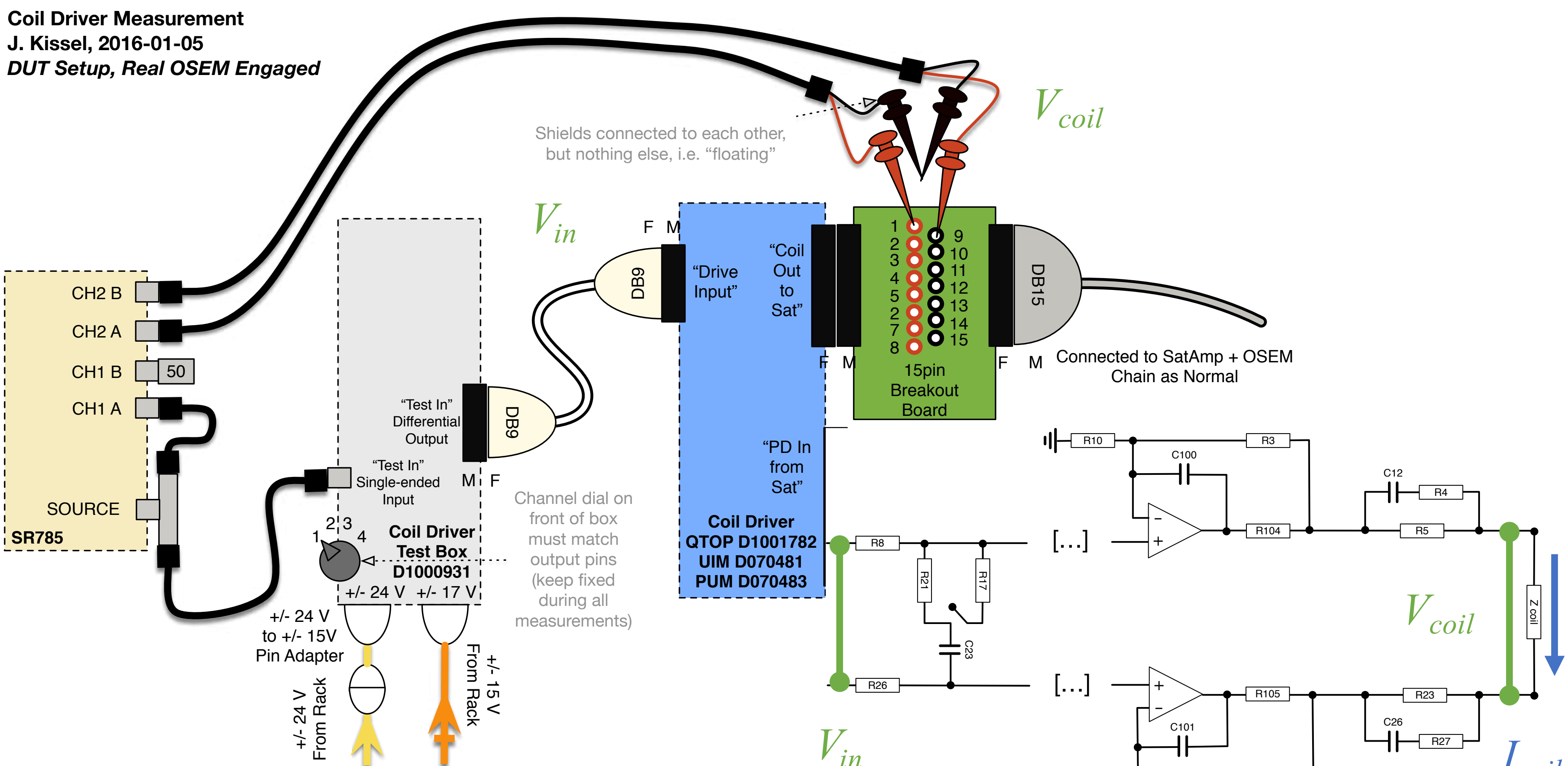
from D070480



$$\frac{V_o}{I_c} = 2 Z_{out} \frac{R_1}{R_2}$$

I.4 The Measurement: Should / Would / Could…

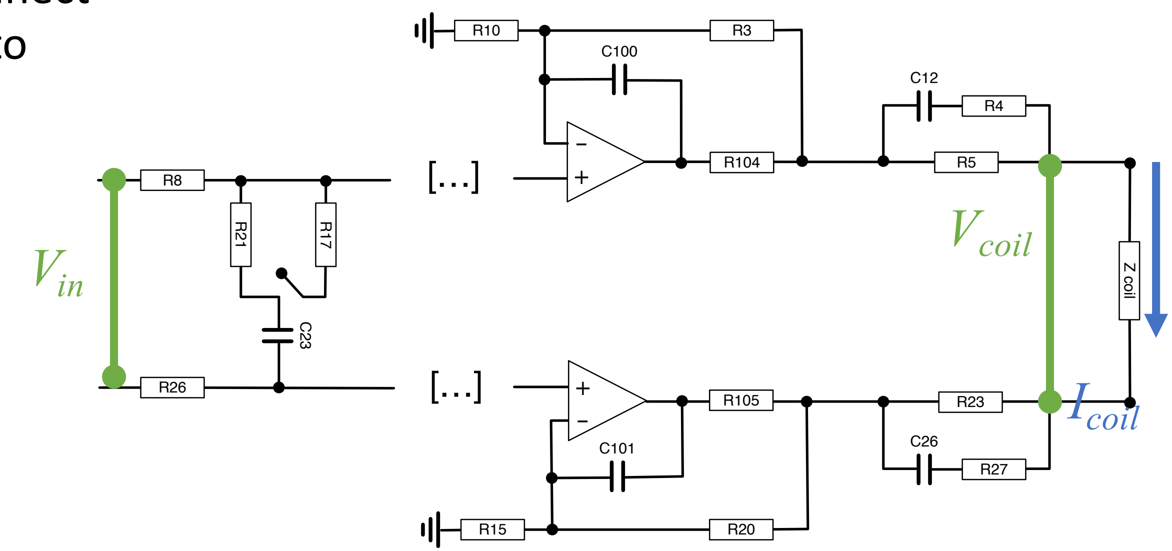
OK Great! Gung-ho Jeff will go out there, he'll take some clip leads, a differential driver, and breakout boards, to measure at the output of the driver – but leaving output connected to the OSEM as normal, "because you need the current to go across the output legs" – the op-amps need to be loaded with *something,* so might as well make it "as accurate as possible."



I.4 The Measurement: Facepalm!

But wait – if you've left the coil connect "as normal" then you're not going to measure…

$$\left.\frac{I_{coil}}{V_{in}}\right|_1 \approx \frac{2.5}{(2Z_{out} + Z_{coil})}$$



But instead…

$$V_{coil} = I_{coil} Z_{coil}$$

$$\left.\frac{V_{coil}}{V_{in}}\right|_1 \approx \frac{2.5\, Z_{coil}}{(2Z_{out} + Z_{coil})}$$

$$f_z^{coil} = 1/(2\pi L_{coil}/R_{coil}) = 571.085\ Hz$$

$$f_p^{out} = 1/(2\pi R_4 C_{12}) = 312.069\ Hz$$

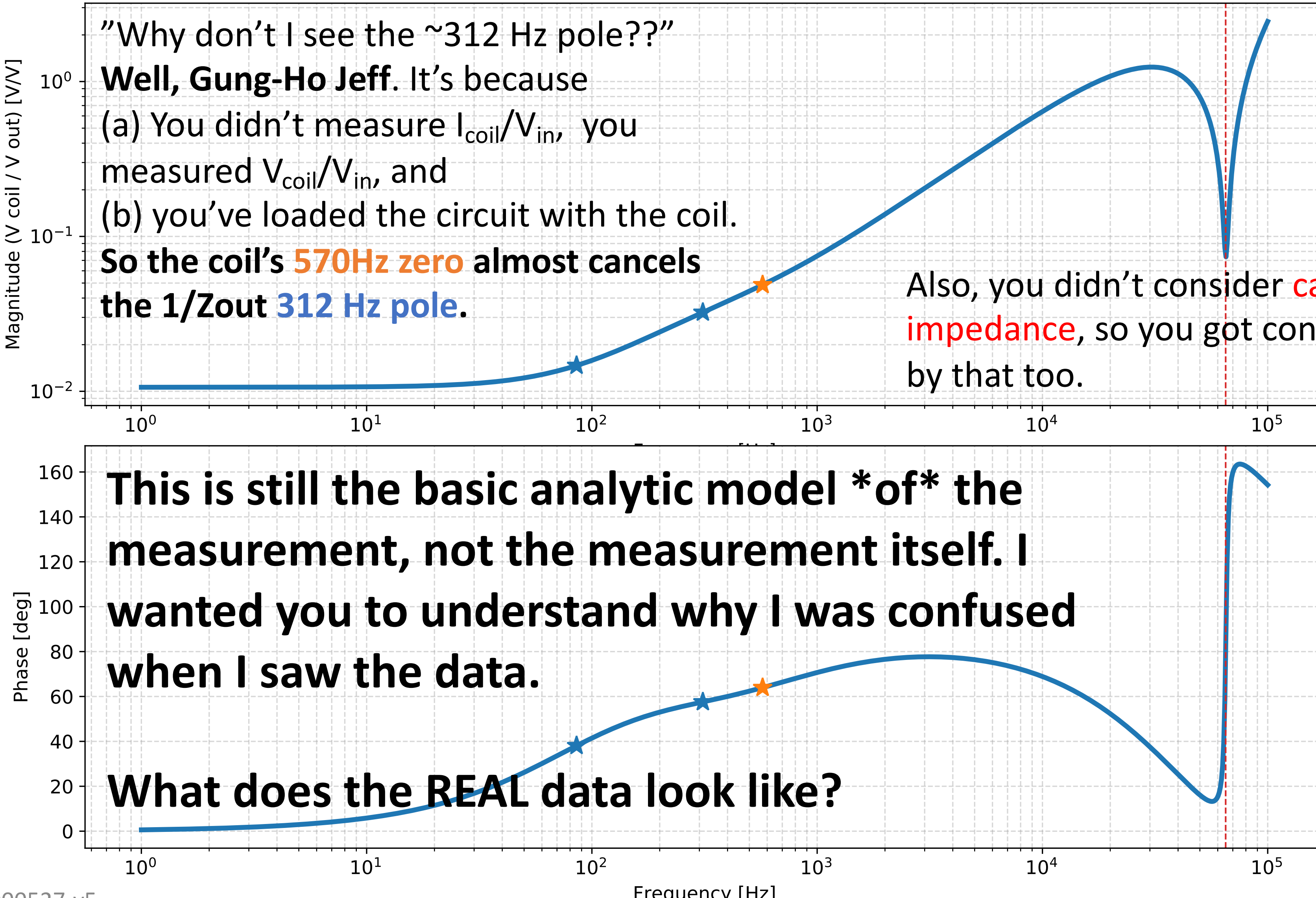
$$f_z^{out} = 1/(2\pi (R_4 + R_5) C_{12}) = 85.110\ Hz$$

flipped because Zout is in the denominator

Which means you're going to be confused for months – YEARS – by your results, until you write this presentation!

I.4 The Measurement: "missing" pole, *really* solved.

$$V_{coil}/V_{in} \approx Z_{coil}/(2Z_{out} + Z_{coil})$$



"Why don't I see the ~312 Hz pole??"
Well, Gung-Ho Jeff. It's because
(a) You didn't measure I_{coil}/V_{in}, you measured V_{coil}/V_{in}, and
(b) you've loaded the circuit with the coil.
So the coil's 570Hz zero almost cancels the 1/Zout 312 Hz pole.

Also, you didn't consider cable impedance, so you got confused by that too.

This is still the basic analytic model *of* the measurement, not the measurement itself. I wanted you to understand why I was confused when I saw the data.

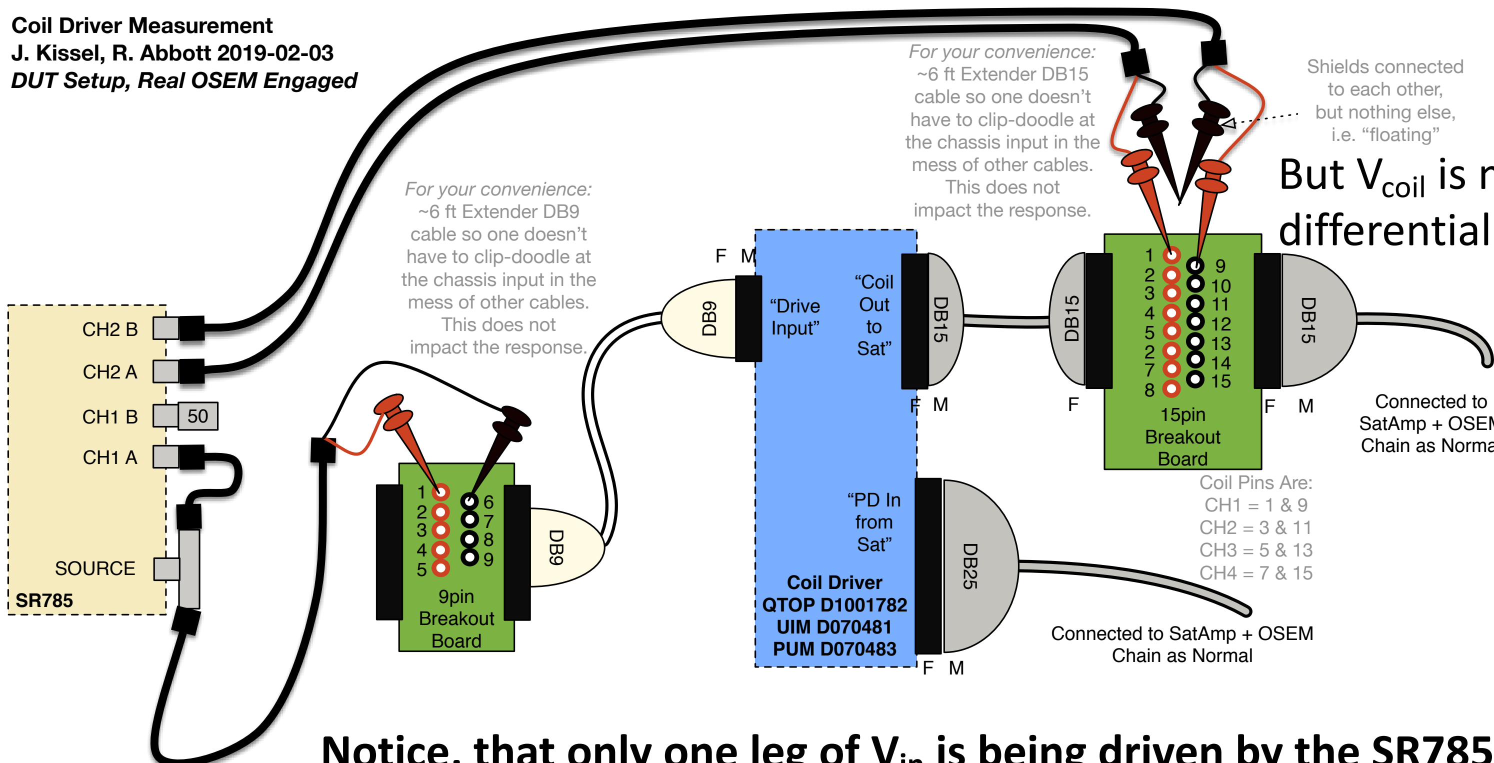
What does the REAL data look like?

I.4 The Measurement: What we really did…

But wait … it gets worse. To quote [LHO:46927]:

"This time (unlike the 2016 attempt; with measurement as shown on the last slide, as in [LHO:24725]) we tried to cut corners by only driving the coil drivers with single-ended input directly from the SR785 -- so we can avoid having to characterize the details of the differential driver box that has been used previously. This failed, causing (what we believe to be saturations) of the coil driver electronics and wonky unphysical*** transfer functions."

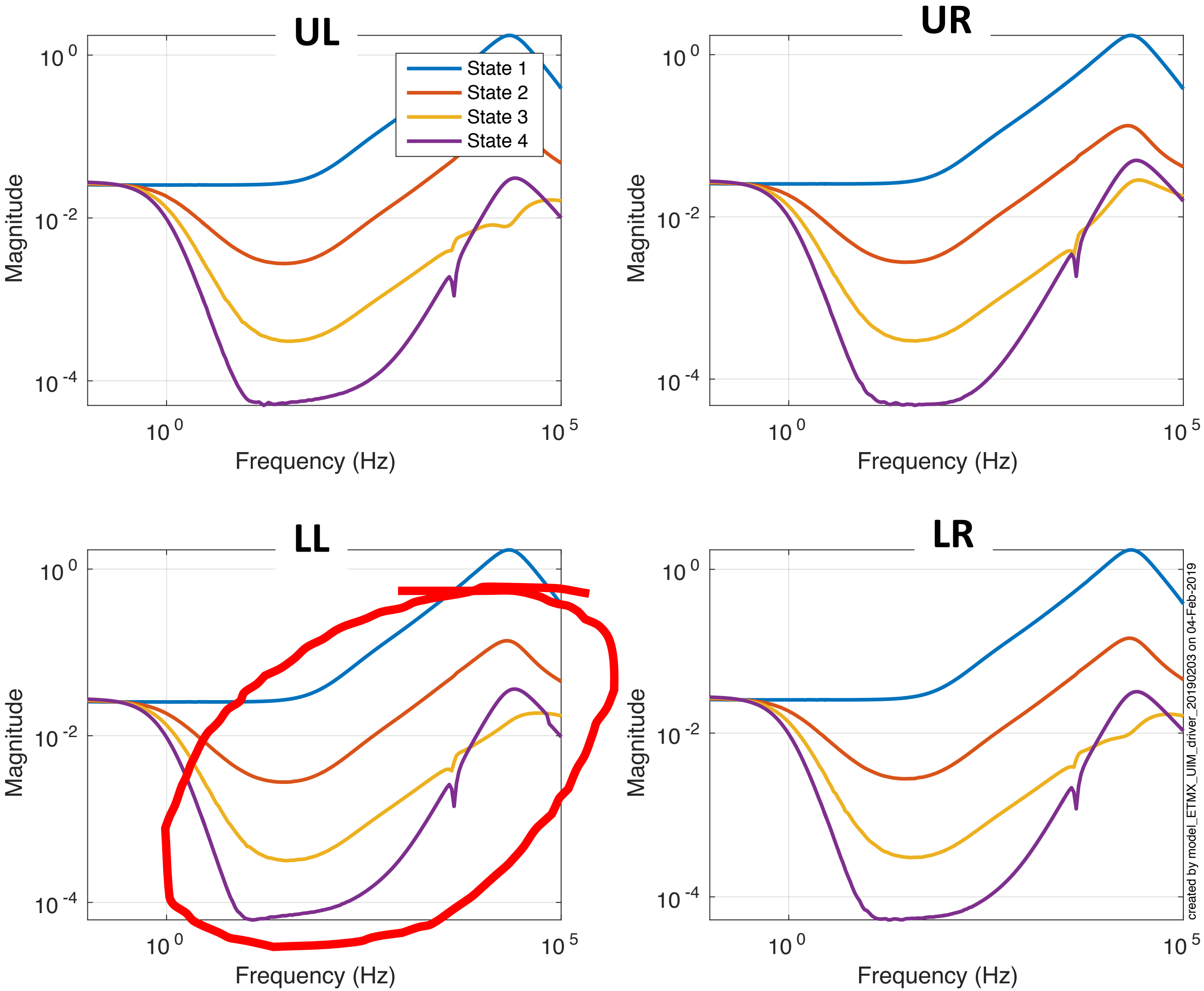
The joys of that Sunday measurement you think will work to save you time…



But V_{coil} is measured differentially…

Notice, that only one leg of V_{in} is being driven by the SR785…

I.4 The Measurement: *** wonky, unphysical TFs



Evan plotted the results 2019-02-03 results the next day (see [LHO:46773]).

Evan apologizes for the lack of tick marks.

Sure, it looks like there's "there's no 300 Hz pole," but we now understand that.

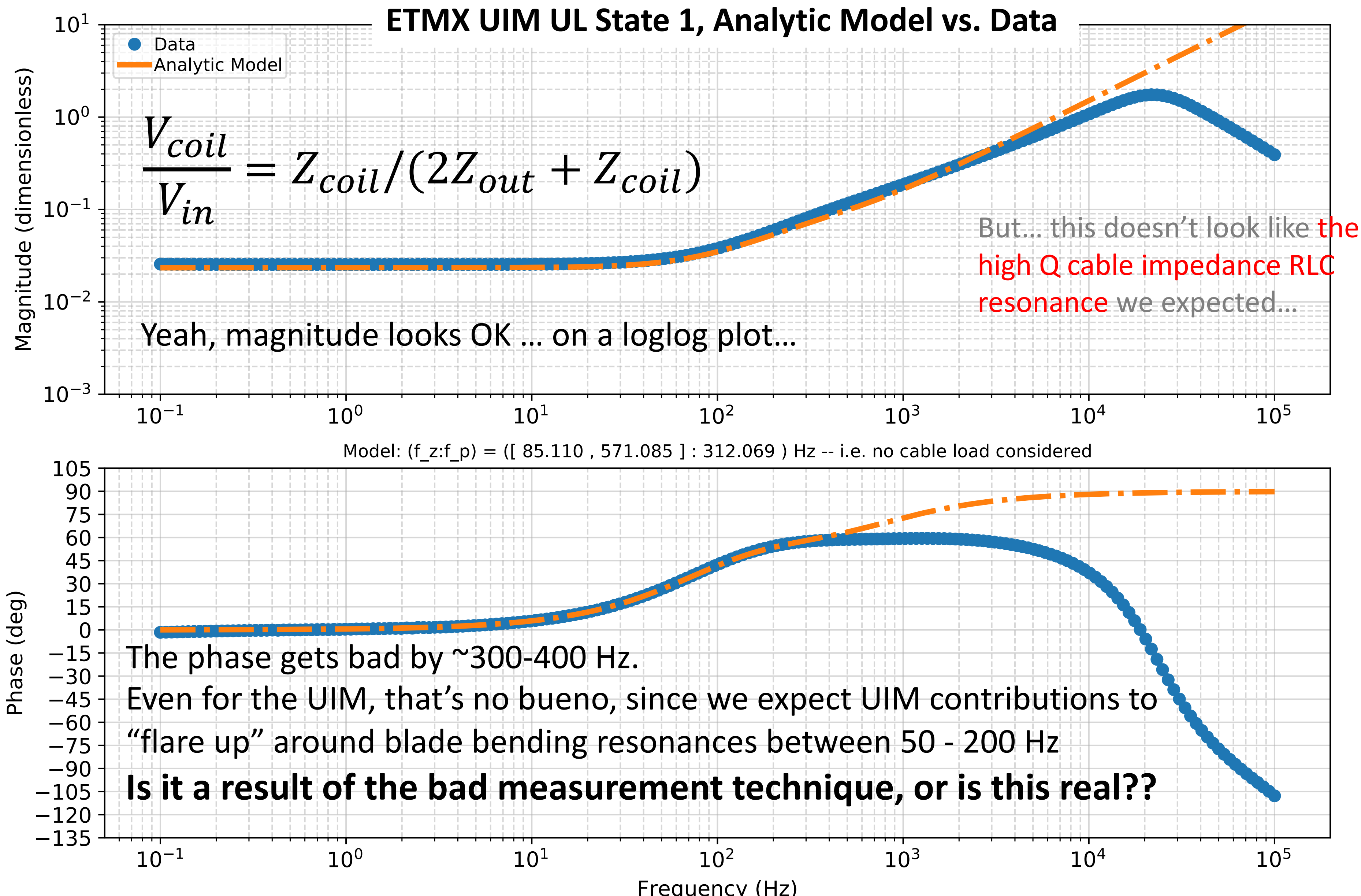
Further, it looks like, for at least State 2, the z:p = 10.5:0.95 Hz low pass shows up, good…

But look at how the magnitude gets distorted at (let's say 500 Hz) and above in States 3 and 4…

^/trunk/Common/Electronics/H1/Data/SUSElectronics/ETMX/UIM/2019-02-03/2019-02-03_UIMdriver_measurementnotes.txt

But … this is the data we have. Maybe we can salvage the data for States 1 and 2?

I.4 The Measurement: Finally, The Data.



Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. **Other models of the circuit**
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.5 Other Models of the Circuit

- **What if we use a more sophisticated model? Can we predict this deviation?** There are lots of more sophisticated modelling tools for circuits out there, LISO, Spice, Altium, etc.

- Chris Wipf put together a LISO model of the UIM circuit in the Noise Budget SVN,
  - https://svn.ligo.caltech.edu/svn/aligonoisebudget/trunk/Dev/SusElectronics/LISO/QUAD/UIM
  - Note that, unfortunately, the LISO models Chris ran didn't export poles and zeros, we so don't have them (we'll find later that re-running to get them won't be worth it)

- It will be instructive to show that model too, especially because
  - More models = more understanding
  - More poles and zeros will appear from the fit than we predict from the analytic model,
  - The LISO model doesn't make approximations for clarity, and
  - The parameters of the cable and coil load are (apparently) quite uncertain

But, also, let's just fit the data.

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. **The Fit and Each Coil Result**
7. Converting fit results in to systematic error in A_{UIM}
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

This parts a four-sub-part doooosey

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

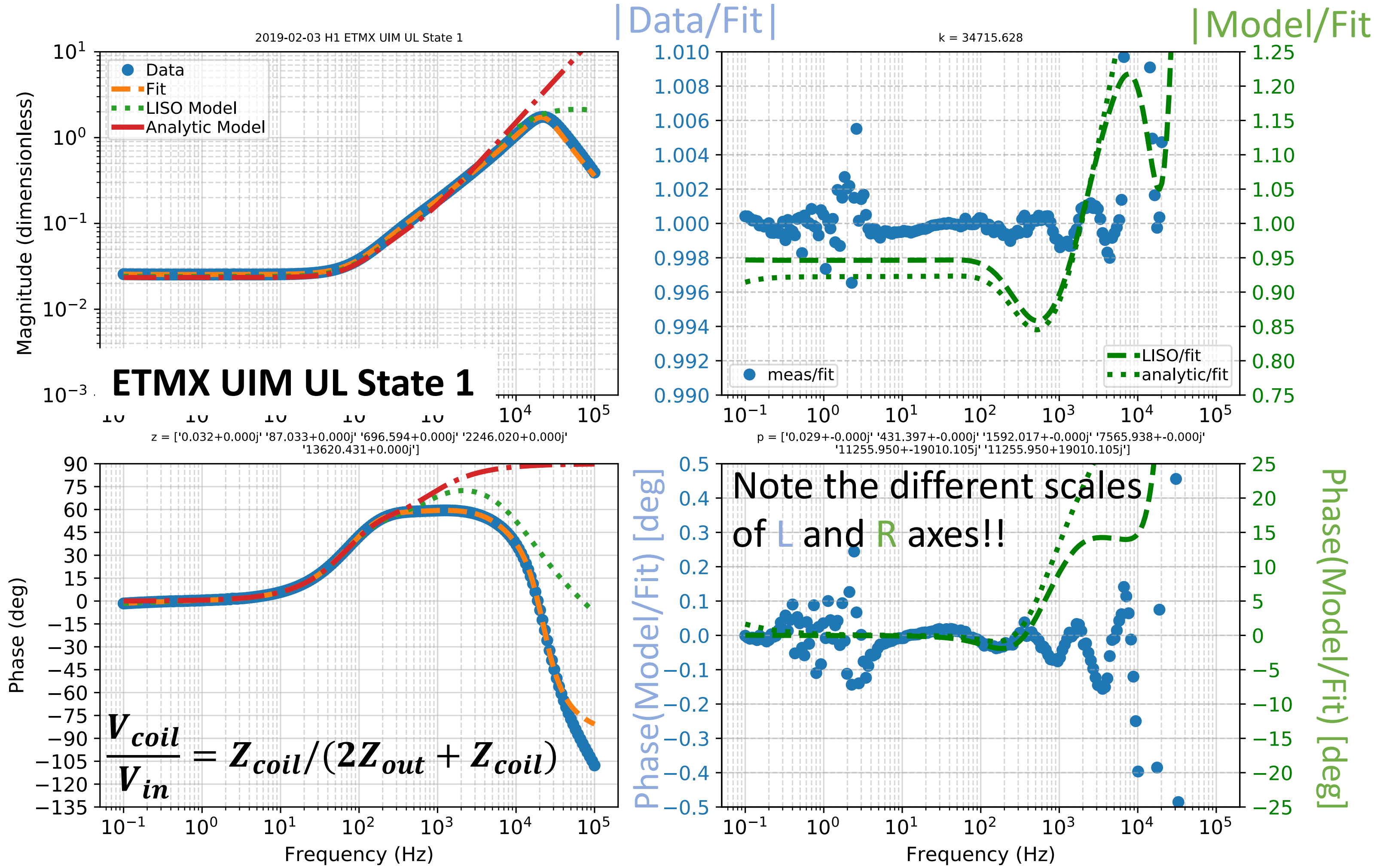
I.6.1 The Fit: IIR Rational is Awesome

- Most transfer function software is unruly: if you don't understand what your data is, or the quality of the data, you're going to have a tough time tailoring the tool to suit your needs, and/or understanding the results.
- A 2016 call to action, G1601173, inspired Lee McCuller to develop IIRrationalv2. I've found it to work excellently, with minimal input.
- The script to run the fit lives here:
- ^/trunk/Common/Electronics/H1/Scripts/fit_ETMX_UIM_driver_20190203_IIRrational_20200401.py
- Here's my environment that I used to get it to work (determined using ^/trunk/Common/Misc/Scripts/versioncheck.py the output of which is quoted here):

```
This is Python version:
3.7.4 (default, Aug 13 2019, 15:17:50)
[Clang 4.0.1 (tags/RELEASE_401/final)]

If you import the following packages, you
will get the versions listed below:
    matplotlib.__version__ = 3.1.1
    numpy.__version__ = 1.17.2
    scipy.__version__ = 1.3.1
    sklearn.__version__ = 0.21.3
    gwpy.__version__ = 1.0.1
    nds2.__version__ = 0.16.5
    IIRrational.__version__ = 2.0.11
    h5py.__version__ = 2.9.0
    emcee.__version__ = 3.0.2
    corner.__version__ = 2.0.1
```

I.6.1 The Fit Results Per Coil: Intro to Plot

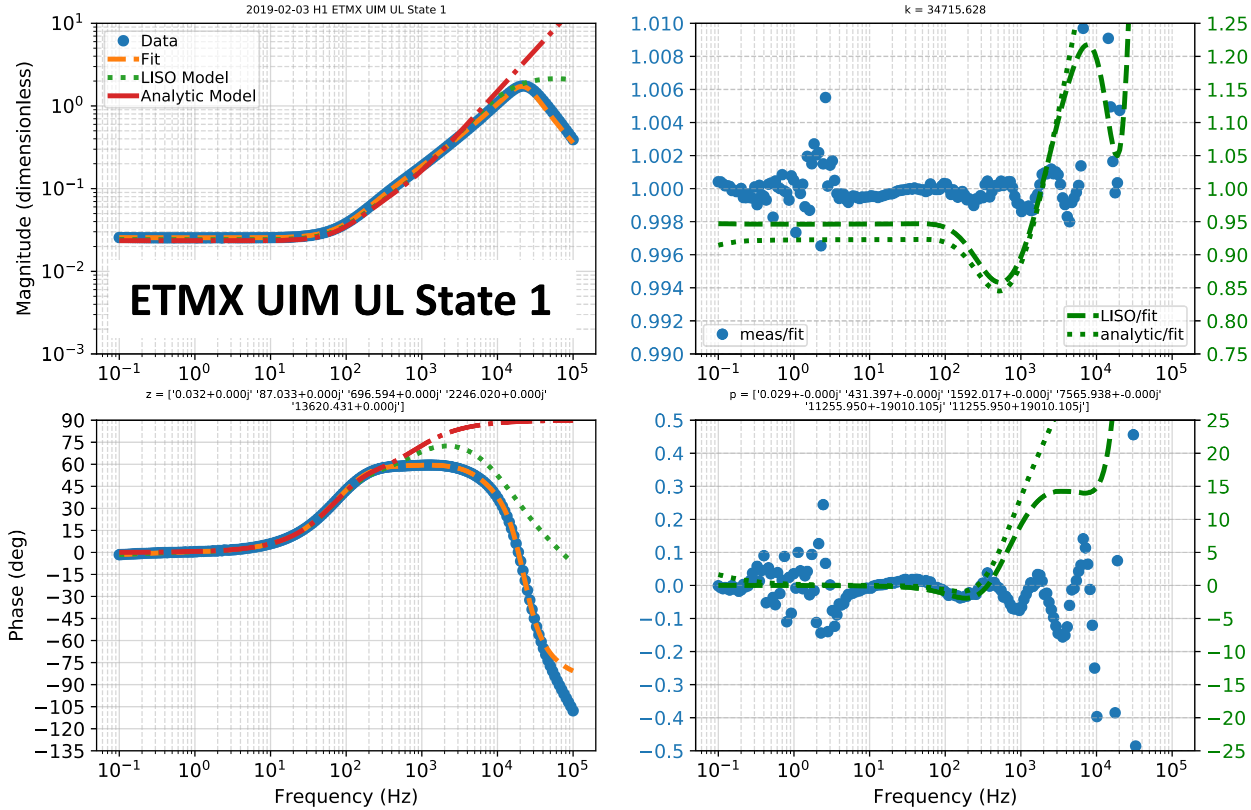


- The **basic analytic model** is as bad as we know from slide 27 (residual shown in dotted green)
- The **LISO model** seems to miss the basic RC and Coil pole and zero frequencies, resulting in magnitude error of ~15% by 300 Hz, and also bad in phase (residual, in dashed green, is 10 deg by 1 kHz).
- The **IIRrational fit** is excellent, all the way out to 10 kHz. But … let's look at all the poles and zeros it returns …

I.6.1 Fit per Coil: State 1 Results Interpretation

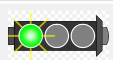
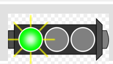
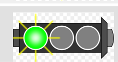
Again, understand the fit results is an important part of the game:

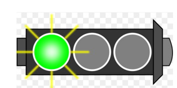
- Do zeros and poles make sense?
- Are there more than you expect?
- Are results consistent across several coils?
- Can we ignore any?



Take UL for example:

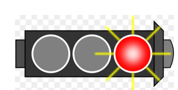
$$\frac{V_{coil}}{V_{in}} = Z_{coil}/(2Z_{out} + Z_{coil})$$

Circuit Feature Assignment	UL Fit Zeros	UL Fit Poles
Coil Impedance	**696.5942 Hz** 	
RC Network	**87.0329 Hz** 	**431.3965 Hz** 
SW Closed LP	**0.0325 Hz** 	**0.0293 Hz** 
?????	**2246.0201 Hz** 	**1592.0174** 
Cable impedance?		**pair(22092.54 Hz, 59.37 deg)** 



f_z or f_p , or combo is right about where we expect. Or the fz:fp is close enough to "canceling" to ignore

f_z or f_p is probably from feature X, but is pretty far off from expected, or maybe high frequency enough to not matter



WUT

I.6.1 Fit per Coil: State 1 Results Summary

Circuit Feature Assignment	UL Fit Zeros	UL Fit Poles
Coil Impedance	696.5942 Hz	
RC Network	87.0329 Hz	431.3965 Hz
SW Closed LP	0.0325 Hz	0.0293 Hz
????	2246.0201 Hz	1592.0174
Cable impedance?		pair(22092.54 Hz, 59.37 deg)

Circuit Feature Assignment	UR Fit Zeros	UR Fit Poles
Coil Impedance	671.7041 Hz	
RC Network	85.9533 Hz	422.2943 Hz
SW Closed LP	No fit?	No fit?
????	2337.1901 Hz	5132.4934 Hz
????	pair(12262.2781 Hz, 15.218 deg) pair(12822.8952 Hz, 21.5666)	pair(11037.6219 Hz, 61.3485 deg)
????	19443.5355	
Cable impedance?		pair(21731.503 Hz, 73.7415 deg)

$$f_z^{coil} = 1/(2\pi L_{coil}/R_{coil}) = 571.085\ Hz$$

$$f_p^{out} = 1/(2\pi R_4 C_{12}) = 312.069\ Hz$$

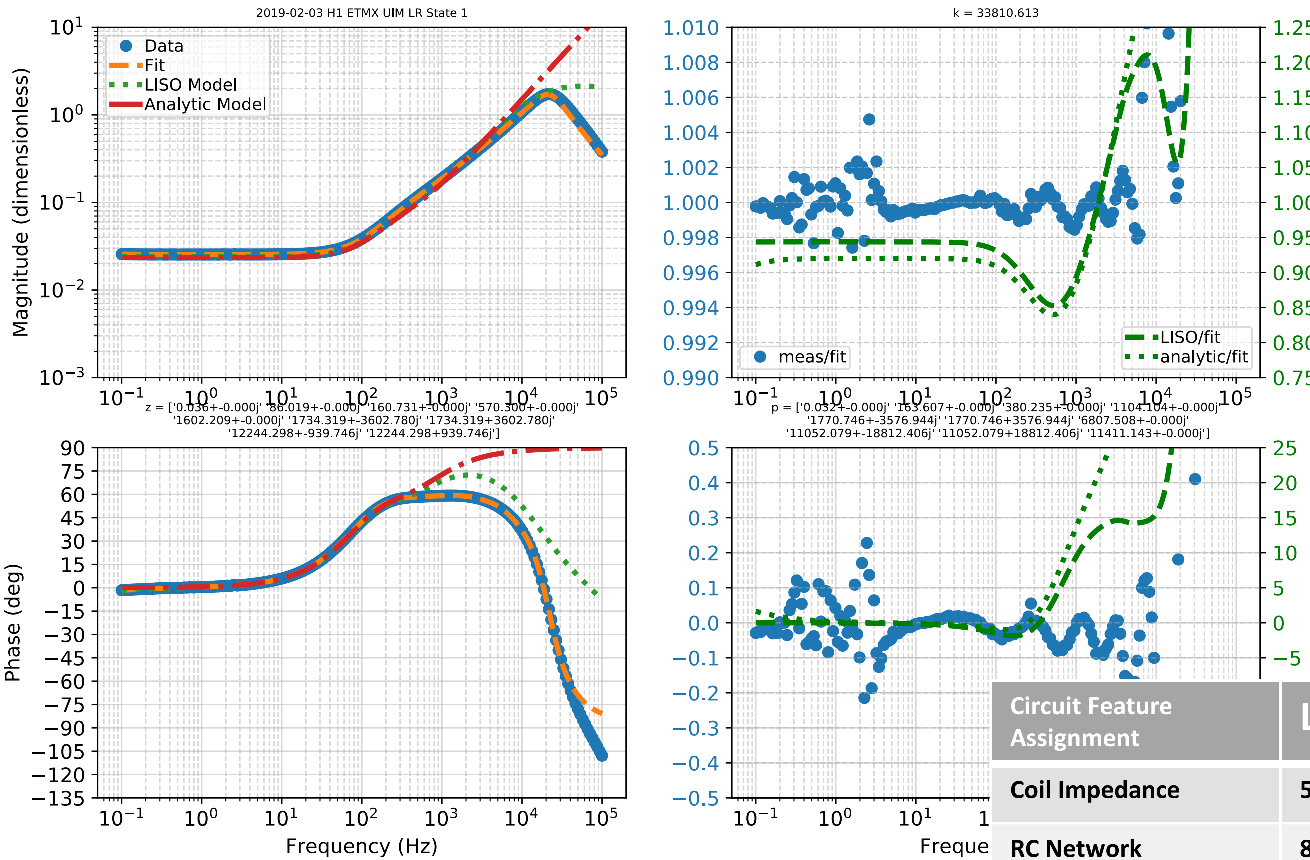
$$f_z^{out} = 1/(2\pi(R_4+R_5)C_{12}) = 85.110\ Hz$$

Circuit Feature Assignment	LL Fit Zeros	LL Fit Poles
Coil Impedance	699.0254 Hz	
RC Network	86.5228 Hz	427.0135 Hz
SW Closed LP	No fit?	No fit?
????	2315.2727, 5247.6252 Hz	1623.5029, pair(5943.6595, 10.6624 deg)
Cable impedance?		pair(21390.090 Hz, 58.138 deg)

Circuit Feature Assignment	LR Fit Zeros	LR Fit Poles
Coil Impedance	570.3 Hz	
RC Network	86.019 Hz	380.235 Hz
SW Closed LP	0.036 Hz	0.032 Hz
????	160.731 Hz	1104.104 Hz
Nearly canceling	pair(3998.485 Hz, 64.2946 deg)	pair(3991.249Hz, 63.6625 deg)
????	pair(12280.307 Hz, 4.400 deg)	6807.508, 11411.143
Cable impedance?		pair(21818.686 Hz, 59.566 deg)

$$f_p^{coil||cable} = 65.063\ kHz$$

I.6.1 Fit per Coil: State 1 Results LR



Circuit Feature Assignment	LR Fit Zeros	LR Fit Poles
Coil Impedance	570.3 Hz	
RC Network	86.019 Hz	380.235 Hz
SW Closed LP	0.036 Hz	0.032 Hz
???	160.731 Hz	1104.104 Hz
Nearly canceling	pair(3998.485 Hz, 64.2946 deg)	pair(3991.249Hz, 63.6625 deg)
???	pair(12280.307 Hz, 4.400 deg)	6807.508, 11411.143
Cable impedance?		pair(21818.686 Hz, 59.566 deg)

Just to show that the data, the fit, or the fit residuals for the LR "poster child" don't look any different, but the fit has poles and zeros right where we expect, AND some ones that we don't expect at all.

I.6.1 The Fit per Coil: State 1 Results Discussion

- Why is LR the poster child, with f_z:f_p = (86, 570: 380) Hz, where the other three are consistently f_z:f_p = (86, 690: 430)?
  - Let's assume that, for whatever reason, the three coils – though not as expected, are fit at real values. 430 vs. 380 Hz means R4, C12 values are in question, and 690 vs. 570 Hz means R_{coil} or L_{coil} are in question.
  - Let's assume we know the resistances well at (R4, R_{coil}) = (750,42.7) Ohm. That means (C12,L_{coil}) are actually ~(0.49e-6 F, 9.8 mH) instead of the drawing/cannon values of (0.68e-6 F, 11.9 mH).
  - Plausible…
- What are all of these mid- kHz poles and zeros? Can we get by with ignoring the fit results above 1 kHz?
  - Is this a manifestation of the bad measurement / saturation?
- Why is the cable impedance so low in frequency and so low in Q?
  - Is *this* a manifestation of the bad measurement / saturation?

I.6.1 The Fit per Coil: What's next?

You feel I'm in the weeds. I know. *I* feel I'm in the weeds. How can we come back up for air? Look at some more weeds.

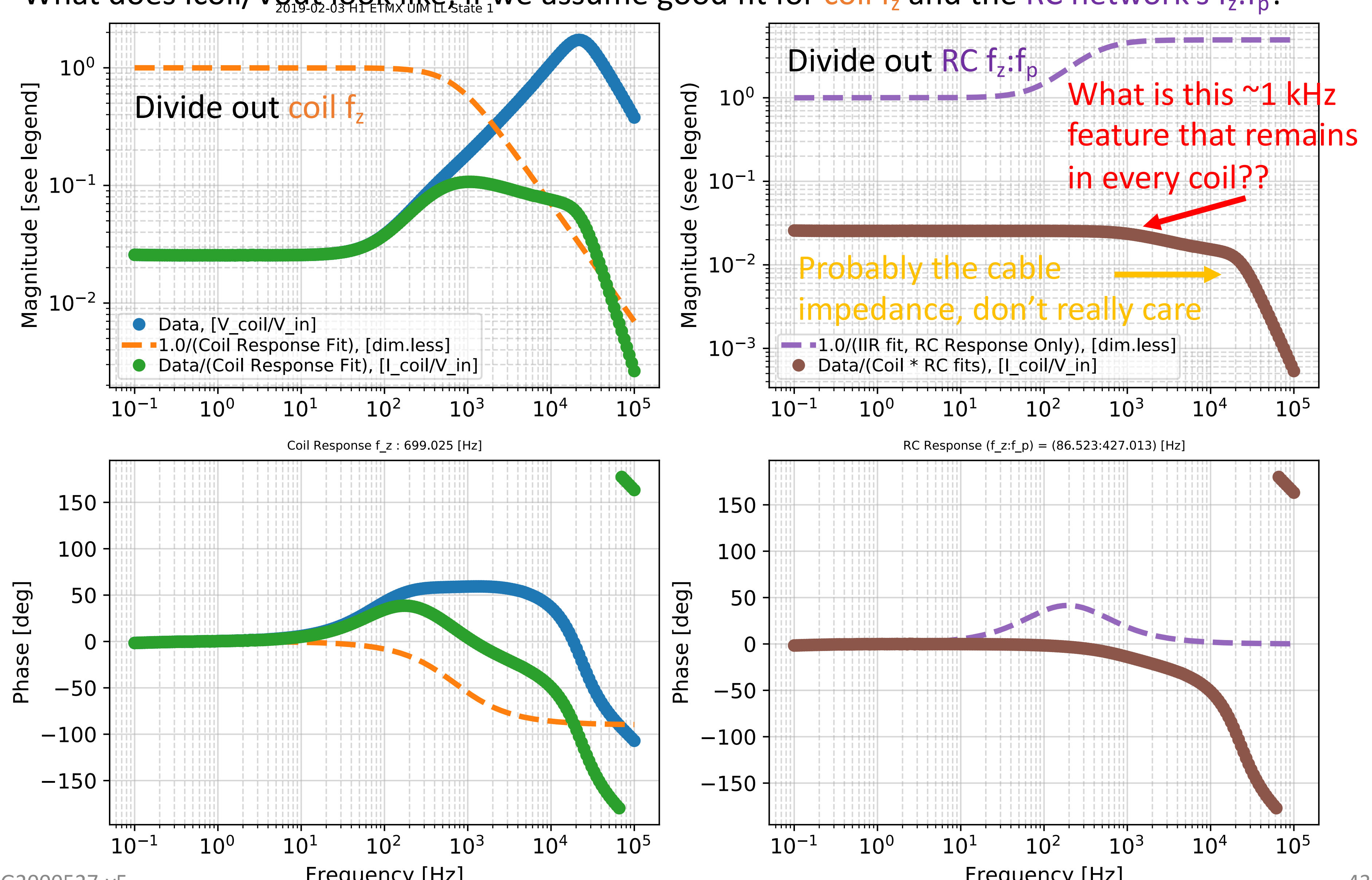
2. We can blindly assume that the fit is perfect for all coils. If so, we'd use the value of the coil f_z, divide it out of the V_{coil} / V_{in} data, and look at the I_{coil} / V_{in} transfer function. Does it make sense? Should we bother (re)fitting *that* data?

3. Look at the ratio of State 2 to State 1. Is getting the low pass f_z:f_p pair from that is as easy as we expect?

4. Look (and fit) at state 2 by itself. Does the data match the State 1 fit * (State 2 / State 1) fit?

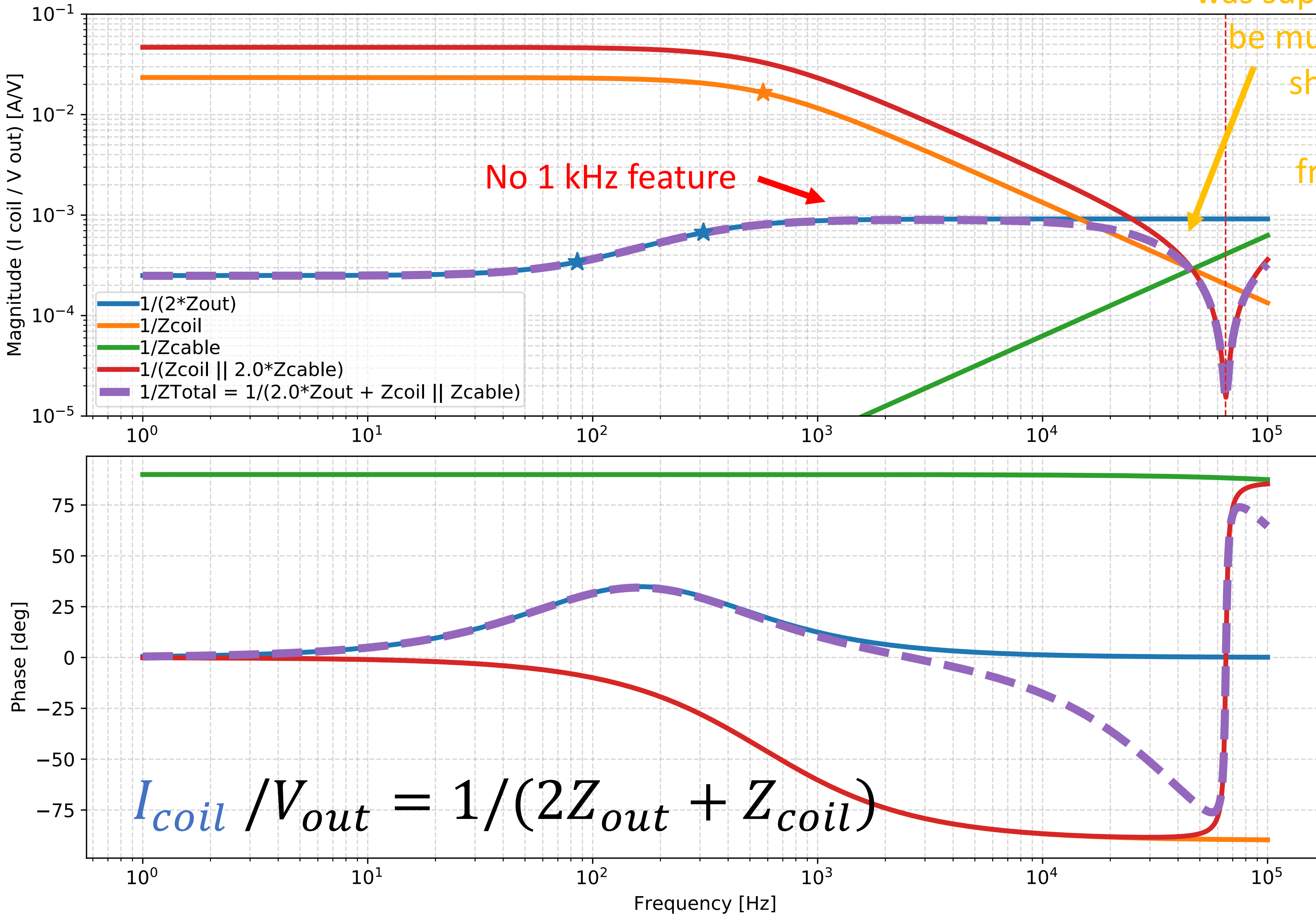
I.6.2 Fit per Coil: State 1 I_{coil} / V_{out}, taking out “knowns”

What does Icoil/Vout look like, if we assume good fit for coil f_z and the RC network’s f_z:f_p?



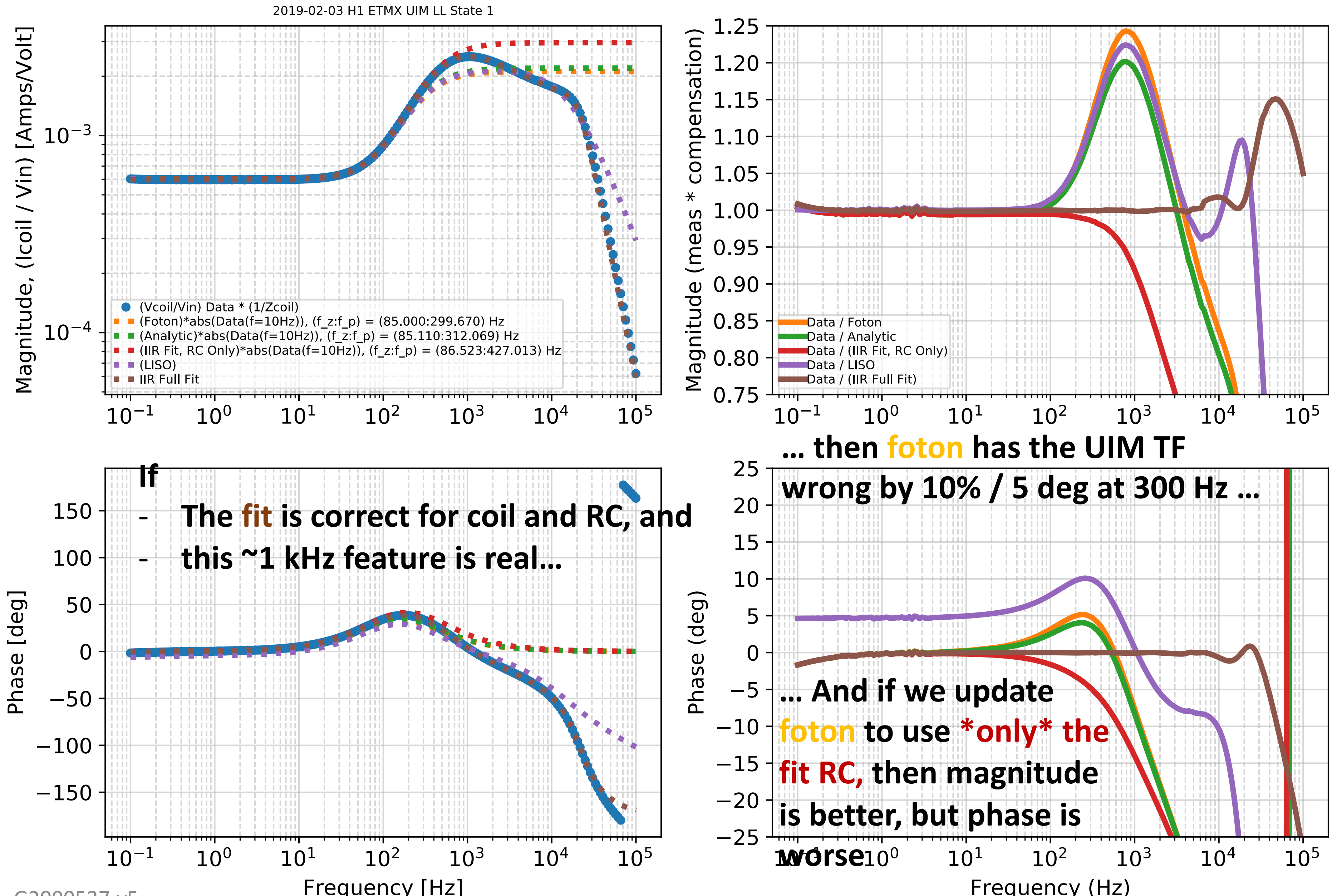
I.6.2 Fit per Coil: State 1 remember our expectations?

Green Solid on previous slide should look like **Purple dashed** here



$$I_{coil}/V_{out} = 1/(2Z_{out} + Z_{coil})$$

I.6.2 Fit per Coil: State 1 How bad would it be?



If
- The **fit** is correct for coil and RC, and
- this ~1 kHz feature is real…

… then foton has the UIM TF wrong by 10% / 5 deg at 300 Hz …

… And if we update foton to use *only* the fit RC, then magnitude is better, but phase is worse

I.6.1 The Fit per Coil: What's next?

You feel I'm in the weeds. I know. *I* feel I'm in the weeds. How can we come back up for air? Look at some more weeds.

2. We can blindly assume that the fit is perfect for all coils. If so, we'd use the value of the coil f_z, divide it out of the V_{coil} / V_{in} data, and look at the I_{coil} / V_{in} transfer function. Does it make sense? Should we bother (re)fitting *that* data?

   Conclude: There's really something weird with this data, manifesting at 1-2 kHz

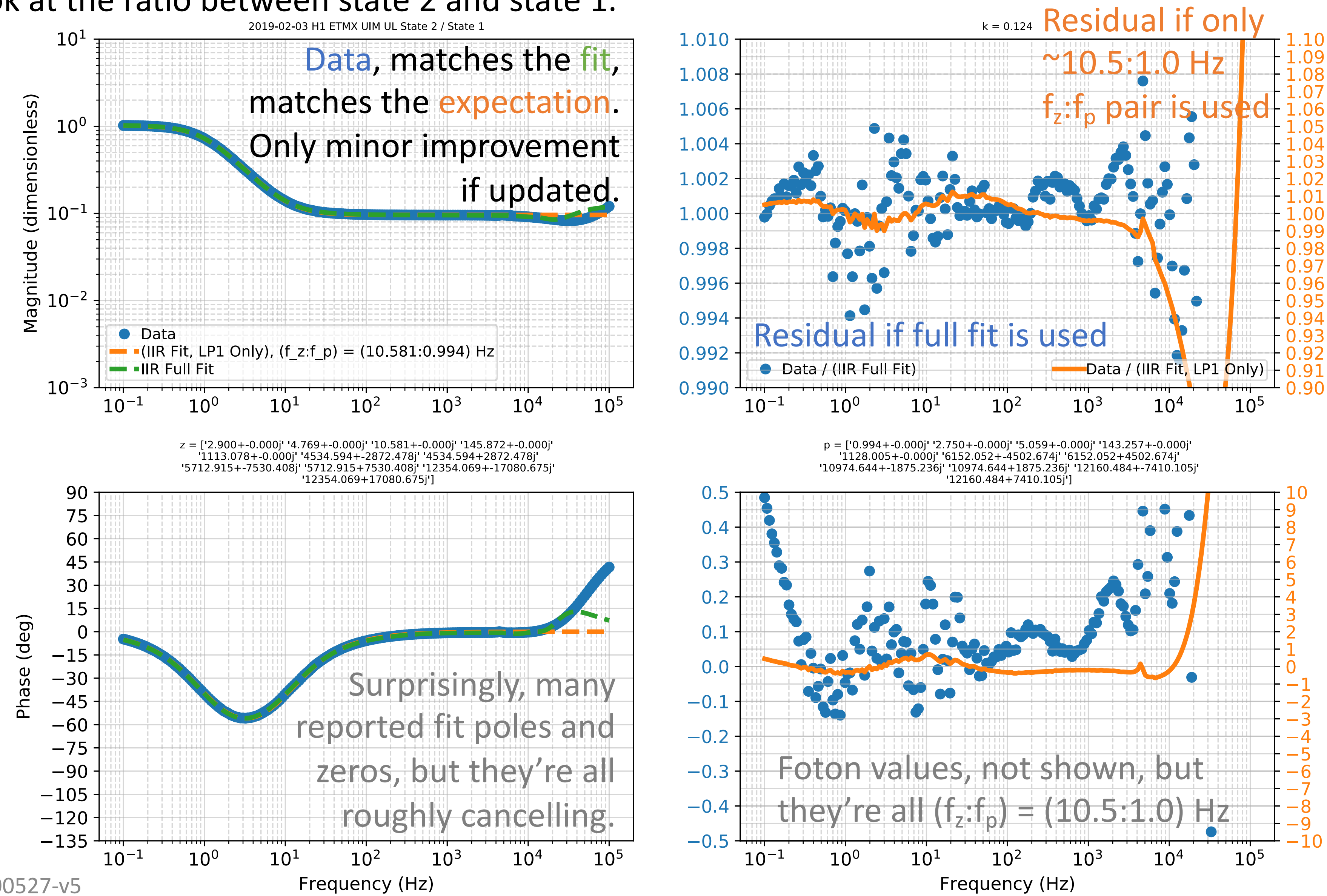
3. Look at the ratio of State 2 to State 1. Is getting the low pass f_z:f_p pair from that is as easy as we expect?

4. Look (and fit) at state 2 by itself. Does the data match the State 1 fit * (State 2 / State 1) fit?

I.6.3 Fit per Coil: State 2/State 1: the LP1 zs and ps

OK. Need some air. Does the analog data we have make any sense? It does.

Look at the ratio between state 2 and state 1.



I.6.3 Fit per Coil: State 2/State 1: Results Summary

UL	Fit Zeros	Fit Poles
Nearly cancelling	2.89952 Hz	2.75042 Hz
Nearly canceling	4.76888 Hz	5.05923 Hz
SW Closed LP	**10.5814 Hz**	**0.99443 Hz**
Nearly canceling	145.8720 Hz	143.2566 Hz
Nearly canceling	1113.0777 Hz	1128.0047 Hz
????	Pair(5367.8369 Hz, 32.3526 deg)	Pair(7623.7668 Hz, 36.2003 deg)
????	Pair(9452.2185 Hz, 52.8143 deg)	Pair(11133.7022 Hz, 9.6965 deg)
????	Pair(21080.1439 Hz, 54.1226 deg	Pair(14240.3312 Hz, 31.3564 deg)

UR	Fit Zeros	Fit Poles
SW Closed LP	**10.3314 Hz**	**0.98556 Hz**
Nearly canceling	52.4826 Hz	51.6746 Hz
Nearly canceling	344.4865 Hz	341.47236 Hz
Nearly canceling	2137.9732 Hz	2163.3018 Hz
????	3211.9846 Hz	5833.4346 Hz
????	pair(4802.2480 Hz, 32.800 deg)	4409.4475 Hz, 5019.2094 Hz
????	pair(11329.7897 Hz, 52.0737 deg	Pair(14962.1253 Hz, 39.2143 deg
????	pair(24347.3447 Hz, 57.547 deg	15509.4456 Hz, 17175.5778 Hz

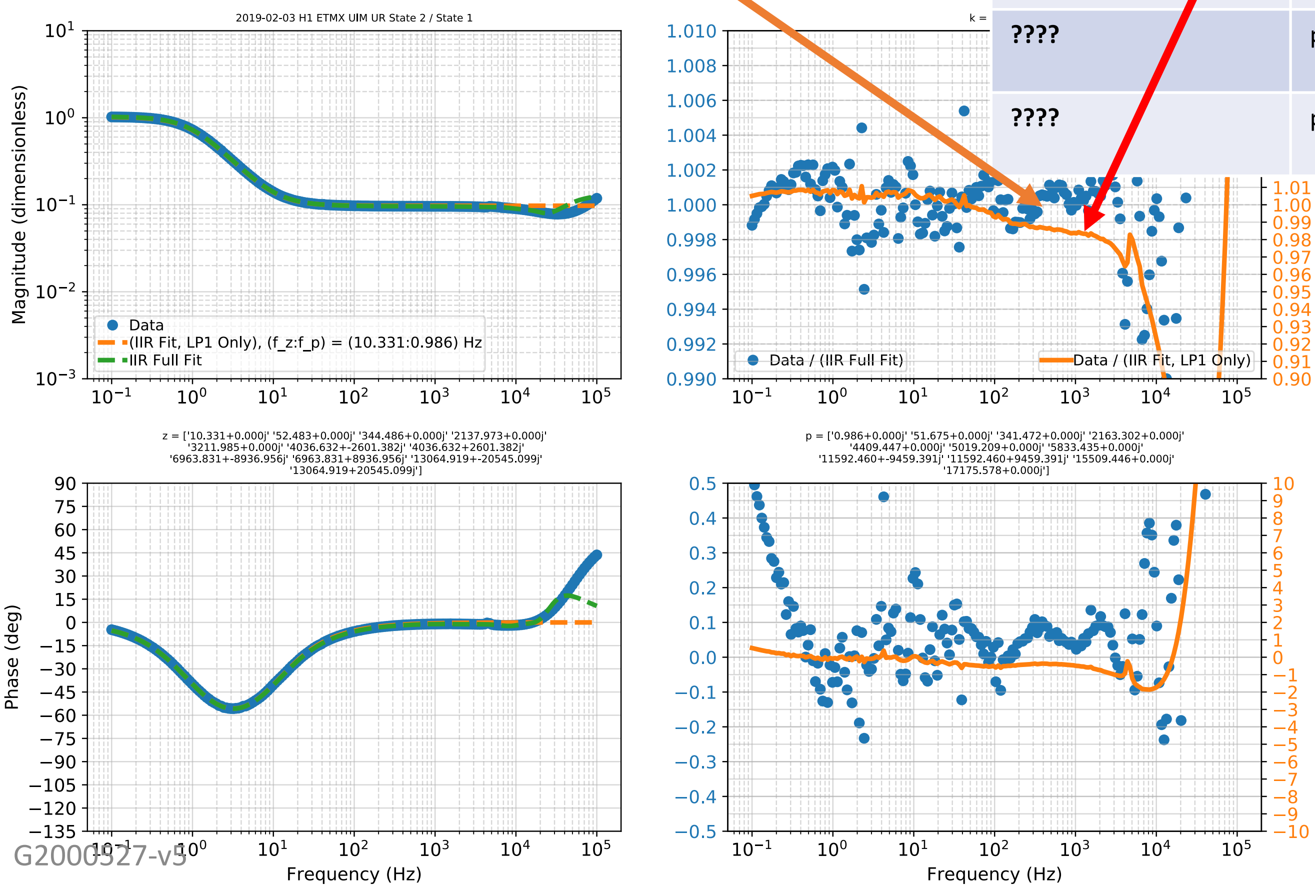
LL	Fit Zeros	Fit Poles
SW Closed LP	**10.3830 Hz**	**0.9820 Hz**
Nearly canceling	61.3351 Hz	60.2293 Hz
Nearly canceling	282.9903 Hz	291.4153 Hz
Nearly canceling	Pair(643.6467 Hz, 11.4552 deg)	630.7973 Hz, 654.7394 Hz
????	Pair(5512.6344 Hz, 39.6531 deg)	Pair(6718.1860 Hz, 65.2573 deg)
????	Pair(7085.8327 Hz, 66.5343 deg)	10061.4559 Hz, 10891.6712 Hz
Nearly canceling	Pair(13638.8270 Hz, 63.1878 deg)	Pair(13419.9763 Hz, 26.1267 deg)
????	Pair(24657.6748 Hz, 61.7029 deg	Pair(15566.9234 Hz, 55.0929 deg)

LR	Fit Zeros	Fit Poles
Nearly cancelling	0.05932 Hz	0.06045 Hz
SW Closed LP	**10.4728 Hz**	**0.98792 Hz**
Nearly canceling	93.0036 Hz	91.4280 Hz
Nearly canceling	1522.7636 Hz	1579.1637 Hz
????	Pair(4255.9612 Hz, 29.7771 deg) Pair(8332.2720 Hz, 54.3682 deg)	5443.8019 Hz, 8077.0312 Hz, Pair(11032.6047 Hz, 39.2195 deg)
Nearly canceling	Pair(13237.5898 Hz, 61.0969 deg)	Pair(13752.7035 Hz, 39.9270 deg)
????	Pair(25155.6858 Hz, 59.6633 deg)	Pair(13801.8767 Hz, 29.9008 deg)

I.6.3 Fit per Coil: State 2 / State 1 Oddball -- UR

Only UR is of concern with the residual of "if we ignore everything but the fit fz:fp that closely matches the expected low pass frequencies" exceeding 1% in magnitude above 100 Hz…

UR	Zeros	Poles
SW Closed LP	**10.3314 Hz**	**0.98556 Hz**
Nearly canceling	52.4826 Hz	51.6746 Hz
Nearly canceling	344.4865 Hz	341.47236 Hz
Nearly canceling	2137.9732 Hz	2163.3018 Hz
????	3211.9846 Hz	5833.4346 Hz
????	pair(4802.2480 Hz, 32.800 deg)	4409.4475 Hz, 5019.2094 Hz
????	pair(11329.7897 Hz, 52.0737 deg	Pair(14962.1253 Hz, 39.2143 deg
????	pair(24347.3447 Hz, 57.547 deg	15509.4456 Hz, 17175.5778 Hz



But … as you'll see (and what is often said with details of these studies): we've got bigger fish to fry…

I.6.1 The Fit per Coil: What's next?

You feel I'm in the weeds. I know. *I* feel I'm in the weeds. How can we come back up for air? Look at some more weeds.

2. We can blindly assume that the fit is perfect for all coils. If so, we'd use the value of the coil f_z, divide it out of the V_{coil} / V_{in} data, and look at the I_{coil} / V_{in} transfer function. Does it make sense? Should we bother (re)fitting *that* data?

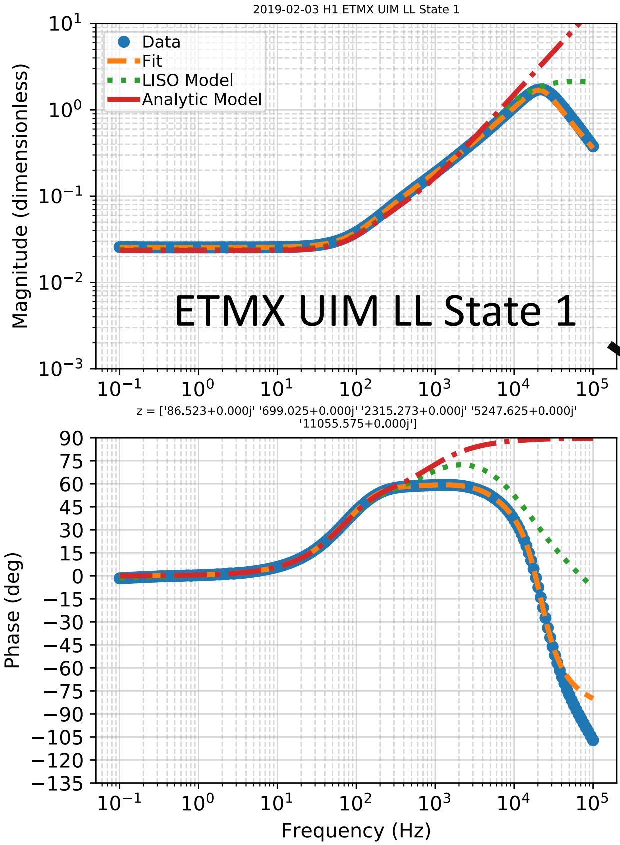
   Conclusion: There's really something weird with this data, manifesting at 1-2 kHz

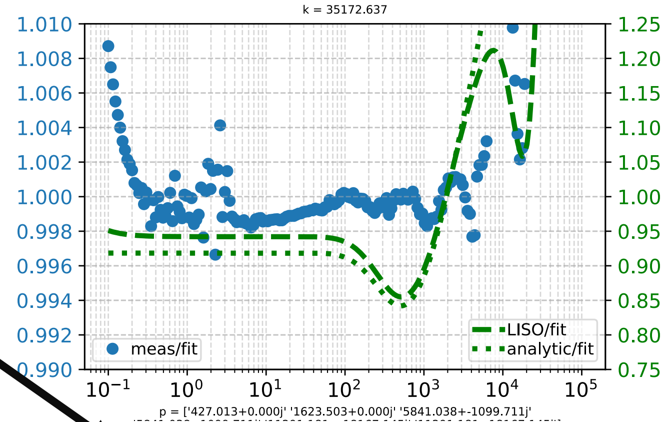
3. Look at the ratio of State 2 to State 1. Is getting the low pass f_z:f_p pair from that is as easy as we expect?

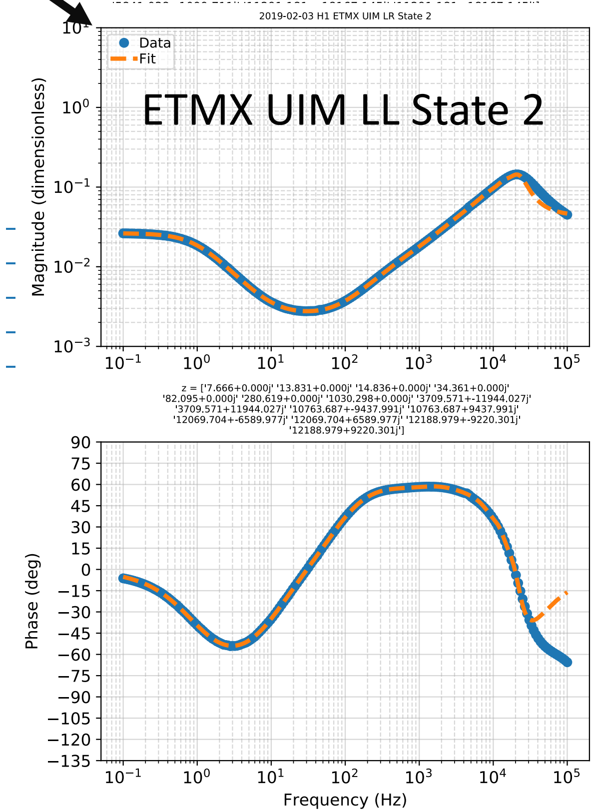
   Conclusion: Yes, we can safely extract the fit low pass f_z:f_p pair.

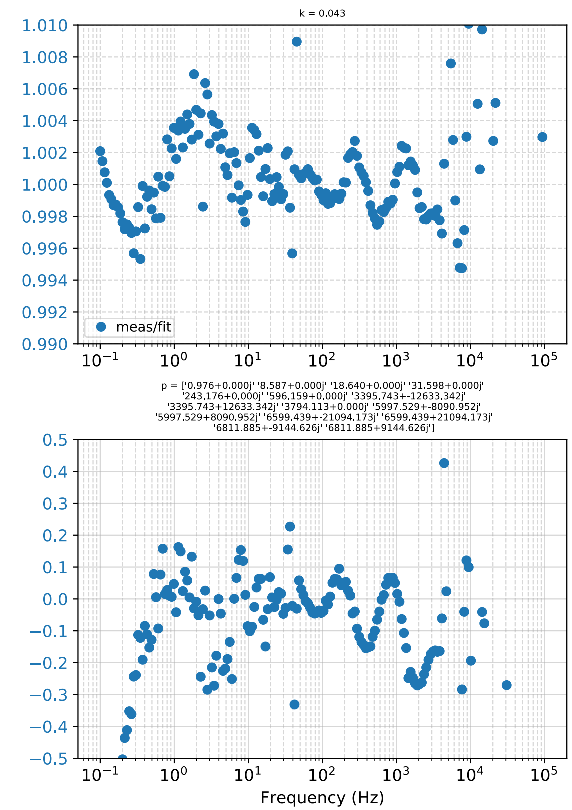
4. Look (and fit) at state 2 by itself. Does the data match the State 1 fit * (State 2 / State 1) fit?

I.6.4 Fit per Coil: State 2 vs State (1) and (2/1) Fits



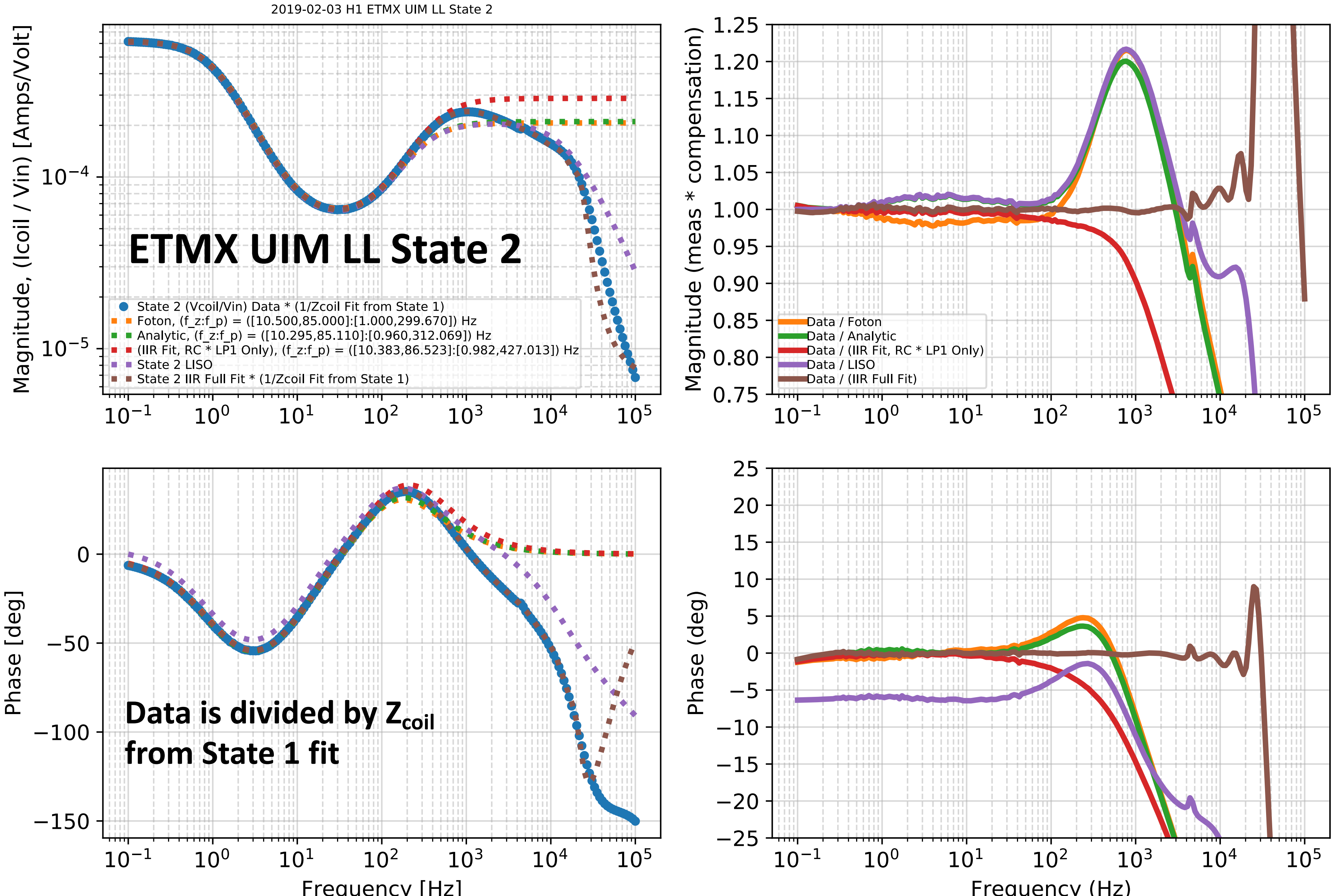




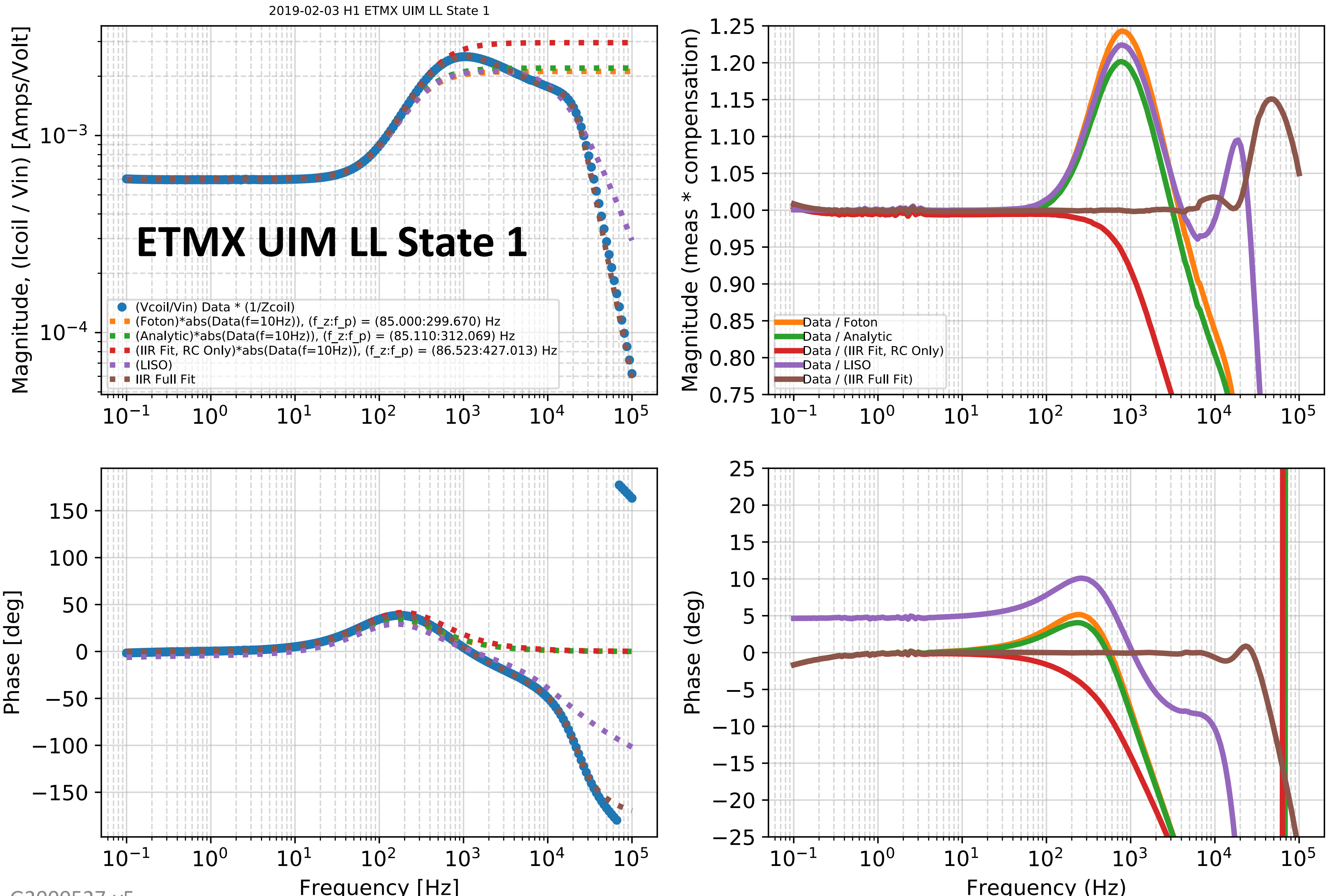


Since the ratio behaved so much like expected, State 2 by itself is probably going to look like the product of the State 1 results and the State2/State1, **and it does.**

I.6.4 Fit per Coil: State 2 I_{coil} / V_{in} Residuals



I.6.4 Fit per Coil: Remember State 1…



I.6.4 Fit per Coil: Fit answer Comparison: UL and LL

State 1 fit and State 2/1 fit results

Circuit Feature Assignment	UL Fit Zeros	UL Fit Poles
Coil Impedance	**696.5942 Hz**	
RC Network	**87.0329 Hz**	**431.3965 Hz**
SW Closed LP	**10.5814 Hz**	**0.99443 Hz**
?????	2246.0201 Hz	1592.0174
Cable impedance?		pair(22092.54 Hz, 59.37 deg)

Hrmm… State 2 fit f_z:f_p numbers are pretty different from State 1 fit and State 2/1 fit, except for LP1 values

Circuit Feature Assignment	LL Fit Zeros	LL Fit Poles
Coil Impedance	**699.0254 Hz**	
RC Network	**86.5228 Hz**	**427.0135 Hz**
SW Closed LP	**10.3830 Hz**	**0.9820 Hz**
?????	2315.2727, 5247.6252 Hz	1623.5029, pair(5943.6595, 10.6624 deg)
Cable impedance?		pair(21390.090 Hz, 58.138 deg)

State 2 fit results

Circuit Feature Assignment	UL Fit Zeros	UL Fit Poles
Nearly canceling	0.028847 Hz	0.026747 Hz
Coil Impedance	**842.2736 Hz**	
RC Network	**89.2645 Hz**	**472.0885 Hz**
SW Closed LP	**10.3110 Hz**	**0.97697 Hz**
?????	4401.5227 Hz	2798.8233 Hz
Cable impedance?		pair(21118.6667 Hz, 42.1804 deg)

Circuit Feature Assignment	LL Fit Zeros	LL Fit Poles
Nearly canceling	0.37481	0.36443
Nearly canceling	6.43273	6.36145
Nearly canceling	183.9139	200.7849
Coil Impedance	**549.8213**	
RC Network	**89.9268**	**346.3705**
SW Closed LP	**10.2956**	**0.9999998**
????	1632.2068	1177.1166 Hz,
????	pair(15713.5244 Hz, 42.2756 deg) 15762.0667 Hz	3750.3363 Hz, pair(7843.1654 Hz. 52.3709 deg),
Cable impedance?	pair(20052.8982Hz. 26.0454 deg) 21877.4091 Hz, 22594.0678 Hz	pair(13919.8163 Hz. 69.7893 deg) pair(22669.5375 Hz, 77.5484 deg)

I.6.4 Fit per Coil: Fit answer Comparison: UR and LR

Circuit Feature Assignment	UR Fit Zeros	UR Fit Poles
Nearly canceling	40.5700 Hz	39.1471 Hz
Nearly canceling	112.8337 Hz	101.2331 Hz
Coil Impedance	**767.4099 Hz**	
RC Network	**77.2791 Hz**	**455.1022 Hz**
SW Closed LP	**10.2308 Hz**	**0.97447**
?????	pair(5414.0903 Hz, 57.7074 deg) pair(8543.0235 Hz, 31.6995 deg) pair(12415.6958 Hz, 20.2785 deg)	2200.0659 Hz pair(4588.0341 Hz, 51.4882 deg) pair(6168.5520 Hz, 63.2901 deg)
Nearly canceling	pair(11209.1738 Hz, 2.010 deg)	pair(11298.7088 Hz, 65.2882 deg)
Cable impedance?		pair(21411.5152 Hz, 71.3476 deg)

Circuit Feature Assignment	LR Fit Zeros	LR Fit Poles
Coil Impedance	**280.6193 Hz**	
RC Network	**82.0951**	**243.1757 Hz**
SW Closed LP	**13.8309**	**0.97606 Hz,**
Nearly canceling	7.66634 Hz	8.58653 Hz
Nearly canceling	34.3607 Hz,	31.5980 Hz
?????	14.8359 Hz 1030.2985 pair(12506.8258 Hz, 72.7462 deg) pair(13751.5651 Hz, 28.6342 deg) pair(14315.4683 Hz, 41.2455 deg)	18.6402 Hz 596.1594 3794.1125 Hz pair(13081.7578 Hz, 74.9549 deg) pair(10071.4374 Hz, 53.4518 deg) pair(11402.8925 Hz,

Circuit Feature Assignment	UR Fit Zeros	UR Fit Poles
Coil Impedance	**671.7041 Hz**	
RC Network	**85.9533 Hz**	**422.2943 Hz**
SW Closed LP	**10.3314 Hz**	**0.98556 Hz**
????	2337.1901 Hz pair(12262.2781 Hz, 15.218 deg) pair(12822.8952 Hz, 21.5666)	5132.4934 Hz pair(11037.6219 Hz, 61.3485 deg)
????	19443.5355	
Cable impedance?		pair(21731.503 Hz, 73.7415 deg)

Circuit Feature Assignment	LR Fit Zeros	LR Fit Poles
Coil Impedance	**570.3 Hz**	
RC Network	**86.019 Hz**	**380.235 Hz**
SW Closed LP	**10.4728 Hz**	**0.98792 Hz**
????	160.731 Hz	1104.104 Hz
Nearly canceling	pair(3998.485 Hz, 64.2946 deg)	pair(3991.249Hz, 63.6625 deg)
????	pair(12280.307 Hz, 4.400 deg)	6807.508, 11411.143
Cable impedance?		pair(21818.686 Hz, 59.566 deg)

I.6.1 The Fit per Coil: What's next?

You feel I'm in the weeds. I know. *I* feel I'm in the weeds. How can we come back up for air? Look at some more weeds.

1. We can blindly assume that the fit is perfect for all coils. If so, we'd use the value of the coil f_z, divide it out of the V_{coil} / V_{in} data, and look at the I_{coil} / V_{in} transfer function. Does it make sense? Should we bother (re)fitting *that* data?

   Conclusion: **There's really something weird with this data, manifesting at 1-2 kHz**

2. Look at the ratio of State 2 to State 1. Is getting the low pass f_z:f_p pair from that is as easy as we expect?

   Conclusion: Yes, we can safely extract the fit low pass f_z:f_p pair.

3. Look (and fit) at state 2 by itself. Does the data match the State 1 fit * (State 2 / State 1) fit?

   Conclusions: **Sort of.** The residuals have same mysterious 1-2 KHz features from State 1, **but the poles and zeros are astoundingly different, some more like expected, some just wrong, with no general trends as each coil is different.**

I.6 Fit per Coil: Grand Conclusions

- We definitely, definitely, definitely need to get good measurements.
  - We should always drive the drivers, and measure the response differentially.
- Unfortunately, we can't assume each coil channel is going to be even roughly the same, and we may get conflicting answers between what should be the same answers when switching between states.
  - e.g. State 2 / State 1 for for LP1 is not the same as State 2 alone
  - So we should be prepare to the "two clocks" situation, where don't know which to choose.
- Make the data going in to the fitter as simple as possible, when it makes physical sense to do so.
  - Never, ever, ever take measurements with the coil as a part of the measurement. Just put a no-capacitance, 40 Ohm dummy OSEM "across the back" of the driver as the "coil" "load" impedance.
  - That also means that we can't use the FAST_I_MONs measurements either -- not because "they don't measure the output network" -- but because they include the coil impedance which drastically confuses the even the best fitting routines
  - We should perform the same analytical analysis on PUM driver vs. the AOSEM to confirm Zcoil << Zout…. another day.

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. **Converting fit results in to systematic error in A_{UIM}**
8. Converting sys error in A_{UIM} to sys error in R and Conclusions

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.7 Converting Individual Coil Fit Results in to Systematic Error in A_{UIM}

- Let's assume we understood and we're happy with everything from section I.6.
  - **Remember: we're not, but let's move on anyways, because this is the data we have.**

- The individual coil results must be used retroactively to predict what error was caused in the *total* longitudinal actuation strength in the UIM.

You can think of it like this:

$$A_U = E2O * \begin{pmatrix} F_{UL} \\ \frac{F_{LL}}{F_{UR}} \\ F_{LR} \end{pmatrix} * \mathrm{DAC} * \ \mathrm{AI} \ * \begin{pmatrix} CD_{UL} \\ \frac{CD_{LL}}{CD_{UR}} \\ CD_{LR} \end{pmatrix} * M * S_U$$

I.7 Fit per Coil >> Error in A_{UIM}: Reality

Or like this:

$$F_{ii}(f) = E2O_{ii} * D_{ii}\,(f) * DAC_{ii} * \; AI_{ii} \; * TC_{ii} * CD_{ii}\,(f) * M_{ii}$$

$$A_{UIM} = S_U(f) * \sum_{ii} F_{ii}$$

where $ii = UL, LL, UR, LR$, and for each coil chain, the actuation strength of each driver/coil/magnet chain, $\boldsymbol{F_{ii}}$, has the following components:

- $\boldsymbol{E2O}$ is the Euler 2 OSEM matrix (exactly 0.25 for each coil),
- $\boldsymbol{D(f)}$ is the normalized digital compensation "COILOUTF" filter for each coil,
- $\boldsymbol{DAC}$, $\boldsymbol{AI}$, and $\boldsymbol{TC}$ are the digital-to-analog converter gain, anti-aliasing filter, and DC transconductance of the coil driver respectively
- $\boldsymbol{CD(f)}$ is the normalized coil driver response,
- $\boldsymbol{M}$ is the magnet strength, and
- $\boldsymbol{S_U}$ is the UIM longitudinal force to TST displacement transfer function response

Ideally, $D_{ii}(f)$ would be the perfect inverse of $CD_{ii}(f)$ for every coil, they would cancel to a unity transfer function and we can exclude it from any model.

That's what we've done for the UIM in the calibration group's DARM loop model.

However, the frequency dependent systematic error in A_{UIM} arises when $D_{ii}(f)$ doesn't perfectly invert $CD_{ii}(f)$, and the fact that the frequency dependent error from each stage is *summed* means that error is not easily intuitable from the individual chain error.

I.7 Fit per Coil >> Error in A_{UIM}: Model

$$F_{ii}(f) = E2O_{ii} * D_{ii}\,(f) * DAC_{ii} * \;AI_{ii}(f)\; * TC_{ii} * C_{ii}\,(f) * M_{ii}$$

$$A_{UIM} = S_U(f) * \sum_{ii} F_{ii}$$

So we need to construct a model with these terms explicitly included.
Let's take the above, and assume everything in between $\boldsymbol{D_{ii}}$ and $\boldsymbol{C_{ii}}$ for each chain (namely DAC_{ii}, $AI_{ii}(f)$, and TC_{ii}) is only a common gain to all four chains. This is an OK assumption because

- we take some effort (eg LHO:42740) to "balance" the gain of each path to minimize length to angle coupling.
- the AI filter response, $AI(f)$, which is a 16kHz elliptic lowpass, in general doesn't start to deviate from "just a gain" until several kHz, and each channel would only have a small difference at that. Including the measured differences is an exercise for some other day.

$$F_{ii}(f) = E2O * DAC * AI(f) * TC * M * D_{ii}(f)\; * C_{ii}\,(f)$$

$$A_{UIM} = S_U(f) * E2O * DAC * AI(f) * TC * M * \sum_{ii} D_{ii}\,(f)\; * C_{ii}\,(f)$$

Under this assumption, the systematic error, η_{UIM}, can be computed using only what we already have!

$$\eta_{UIM} = \frac{A_{UIM}^{(\text{“no” sys. error})}}{A_{UIM}^{(w/\ sys. error)}} = \frac{A_{UIM}^{(well-compensated)}}{A_{UIM}^{(poorly-compensated)}} = \sum_{ii} \frac{\left[C_{ii}^{fit}\right]^{-1} C_{ii}^{meas}}{\left[C_{ii}^{foton}\right]^{-1} C_{ii}^{meas}} = \sum_{ii} \frac{C_{ii}^{foton}}{C_{ii}^{fit}}$$

I.7 Fit per Coil >> Error in A_{UIM}: Model

- But wait! Remember the whole reason we got in to this game was to find out what error was caused by *switching* from State 1 to State 2,

- So we should also compute

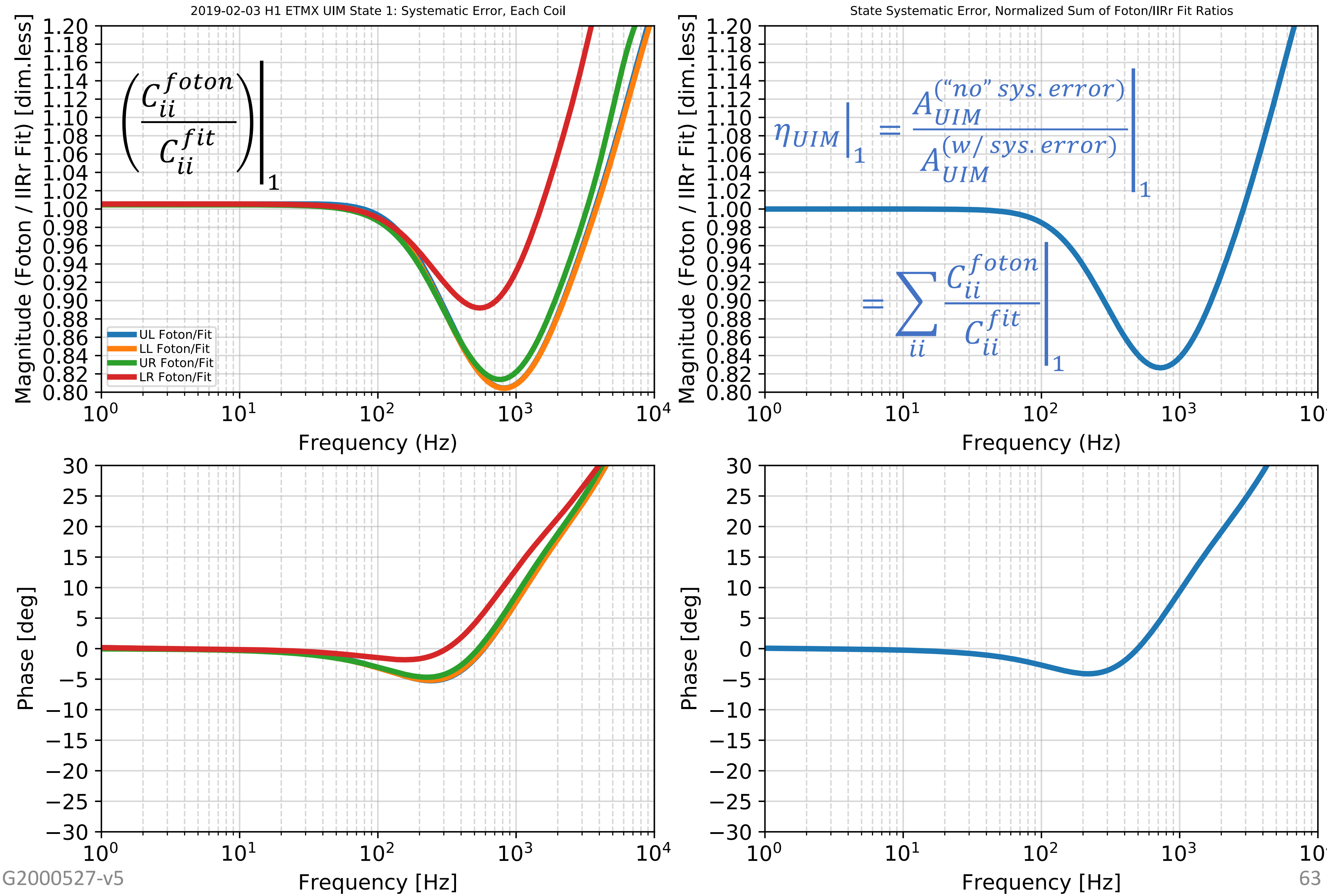
$$\frac{\eta_{UIM}|_2}{\eta_{UIM}|_1} = \frac{\sum_{ii}\left(C_{ii}^{foton}/C_{ii}^{fit}\right)\Big|_2}{\sum_{ii}\left(C_{ii}^{foton}/C_{ii}^{fit}\right)\Big|_1}$$

such that we'll know, not only the systematic error under "normal" operation (i.e. in state 1), but also during this Nov 27 – Dec 03 2019 time period.

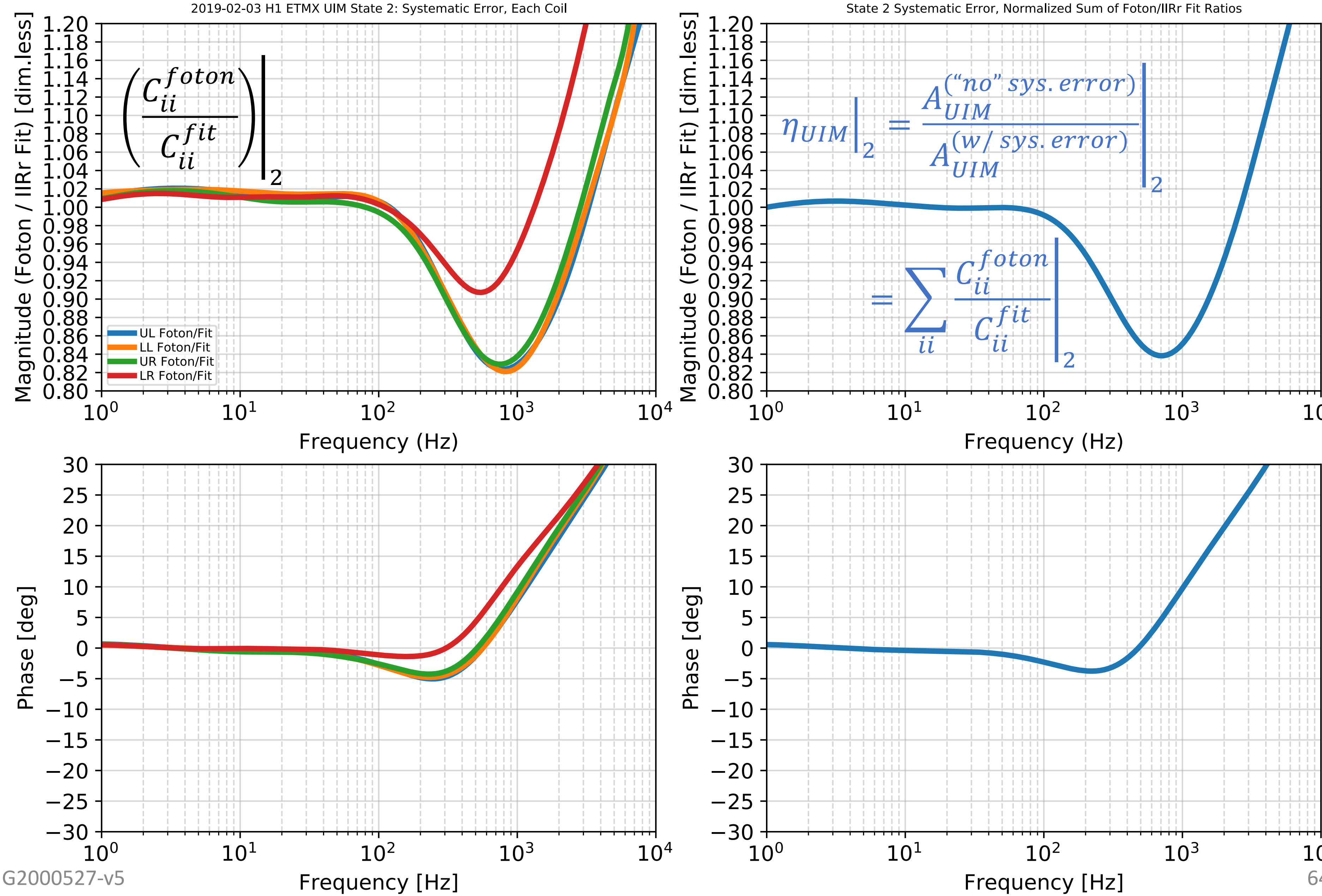
$$A_{UIM}^{(\text{“}no\text{”}\ sys.error)}(most\ times) = \eta_{UIM}|_1 A_{UIM}(20200113\ Model)$$

$$A_{UIM}^{(\text{“}no\text{”}\ sys.error)}(Nov\ 27 - Dec) = \eta_{UIM}\Big|_2 A_{UIM}(Nov\ 27 - Dec\ 03)$$
$$= \eta_{UIM}|_1 \left(\frac{\eta_{UIM}|_2}{\eta_{UIM}|_1}\right) A_{UIM}(20200113\ Model)$$

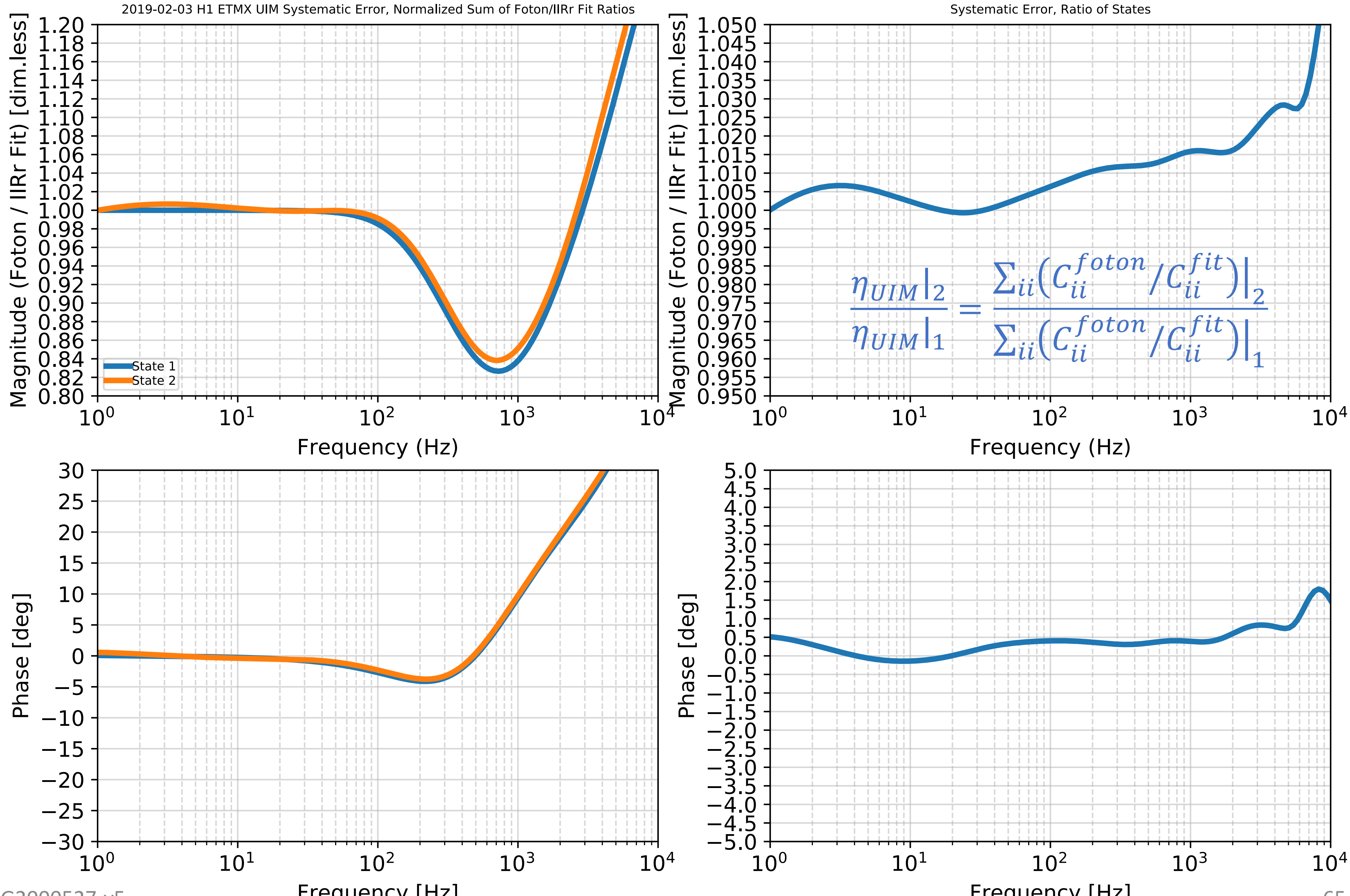
I.7 Fit per Coil >> Error in A_{UIM}: State 1 Results



I.7 Fit per Coil >> Error in A_{UIM}: State 2 Results



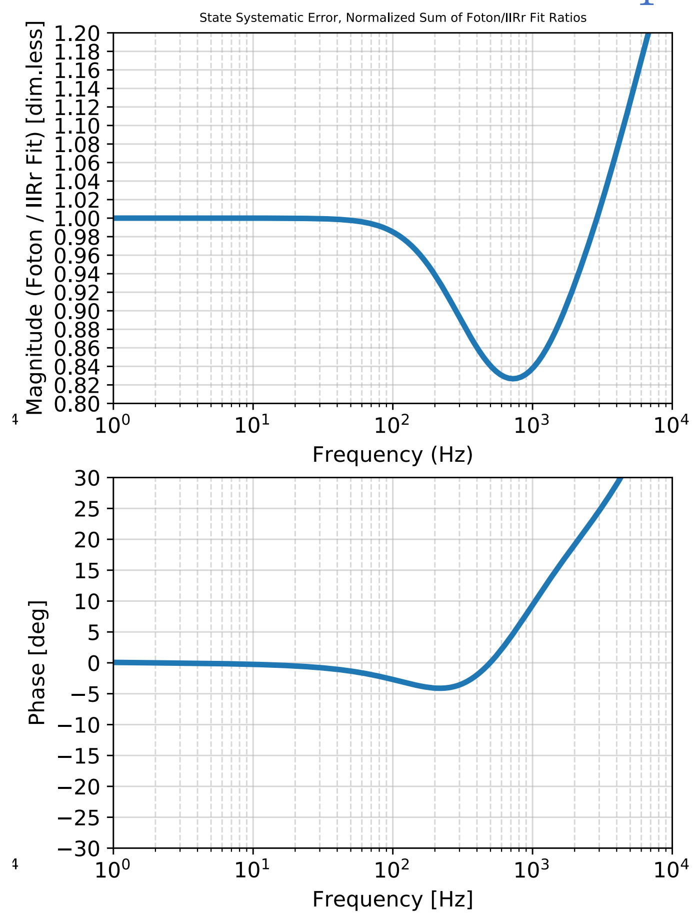
I.7 Fit per Coil >> Error in A_{UIM}: Results Compared



I.7 Fit per Coil >> Error in A_{UIM}: Discussion

$$\eta_{UIM}\Big|_1 = \frac{A_{UIM}^{(\text{"no" } sys.\,error)}}{A_{UIM}^{(w/\ sys.\,error)}}\Bigg|_1$$

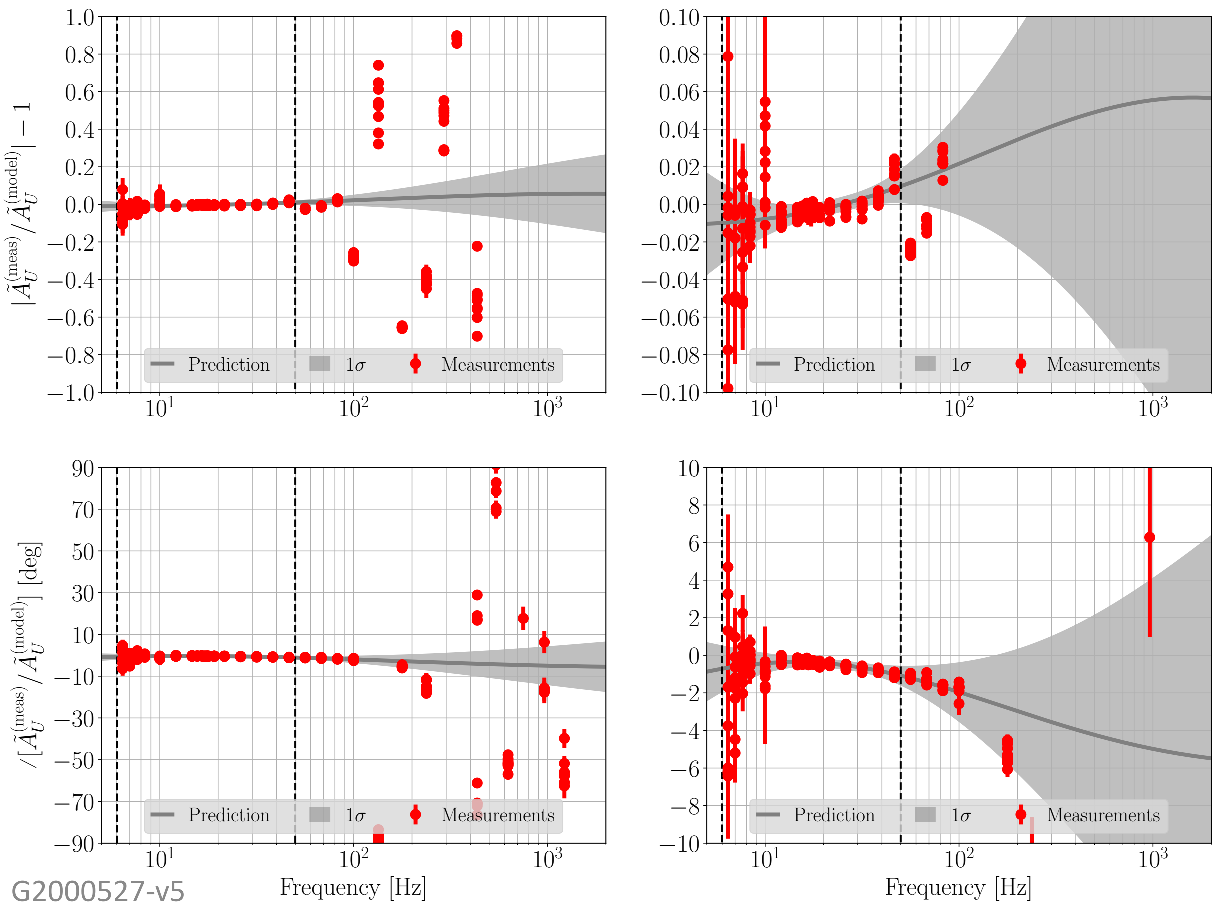
- Huh! So – it looks like the error in State 1 compensation is really of much more concern that the switch between State 1 and State 2 for a short time period.
- That's pretty much it. At least all of this careful study was worth it for some reason.
- On to showing how this manifests in the response function!
- **But also – do remember that this is based on fits of data that doesn't make sense. So hold these truths to be full of salt grains until we get a better measurement.**

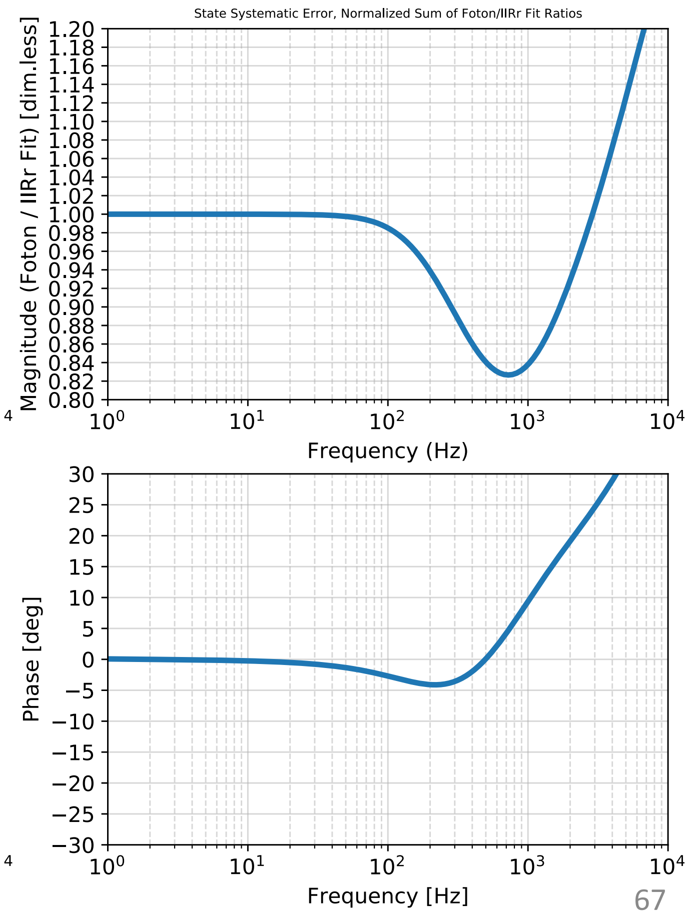


I.7 Sys. Err in A_{UIM} Recap

But, **if we believe the measurement**, is this error big w.r.t. other errors in the UIM?
Yeah – it kinda is!

Namely – the blade spring bending nonsense completely fools the GPR above 50 Hz. So this kind of smoothly varying function just would not be found in / “accounted for with” the GPR. So, we’re stuck having to model it all and estimate the impact on the Response Function systematic error.





Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

1. Why do you care about the UIM?
2. Review where we were before we started
3. Review of the Circuit
4. The Measurement
5. Other models of the circuit
6. The Fit and Each Coil Result
7. Converting fit results in to systematic error in A_{UIM}
8. **Converting sys error in A_{UIM} to sys error in R and Conclusions**

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

I.8 Converting Sys. Err in A_{UIM} to that in R

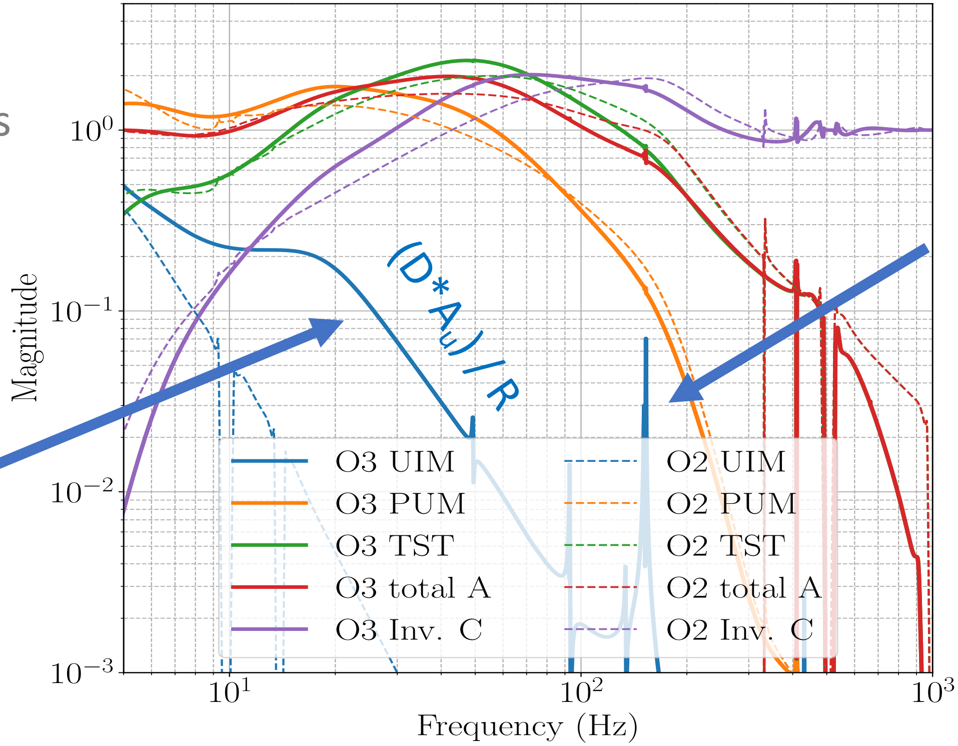
- Hey! We wrote a paper on this! Check out Eq. 11 in P1900245:

$$\tilde{\eta}_{R;A_i} = \frac{1}{R^{(\mathrm{model})}} \left[\frac{1}{C^{(\mathrm{model})}} + \left(\tilde{\eta}_{A_i} \tilde{A}_i^{(\mathrm{model})} + \sum_{j \neq i} \tilde{A}_j^{(\mathrm{model})} \right) \widetilde{D} \right]$$

H1 O3

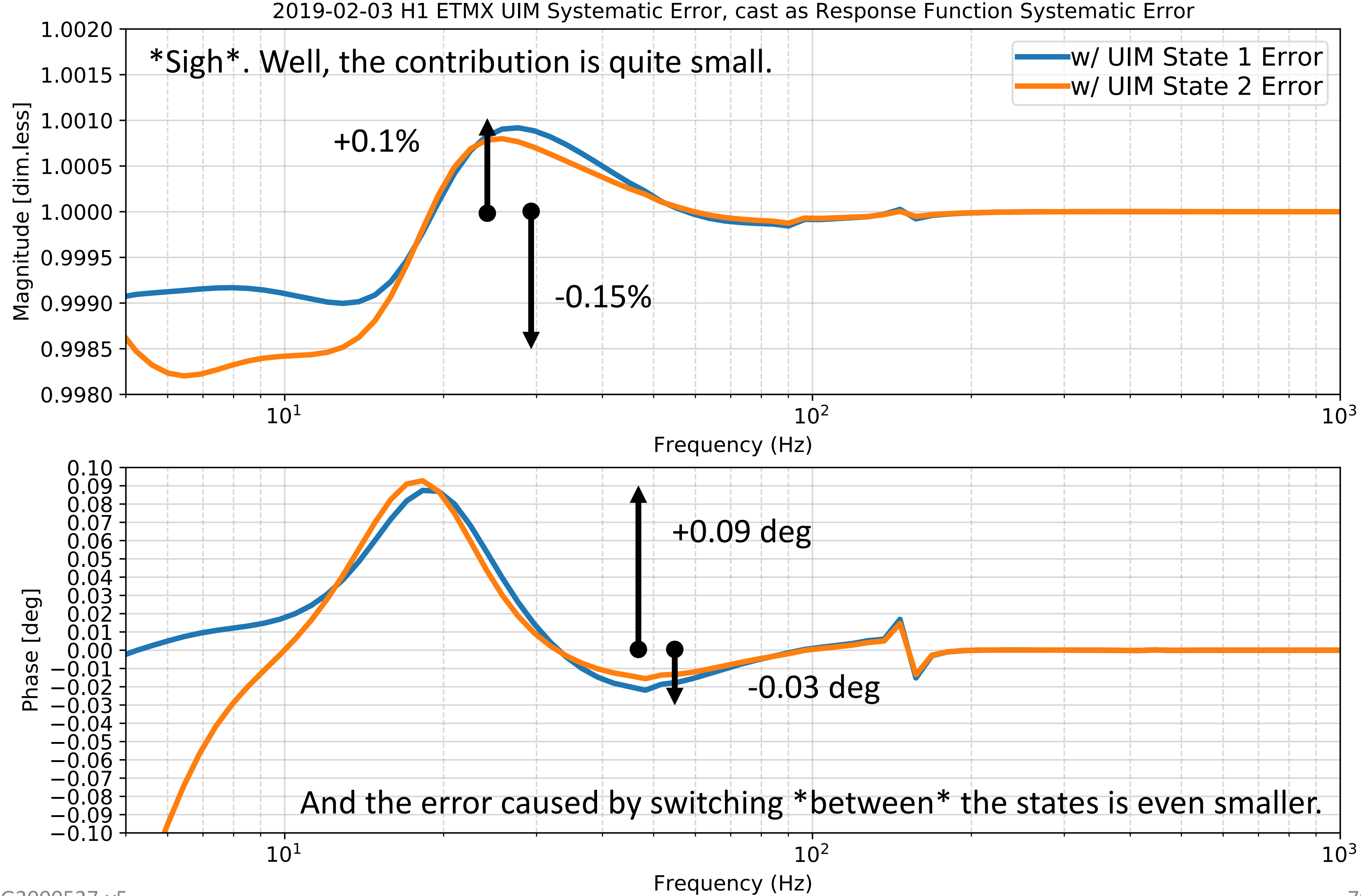
Error contributions to the response function recapped from Slide 6

UIM Contributes at the ~10% level out to ~25 Hz



Vertical Blade Spring Twisting / Bending in L direction causes UIM contribution to spike back in to play at 150 Hz

I.8 Sys. Error R as a result of A_{UIM} Error



I.8 UIM Electronics Error Conclusions

- **The executive summary:** non-Jeff's everywhere whom guessed the answer ahead of time are vindicated in that the UIM electronics error -- either from differences in compensation between states, or poor compensation in general – doesn't substantially contribute to the response function systematic error.

- We may safely proceed with O3B chunk 1 uncertainty budget development without including this systematic error.
  - Note that this would have *not* been "covered" by the GPR even it it were non-negligible.

- BUT: we've now learned many valuable lessons about:
  - How to take the right measurement of a coil driver
  - How to make sense of a fit to data using rough analytic expectations from converting a circuit diagram in to a collective transfer function
  - How bad the compensation is for the UIM driver response
  - How to propagate electronics errors to the response function

PART II:
The OMC Whitening Chassis, from Mar 16 to 27 2020

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What's the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.1 What’s the issue?

[LHO:55620](#)
[IIET:14567](#)

Monday Afternoon, Time of first observation ready segment after change is

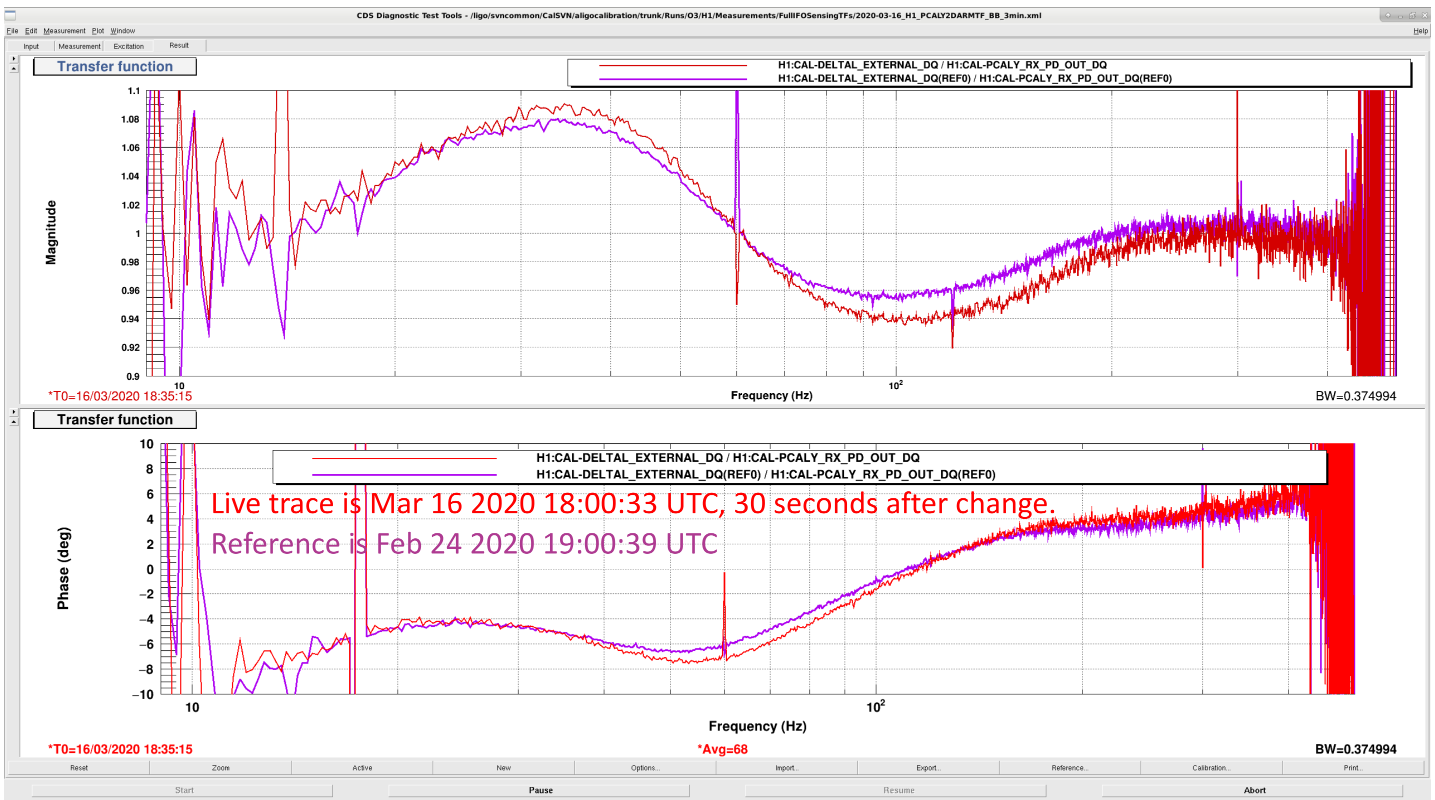
Mar 16 2020 14:18:54 PDT

Mar 16 2020 21:18:54 UTC

GPS 1268428752

Reduced the amount of OMC analog whitening from all 3 stages (2 whitening stages and one low pass) to just 1 stage of whitening. Results in 1-1.5% systematic error change.

Could have modelled sooner and created a new pyDARM parameter file, but ran out of time, so now we must predict the systematic error to be included in the uncertainty budget.



Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

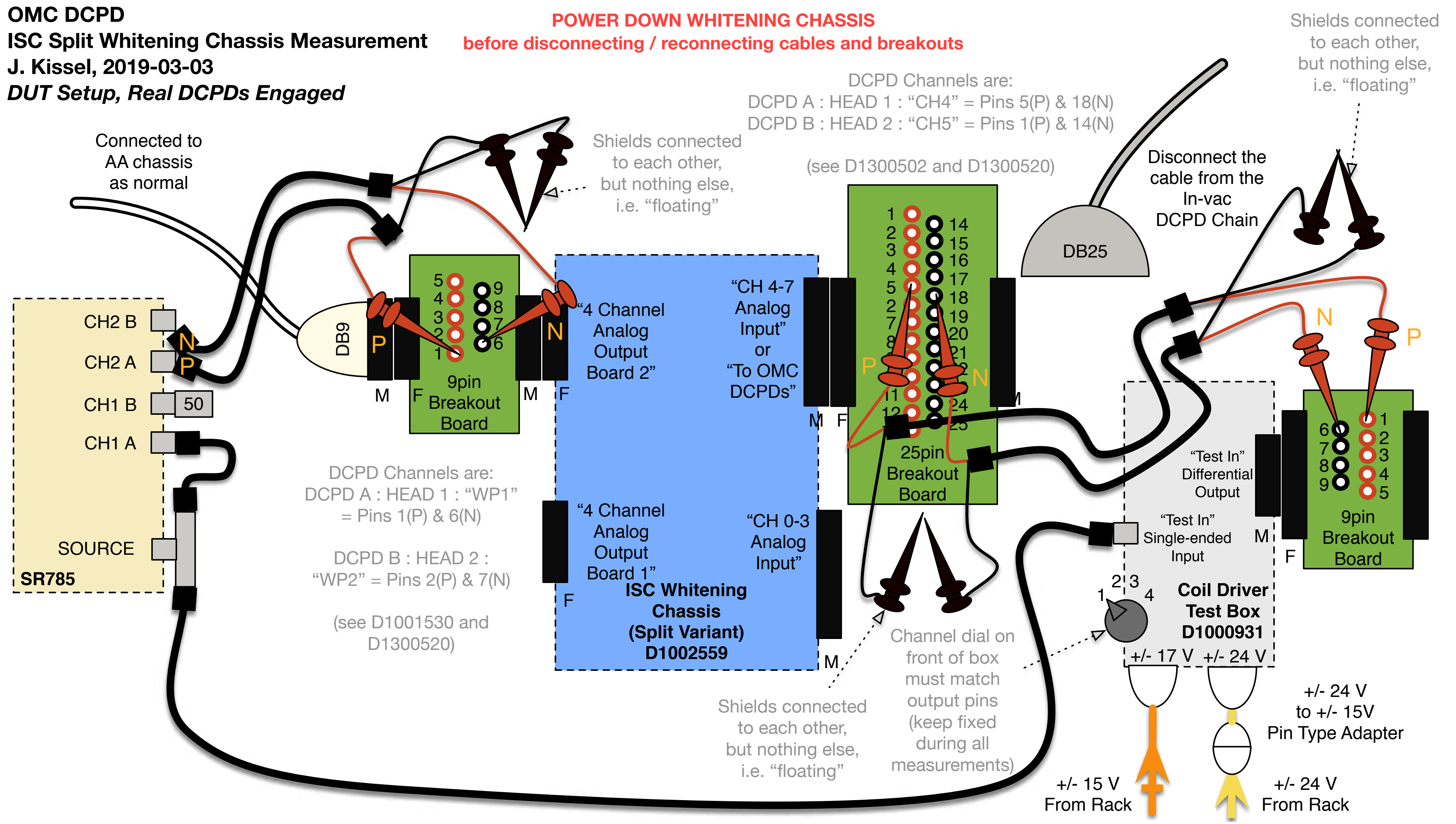
PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.2 Game Plan

- Revisit the Stefan / Lilli aLOG, find the fitting script, make sure we can reproduce the results.
  - Will need fit-from-measured zpks's of each stage of both channels. Do we even have the right data to do this?
  - Find that data, if it's not used in Lilli's fitting script already.
  - If need be replot / innovate Lilli's script.
- Find the right Foton Filter File, get all the z's p's and k's for each filter module.
- Make a prediction of
  - All three stages ON and compensated, vs.
  - Only one stage ON and compensated
- Once DCS frames become available, re-process broadband injections before vs. after change.
  - Maybe Maddie has processed them already? check email.
  - Compare against prediction above.
- Build the ability to include that frequency-dependent systematic error in to RRNom.

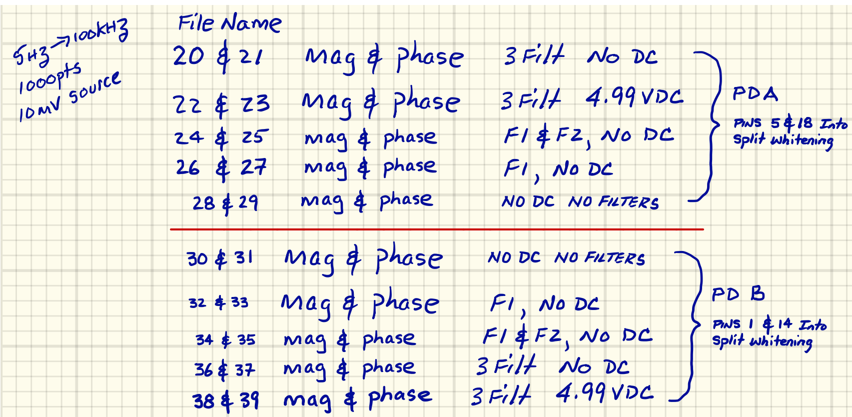
II.2 Review of the Data: Measurement Setup [D1900027]



Open questions (now that we've learned such a valuable lesson in Part I and in [LHO:48358] about paying attention to electrical grounds): Should we have floated the shields? Does it matter that we're "connected to the AA chassis as normal?"

II.2 Review of the Data: Lots of renaming…

1. Rich measured the chassis, zipped up the files, and gave them to Stefan as .78d files: LHO:47418
2. Stefan converted the .78d files to useable .asc data, and wrote a script to build the compensation filters: LHO:47257
3. The script and data were renamed and moved by Jeff to CalSVN: LHO:47290
4. Lilli renamed the files (LHO:47358) that based on Rich's configuration map (from LHO:47418)


5Hz → 100kHz
1000pts
10mV source

File Name
20 & 21 Mag & phase 3 Filt No DC
22 & 23 Mag & phase 3 Filt 4.99VDC
24 & 25 mag & phase F1 & F2, No DC
26 & 27 mag & phase F1, No DC
28 & 29 mag & phase NO DC NO FILTERS
PDA
PWS 5 & 18 Into Split whitening

30 & 31 Mag & phase NO DC NO FILTERS
32 & 33 Mag & phase F1, No DC
34 & 35 mag & phase F1 & F2, No DC
36 & 37 mag & phase 3 Filt No DC
38 & 39 mag & phase 3 Filt 4.99VDC
PD B
PWS 1 & 14 Into Split whitening

Here "F1", "F2" (Rich) or "FM1," "FM2" (Lilli) is for "filter module," [as in the corresponding compensation filter] but really means "whitening stage" [as in the corresponding analog filter stage].
All of these newly renamed files were processed with
fit_OMCDCPDWhiteningChassis_20190304.py
and for the record, used a former measurement of the coil driver test box that was used from the 2019-03-07 measurement of S1900070.

Rich's Measurement configuration	DCPD A (Chan 4) SR785 File Name	DCPD A (Chan 4) Lill's File Name	DCPD B (Chan 5) SR785 File Name	DCPD B (Chan 5) Lilli's File Name
All three stages ON	srs0020.78d	PDA_FM1FM2FM3_mag.asc	srs0036.78d	PDB_FM1FM2FM3_mag.asc
no DC voltage offset	srs0021.78d	PDA_FM1FM2FM3_pha.asc	srs0037.78d	PDB_FM1FM2FM3_pha.asc
First two stages ON	srs0024.78d	PDA_FM1FM2_mag.asc	srs0034.78d	PDB_FM1FM2_mag.asc
no DC voltage offset	srs0025.78d	PDA_FM1FM2_pha.asc	srs0035.78d	PDB_FM1FM2_pha.asc
First stage ON	srs0026.78d	PDA_FM1_mag.asc	srs0032.78d	PDB_FM1_mag.asc
no DC voltage offset	srs0027.78d	PDA_FM1_pha.asc	srs0033.78d	PDB_FM1_pha.asc
All stages OFF	srs0028.78d	PDA_noFM_mag.asc	srs0030.78d	PDB_noFM_mag.asc
no DC voltage offset	srs0029.78d	PDA_noFM_pha.asc	srs0031.78d	PDB_noFM_pha.asc

Measurement was taken *after* all ECRs (including fixing sensitivity to DC offset) were implemented
So Lilli using "no DC offset" data is OK, even though DCPDs during NLN have ~50 mA = ~5V running through the chassis

II.2 Review of the Data: Welp, I need to refit the data

1. I need to understand with what raw data I'm working, so I can understand how to work with it.
2. Stefan's metric for success in LHO:47257 was "A / B", but the DCPDs are *added*, so we need something like "A + B"
3. Although the fit looks pretty fantastic in LHO:47358, Lilli just fit each measurement straight up, and didn't compute any ratios of transfer functions to isolate poles and zeros from each stage, so hard to verify which poles are coming from which stage (and if say, FM1's fit changes when it's a part of FM1, FM2, and FM3).
4. When computing the residual contribution to R in LHO:47453, Lilli plotted (A+B)/2, instead of (b_A*A+b_B*B), where b_A and b_B are the balance matrix values,
   - (no need for the "divide by two" part of the "average" since it's an over all gain, and captured in H_C)
5. I think there's a bug in fit_OMCDCPDWhiteningChassis_20190304-compareTF.py, fit_OMCDCPDWhiteningChassis_20190304-compareRes.py, and fit_OMCDCPDWhiteningChassis_20190304-compareRes-writeTXT.py, which all mis-represent the poles and zeros installed in foton; we should just import the design from the real foton file, H1OMC_1239468752.txt
6. IIRrational has improved, and should now need a lot less input, so we can make the code cleaner.

ALSO, Again, Open Question: As discussed in LHO:48358 (learned, unfortunately, a month later), the DUT and coil driver test box *must* be properly referenced to ground, or one may get distortions in the frequency response at high frequency. The data for S1900070 clearly shows a gain of 2.002, confirming that at least the test box measurement was done with an errant "floating" ground. Nothing we can do about it now, but the OMC whitening chassis might behave like the AA chassis, where the high-frequency response gets distorted without measurement shields properly referenced to ground.

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.3 Review of the Circuit: Squad goals

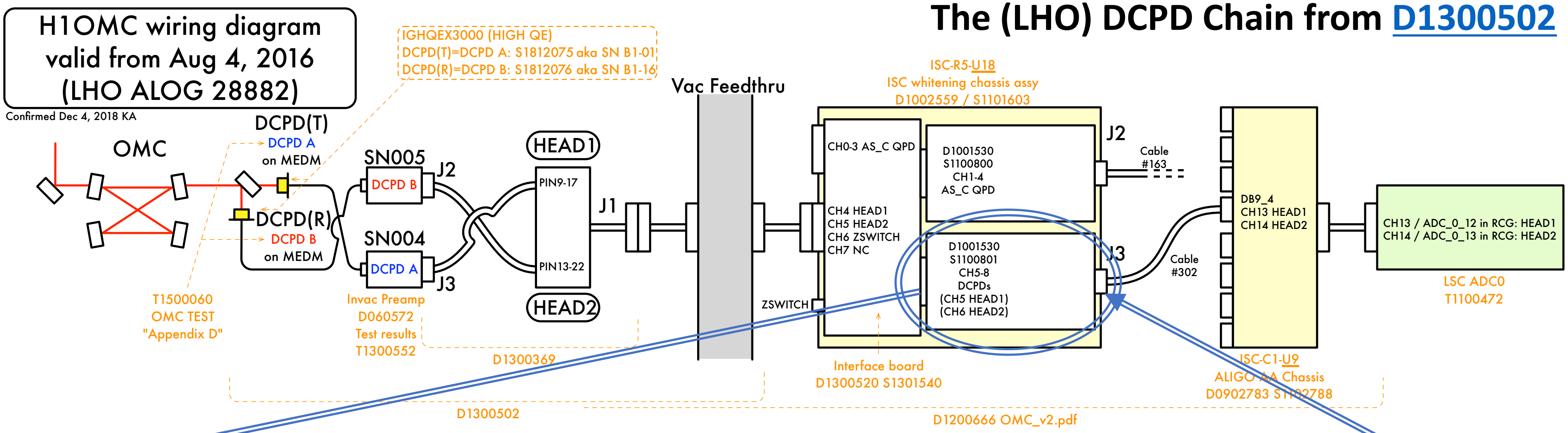
- I'm gunna do the same thing I did for the UIM driver and break down the circuit into manageable pieces (I'll go a little faster, assuming you're used to the methods now having gone through Part I).

- Again, this is (a) to teach you, the next generation, and (b) **so can be sure we know what to expect when fitting, as well as verifying that we're modelling it correctly.**

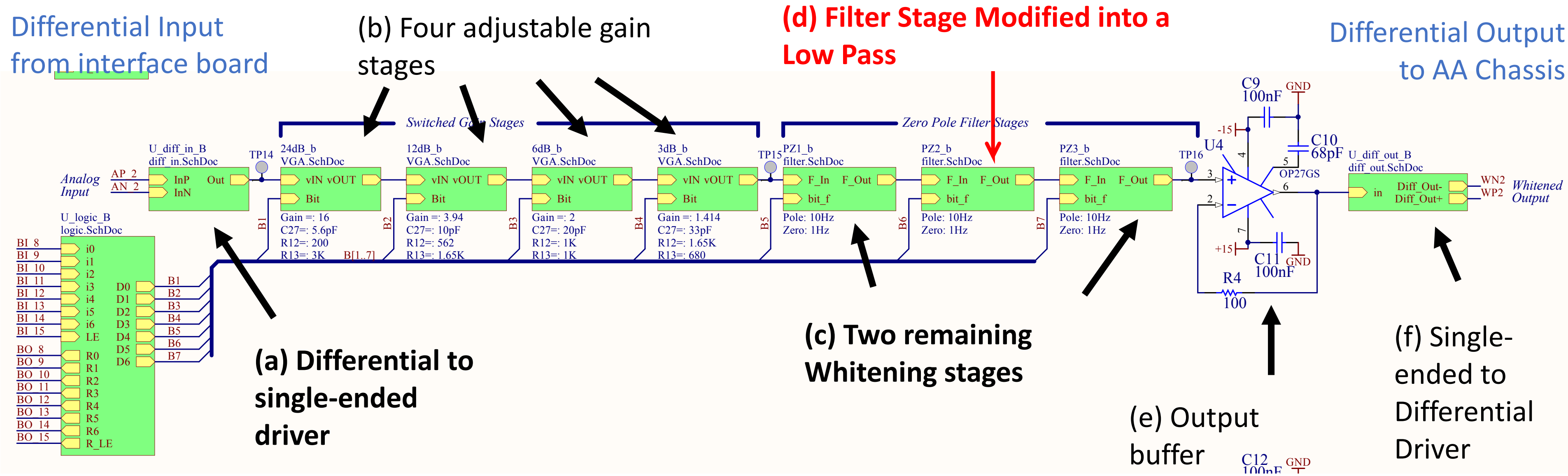
- As before, we'll be focusing on finding frequency response, since the over all gain of the OMC DCPD chain is measured as a whole. You'll find that the super-Nyquist high-frequency poles are especially interesting…

II.3 Review of the Circuit: Forest through the Trees

The (LHO) DCPD Chain from D1300502

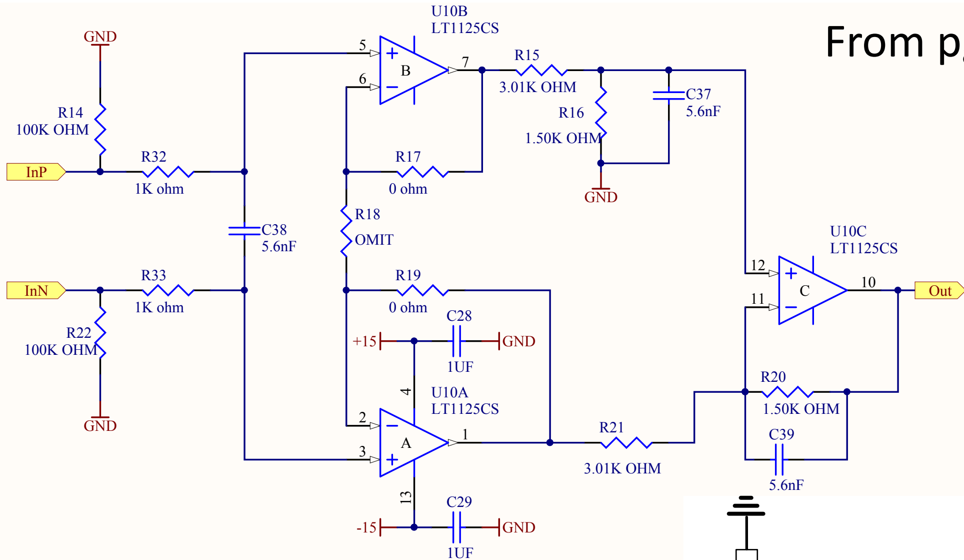


The complex chain internal to the whitening chassis alone from pg 1 D1001530-v7



***Bold things** we'll find to have interesting frequency response*

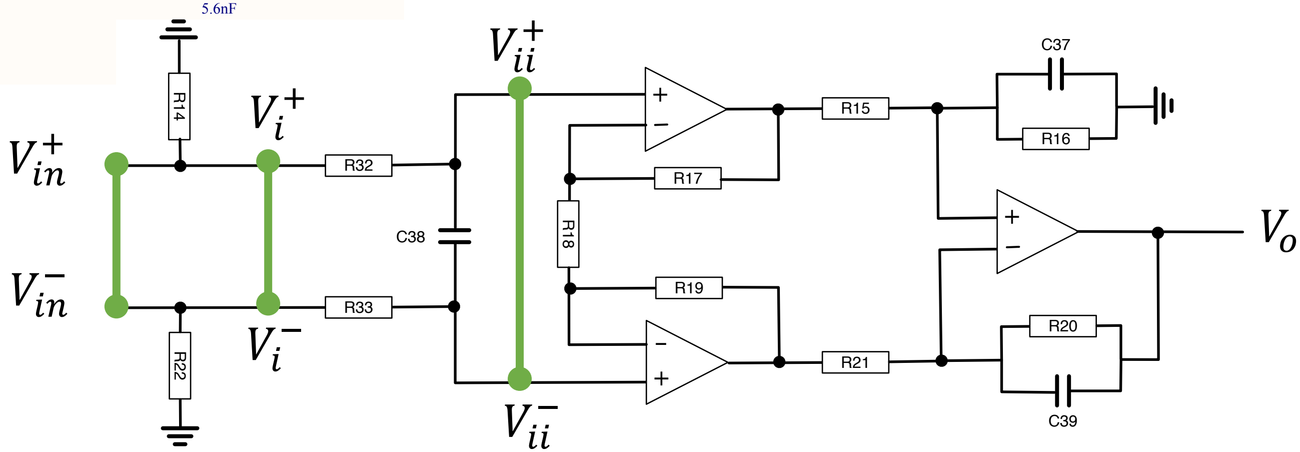
II.3 Review of the Circuit: (a) Diff.-to-S.E. Driver



From pg 2 of D1001530-v7

To analyze, let's break in to three parts

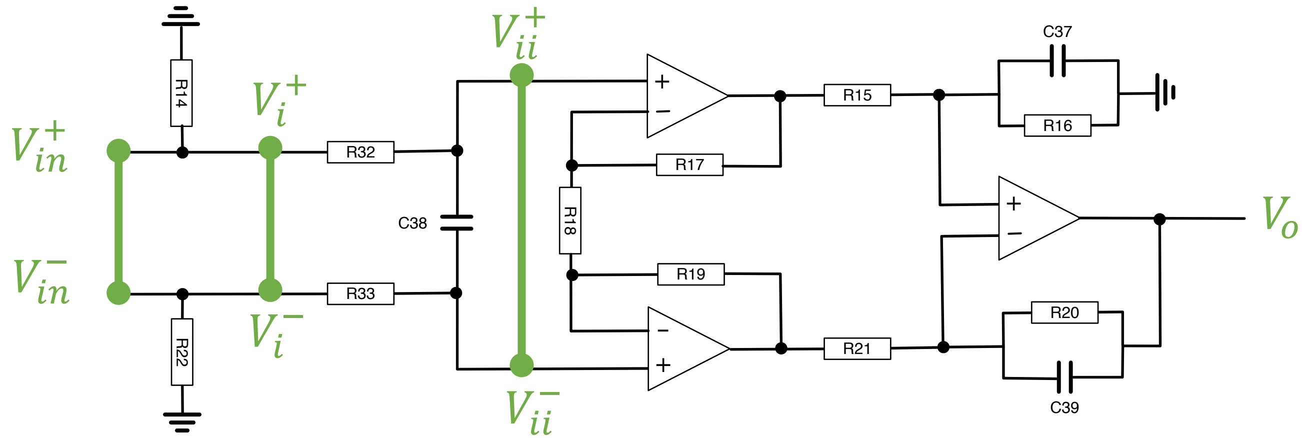
i. The voltage divider
ii. some passive input low pass filter
iii. the *actual* differential-to-single-ended driver



For (a), it's our typical LIGO input voltage divider, aka load termination, for the long cable run from in vacuum. That means we have to understand to what this board is connected (guh). So, we go upstream to the interface board, D1300520-v3 (see no impedance), then further to the head D1300369-v1 (see no impedance), then finally to the in-vac preamp D060572-v1, to see an impedance of 1k (plus whatever small resistance from the cable run). *This* time, unlike the UIM, I think we can ignore parasitic cable capacitance, so this is just a gain "drop" of

$$G_{in} = V_i^+/V_{in}^+ = V_i^-/V_{in}^- = R_{14}/(R_L + R_{14}) \approx 0.99$$

II.3 Review of the Circuit: (a) Diff.-to-S.E. Driver



For (ii), the passive low-pass filter, we can do the usual trick of splitting the different circuit in half, and dividing the impedance of components that span the legs in two. In this case just a capacitor, so

$$\left.\frac{(V_{ii}^+ - V_{ii}^-)}{(V_i^+ - V_i^-)}\right|_{C_{38}} = \left.\frac{V_{ii}^+}{V_i^+}\right|_{2C_{38}} = \frac{1/i\omega 2C_{38}}{(R_{32} + 1/i\omega 2C_{38})} = \frac{1}{(1 + i\omega(2R_{32}C_{38}))}$$

$$f_p^{LP} = 1/\big(2\pi(2R_{32}C_{38})\big) = 28.42\text{ kHz}$$

For (iii), the actual differential driver, you can see that I've re-drawn it to better match the traditional analysis for of an instrumentation amplifier, make it easier to agree that the transfer function takes the form

$$G_{ia} = \frac{V_o}{(V_{ii}^+ - V_{ii}^-)} = \left(1 + \frac{2R_{17}}{R_{18}}\right)\frac{Z_{RC}}{R_{15}} \qquad \left.G_{ia}\right|_{DC} = \frac{R_{16}}{R_{15}} = 0.498$$

because, as expected, the circuit is symmetric (e.g. R17=R19, R15=R21, etc.). Since R17 = 0 Ω, and R18 has been omitted, i.e. R18 = ∞ Ω, the first term reduces to 1.0. With

$$Z_{RC} = R_{16}/(1 + i\omega R_{16}C_{37})$$

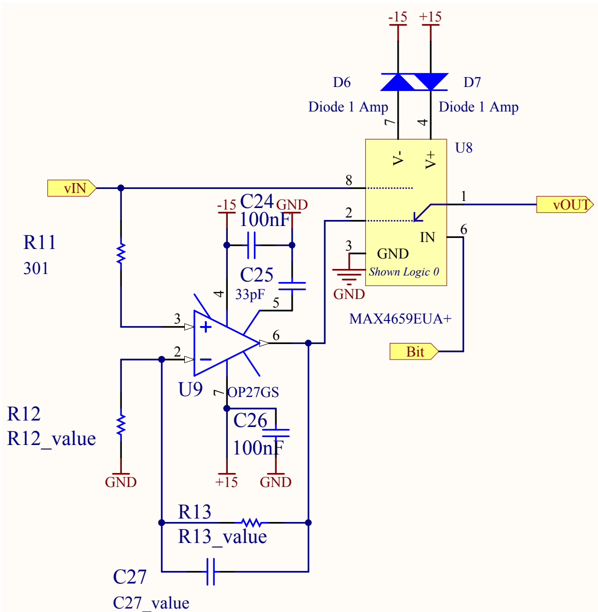
then

$$G_{ia} = \frac{R_{16}}{R_{15}}\frac{1}{(1 + i\omega R_{16}C_{37})}$$

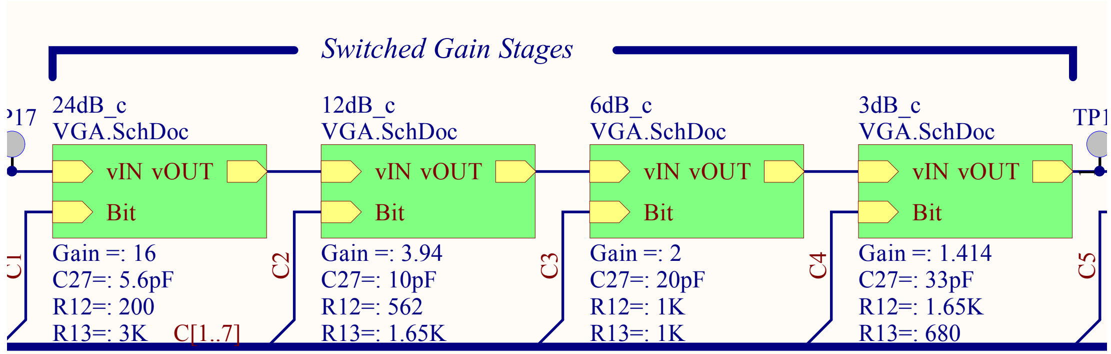
$$\boldsymbol{f_p^{ia} = 1/(2\pi R_{16}C_{37}) = 18.95\text{ kHz}}$$

II.3 Review of the Circuit: (b) Adjustable Gain Stages

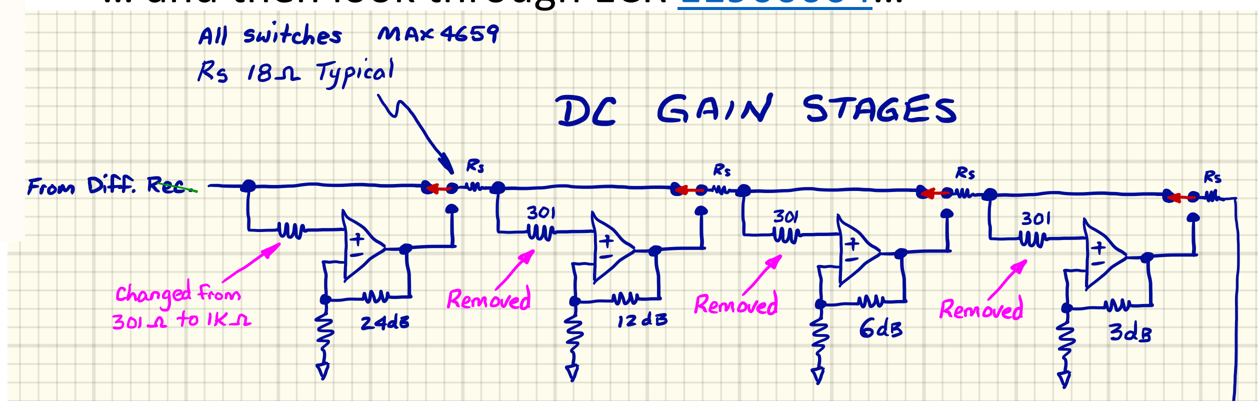
From pg 3 of D1001530-v7…



… and back to pg 1 of D1001530-v7 to get values…



… and then look through ECR E1900064…



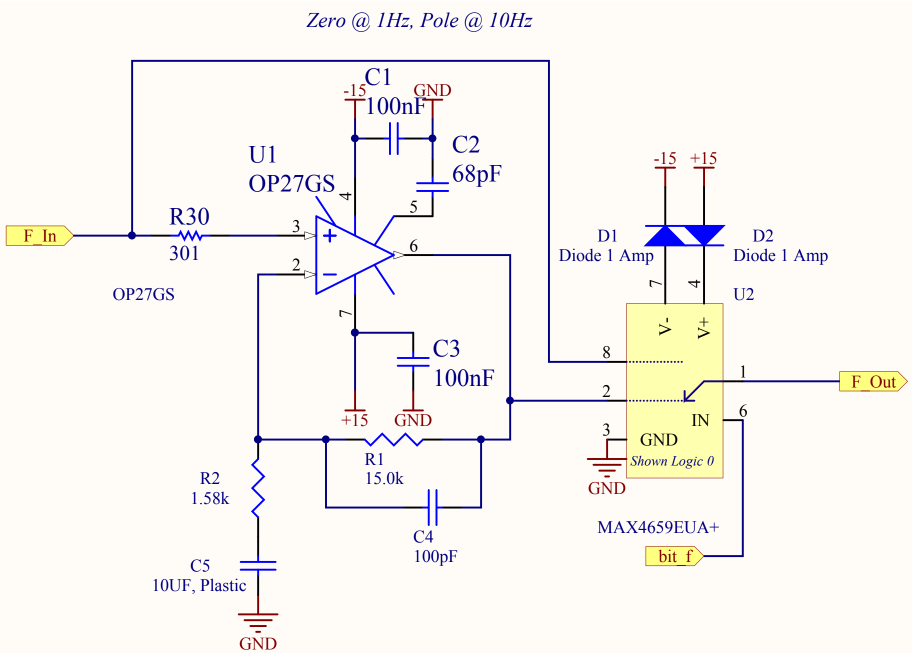
To remember that ***all* of the switchable gain stages have been disabled / bypassed in O3**:

- the removal of each R11 in the last three stages makes an infinite resistance to ground, so there's no longer any input, and further (though no where so explicitly stated) the switches are bypassed with a jumper so the output of the opamp is no longer connected.
- R11 in the first stage remains, but is increased to 1k. Because it's connected to the ZRC network of the "Diff. Rec." aka "Differential to Single Ended Driver" aka "Instrument Amplifier," and the real part of ZRC is dominated by R20 = 1.5 kΩ, which means the only remaining, fixed gain is

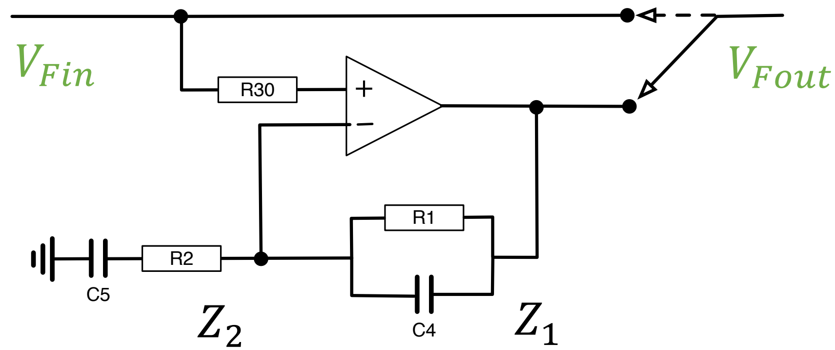
$$G = R_{11}/(R_{20} + R_{11}) = 1500/(1000 + 1500) = 0.4$$

II.3 Review of the Circuit: (c) Whitening Stages

From pg 4 of D1001530-v7…



These can be redrawn as a non-inverting amplifier…



$$Z_1 = \frac{R_1(1 + 1/i\omega C_4)}{(R_1 + 1/i\omega C_4)} = \frac{R_1}{(1 + i\omega R_1 C_4)}$$

$$Z_2 = R_2 + 1/i\omega C_5 = \frac{(1 + i\omega R_2 C_5)}{i\omega C_5}$$

$$\frac{V_{Fout}}{V_{Fin}} = 1 + \frac{Z_1}{Z_2} = 1 + \frac{\dfrac{R_1}{(1 + i\omega R_1 C_4)}}{\dfrac{(1 + i\omega R_2 C_5)}{i\omega C_5}} = \frac{1 + i\omega(R_1 C_5 + R_1 C_4 + R_2 C_5) - \omega^2 (R_1 R_2)(C_4 C_5)}{(1 + i\omega R_2 C_5)(1 + i\omega R_1 C_4)}$$

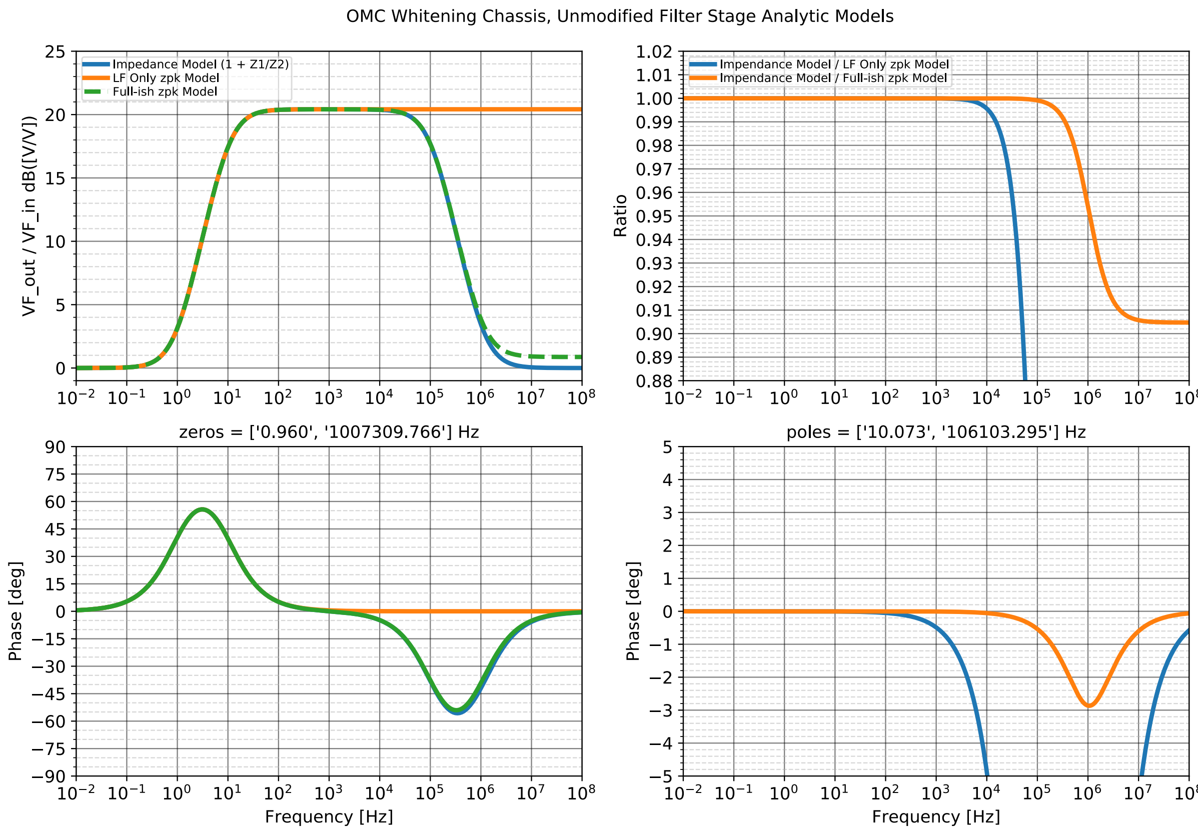
$$\boldsymbol{f_z^F = 1/\big(2\pi(R_1 C_5 + R_1 C_4 + R_2 C_5)\big) = 0.960\ \mathrm{Hz}}$$

$$f_z^F = 1/[(2\pi)^2 (R_1 R_2 C_4 C_5)] = 1.069\ \mathrm{MHz}$$

$$\boldsymbol{f_p^F = 1/\big(2\pi(2 R_2 C_5)\big) = 10.073\ \mathrm{Hz}}$$

$$\boldsymbol{f_p^F = 1/\big(2\pi(2 R_1 C_4)\big) = 106.103\ \mathrm{kHz}}$$

II.3 Review of the Circuit: (c) Whitening Stages



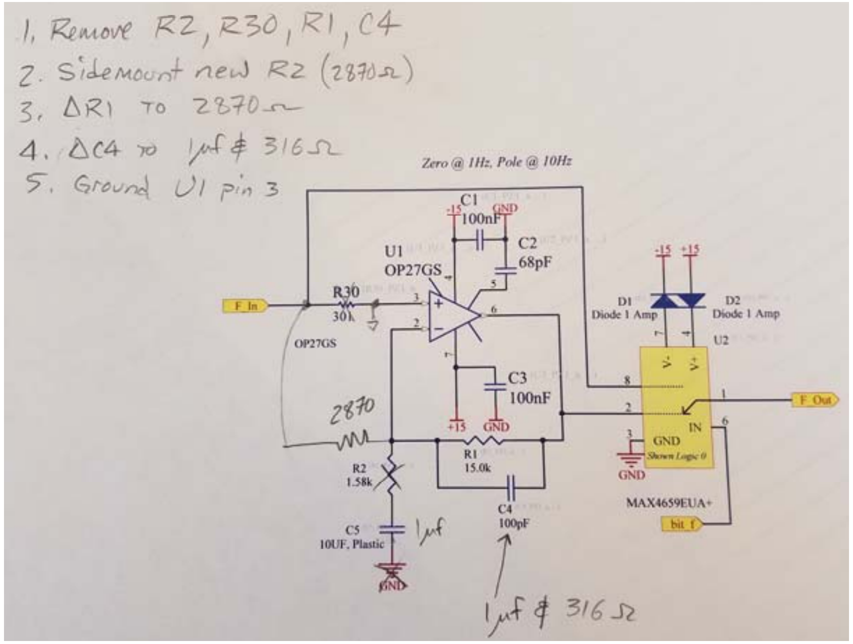
The z:p pair we really want at 0.96:10.07 Hz i.e. the “1:10” whitening as expected.
But... there is the z:p pair at 1.06:0.103 MHz that causes 0.5 deg phase loss at 1 kHz if we exclude it.

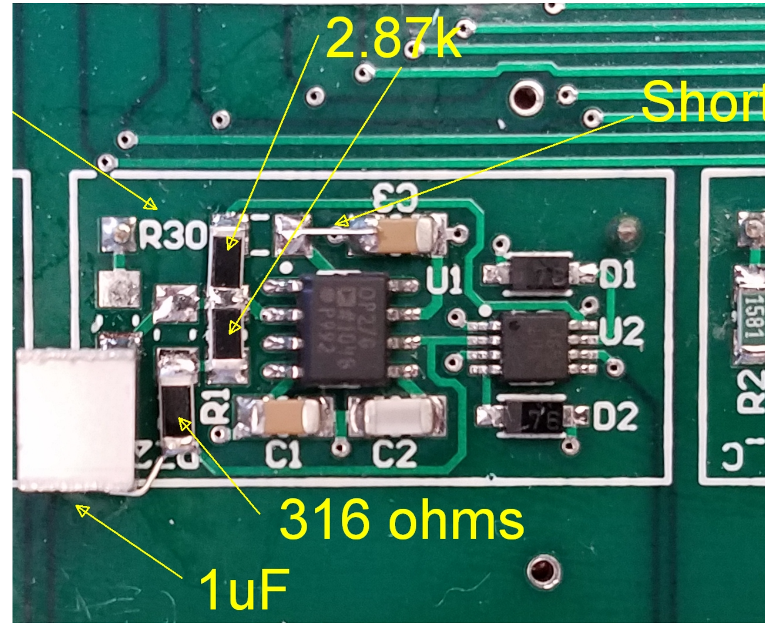
Two-take-aways:

1. **This 1.06 : 0.103 MHz response is a part of each filter stage**, so it comes and goes, depending on whether you’re using the stage (you need a “high-frequency pole” for each stage you use)
2. **Very important lesson for measurement taking and fitting:** know where the zeros and poles are ahead of time, so you don’t end up fitting for them with data that doesn’t sufficiently cover their frequency region (fitting for zeros or poles outside of data frequency region = BAD, prone to errors)

II.3 Review of the Circuit: (d) The New Low Pass

Another documentation doozy… images from [E1600252]

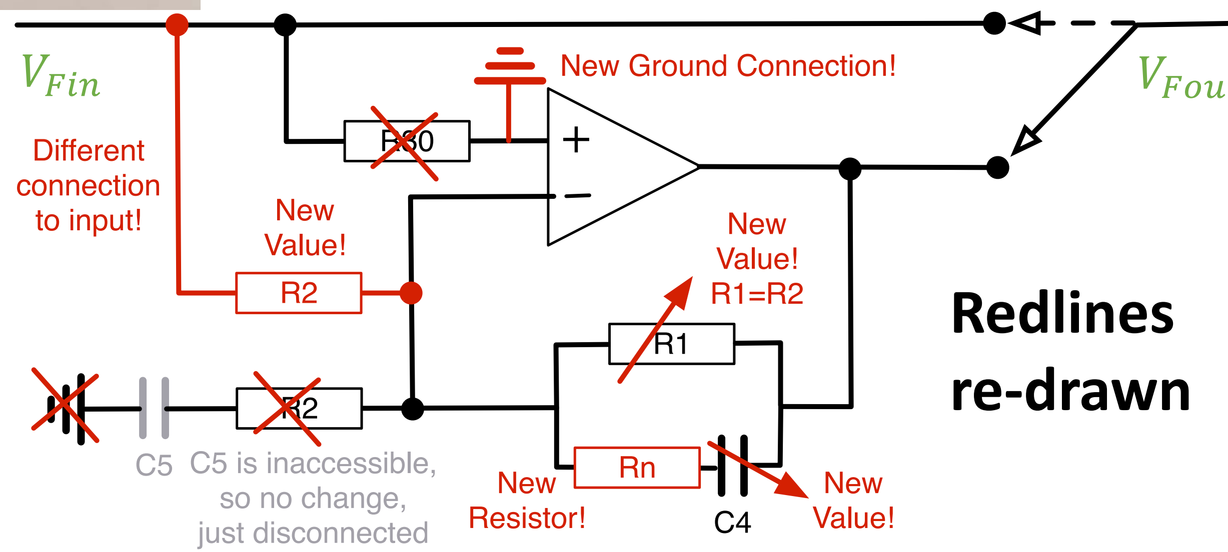




Seriously: While I chide Rich Abbott for his documentation, this change is indicative of his true engineering brilliance, and artistic mastery of circuits.

He's a busy man, just getting the job *done*.

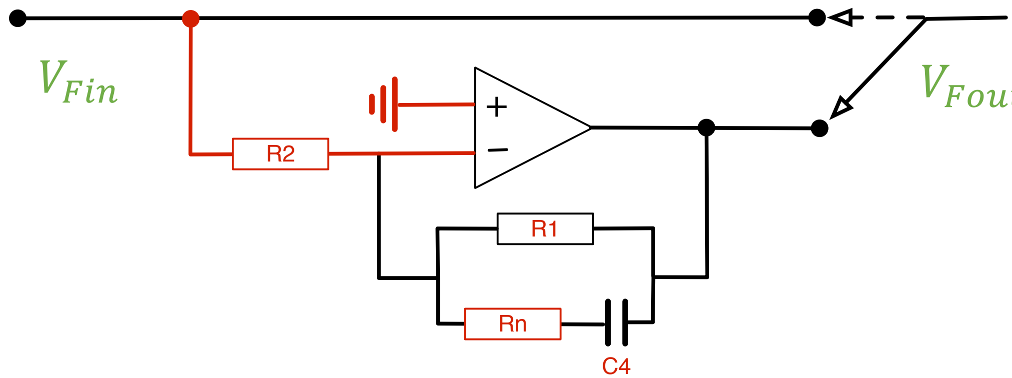
It's *literally* only us who care at this level.



Redlines re-drawn

II.3 Review of the Circuit: (d) The New Low Pass

Redlines re-drawn again, shown with functionally equivalent design intent as an inverting op-amp.



Now “easy” to analyze:

$$Z_1 = \frac{R_1(R_n + 1/i\omega C_4)}{((R_1 + R_n) + 1/i\omega C_4)} = R_1 \frac{(1 + i\omega R_n C_4)}{(1 + i\omega(R_1 + R_n)C_4)}$$

$$\frac{V_{Fout}}{V_{Fin}} = -\frac{Z_1}{Z_2} = -\frac{R_1}{R_2}\frac{(1 + i\omega R_n C_4)}{(1 + i\omega(R_1 + R_n)C_4)}$$

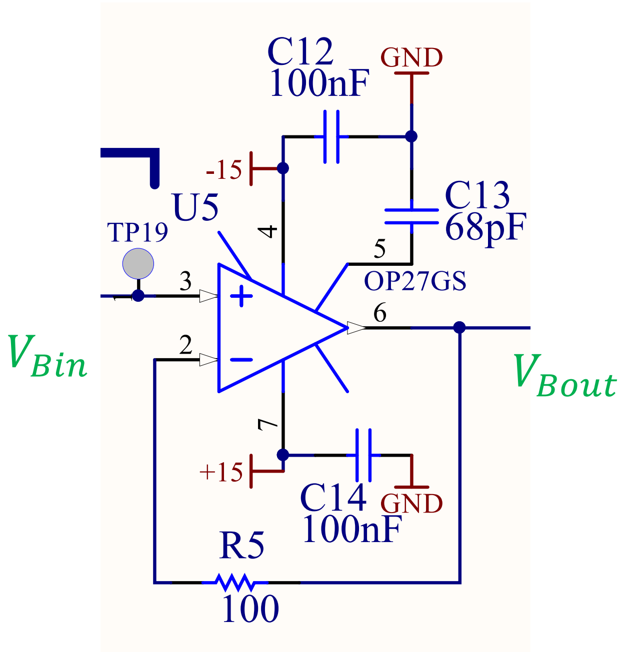
$$G_{ia}\Big|_{DC} = -\frac{R_1}{R_2} = -1.0$$

$$\boldsymbol{f_z^F = 1/(2\pi R_n C_4) = 503.655\ \mathrm{Hz}}$$

$$\boldsymbol{f_p^F = 1/(2\pi(R_1 + R_n)C_5) = 49.954\ \mathrm{Hz}}$$

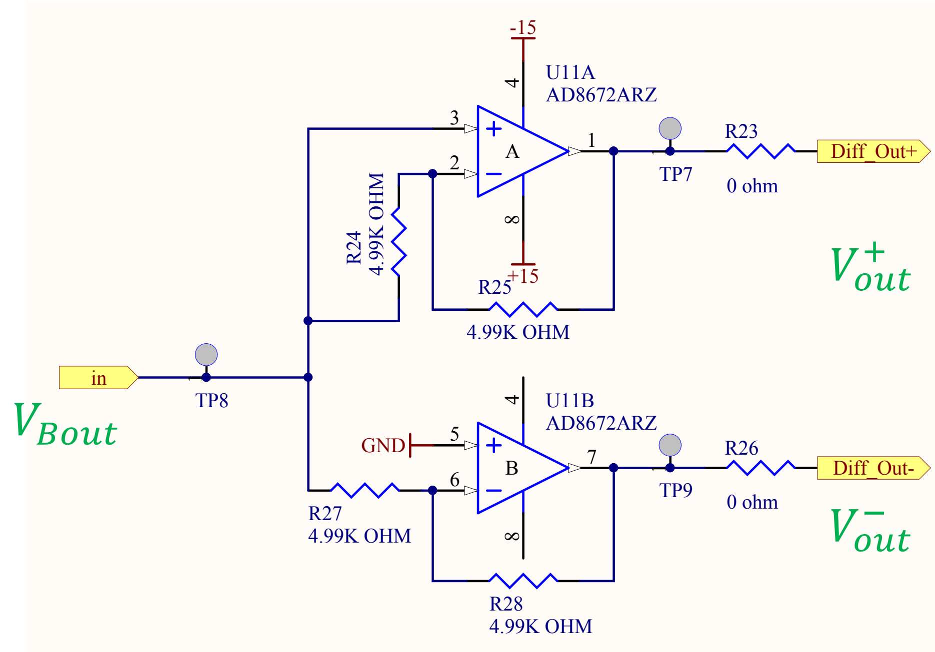
OK, cool. No sneaky high frequency poles here…

II.3 Review of the Circuit: (e) and (f) buff and SE-D amp



(e) is a standard gain buffer,

$$G_B = \frac{V_{Bout}}{V_{Bin}} = 1 + \frac{R_5}{\infty} = 1.0$$



(f) is response-free single-ended to differential amplifier

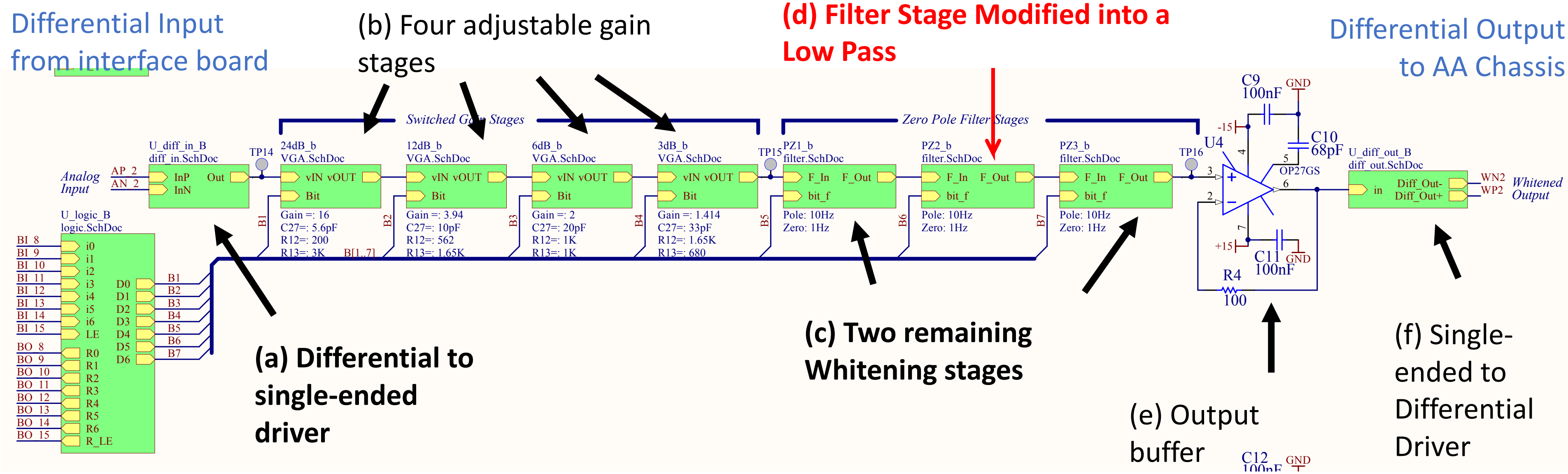
$$G_{oa} = \frac{(V_{out}^{+} - V_{out}^{-})}{V_{Bout}}$$

$$= \frac{V_{out}^{+}}{V_{Bout}} - \frac{V_{out}^{-}}{V_{Bout}}$$

$$= \frac{R_{25}}{R_{24}} - \left(-\frac{R_{28}}{R_{27}}\right)$$

$$\boldsymbol{G_{oa} = 2.0}$$

II.3 Review of the Circuit: Conclusions



(a) $G_{in} \approx 0.99$

$$G_{ia}\Big|_{DC} = 0.498$$

$$f_p^{ia} = \mathbf{18.95\ kHz}$$

(b) $G = 0.4$

(e) $G_B = 1.0$

(f) $G_{oa} = 2.0$

(c) Two whitening stages available,
Per stage you get:

$$f_z^F : f_p^F = \mathbf{0.960 : 10.073\ Hz}$$

$$f_z^F : f_p^F \approx 1.069 : 0.106\ \mathrm{MHz}$$

(d) One low-pass available:

$$f_z^F : f_p^F = \mathbf{503.655 : 49.954\ Hz}$$

The overall gain (differential in, differential out) is therefore

$$\boldsymbol{G_{overall} = G_{in}\, G_{ia}\Big|_{DC}\, G\ \ G_B\, G_{oa} \approx 0.4}$$

Outline

Two Parts, each quite long. *sigh*

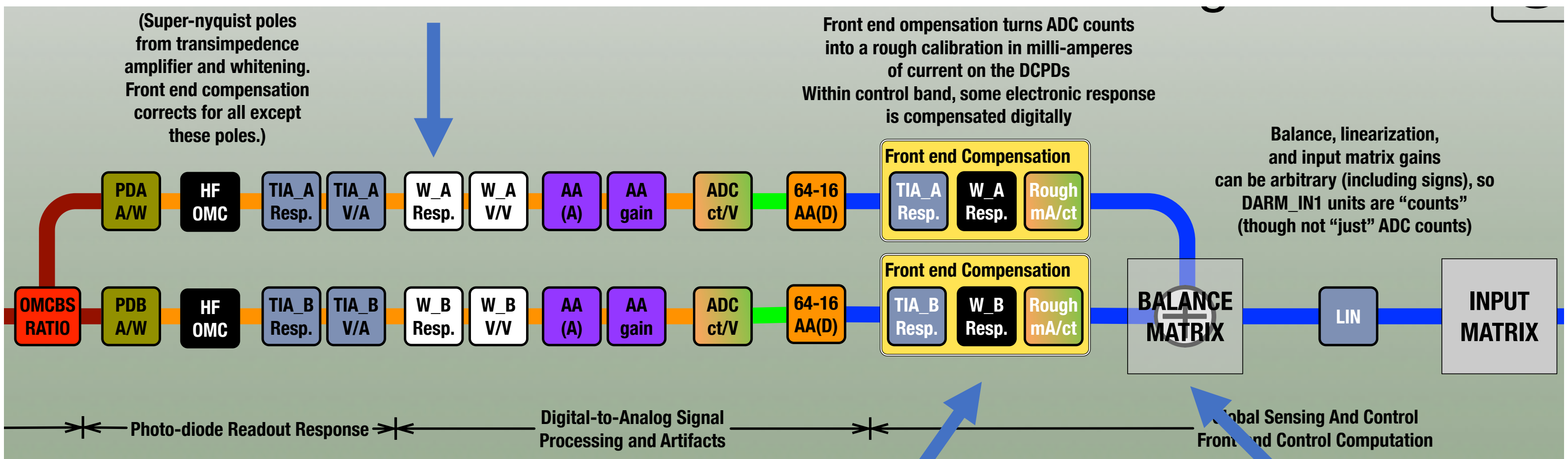
PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What's the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.4 Compensation Scheme: Context

Using a zoom in on the relevant corner of the Subway Map G1501518, each channel's zeros and poles of OMC whitening chassis response is represented as the white "W_A Resp." and "W_B Resp." boxes



For the low frequency zeros and poles, i.e.

$$f_z^F : f_p^F = 0.960 : 10.073 \text{ Hz}$$

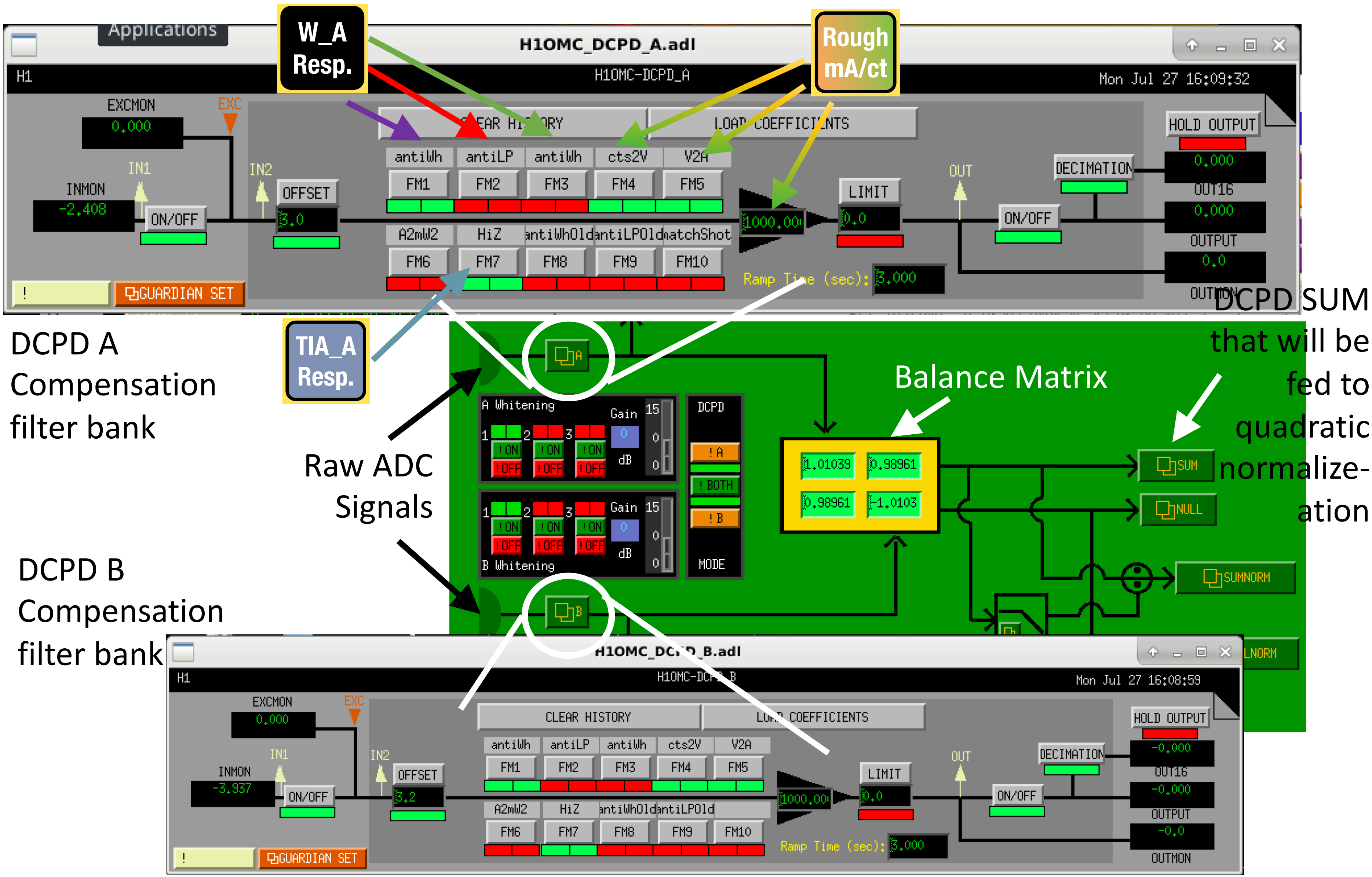
$$f_z^F : f_p^F = 503.655 : 49.954 \text{ Hz}$$

$$f_z^F : f_p^F = 0.960 : 10.073 \text{ Hz}$$

we "compensate" (multiply) the raw ADC signal with (by) digital filters "in loop," with digital IIR filters in the front-end shown as black boxes with the same name

The Balance Matrix digitally adds the two paths together, with coefficients that are *close* to 1.0 (not 0.5!) These take care of the *differential* gain between the channels

II.4 Compensation Scheme: Context



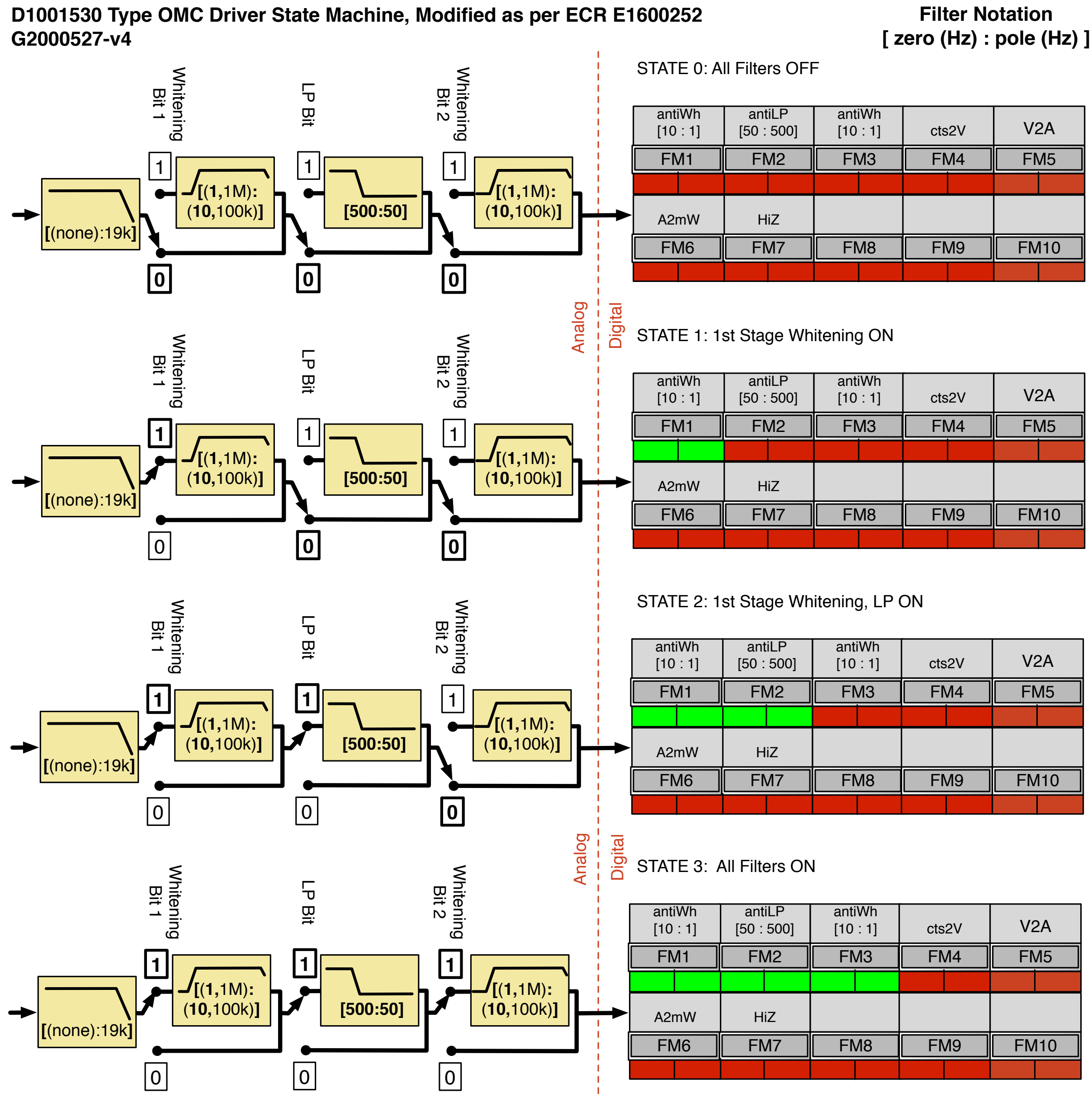
II. 4 Compensation Scheme: The switchable response

Here's a new state machine diagram to show what's is done to compensate the switchable filter response.

Notice, that the fixed 19 kHz pole and the 1 MHz:100 kHz zero:pole portion of the switchable response is not compensated.

These are the filters that Stefan updated quickly, and documented in LHO:47257.

(There are other filters in here to compensate for the switchable trans-impedance and other gain-only compensation, but not relevant to this discussion.)



II.4 Compensation Scheme

For all the high-frequency response values,

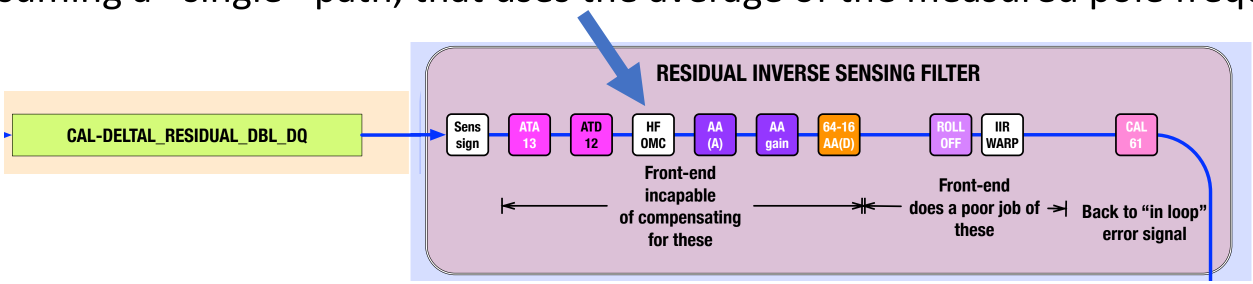
$$f_p^F = 0.106 \text{ MHz}$$
$$f_p^{ia} = 18.95 \textbf{ kHz}$$
$$f_p^F = 0.106 \text{ MHz}$$

(Notice the 1 MHz zero is dropped from the discussion, since it is assumed to have negligible impact in the response below 1 kHz)

the front-end code isn't running fast enough

- clock cycle is 61 usec,
- i.e. a rate of 16.384 kHz,
- thus a Nyquist frequency of 8.192 kHz,
- but with IIR filter warping, practically that means zeros and poles can't be accurately represented above ~5 kHz

Therefore we must rely on the acausal GDS (or DCS) pipelines to compensate these poles.
We do so assuming a *single* path, that uses the average of the measured pole frequencies from each channel.



pyDARM creates this filter, informed by the model parameter files …

Now you know what we'll be comparing our *new* fit against, and you've got a good handle on how things fit in to the bigger compensation picture. So let's get to fitting, and estimating the systematic error.

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.5 Fitting and Predicting Systematic Error

- Order of operations for fitting and predicting the error:

A. For each of the two DCPD Channels:

  i. Import measurement (which are of cascaded application of filtering), divide by test box
  ii. Take ratios of measurements to obtain individual stage response
  iii. Fit individual stages, obtain expected zeros and poles
  iv. Stack fit zeros:poles to re-create cascading application of filtering, compare against measurement to create residuals
  v. Reconstruct residuals of compensation we actually used (both in front-end for LF and GDS for HF)

B. Then, for each filter state

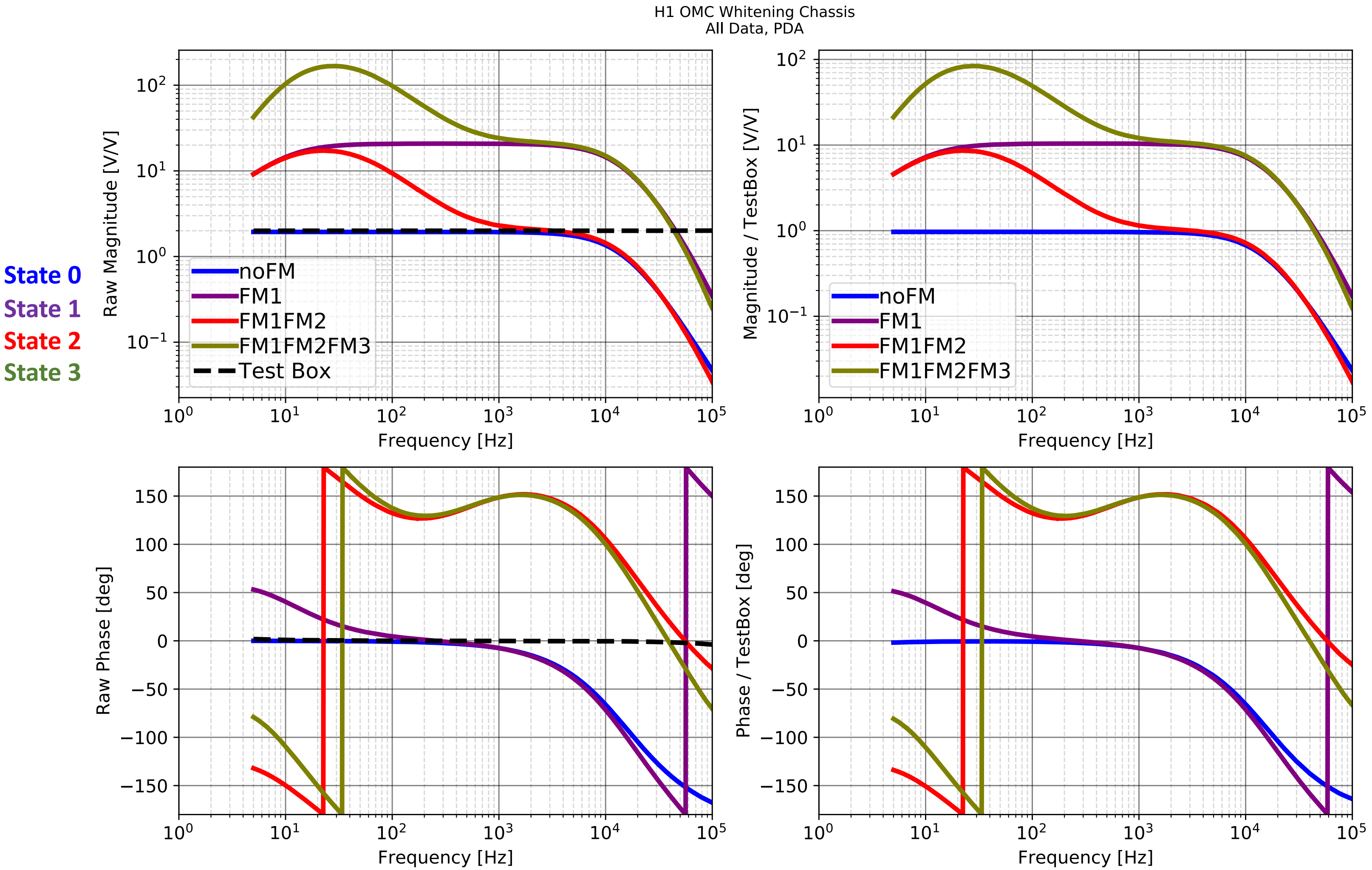
  i. Sum each paths residuals (use “balance matrix” for compensation version, and apply HF poles)
  ii. Create a residual (both with existing compensation, and with fit)
  iii. Take the ratio of State 1 (one Whitening Stage – i.e. configuration after Mar 10), and State 3 (both whitening stages and the low pass – i.e. what we had for all prior times in O3)

C. Finally, estimate response function error

  i. Use ratio of State 1 to State 3 as “η_{OMCWC}” for C
  ii. Compute reference R from 20200103 model, then a modified R using C* η_{OMCWC}
  iii. **FINAL ANSWER ESTIMATE** Take ratio of R_η_{OMCWC} / R_reference

II.5 Fitting and Predicting Systematic Error

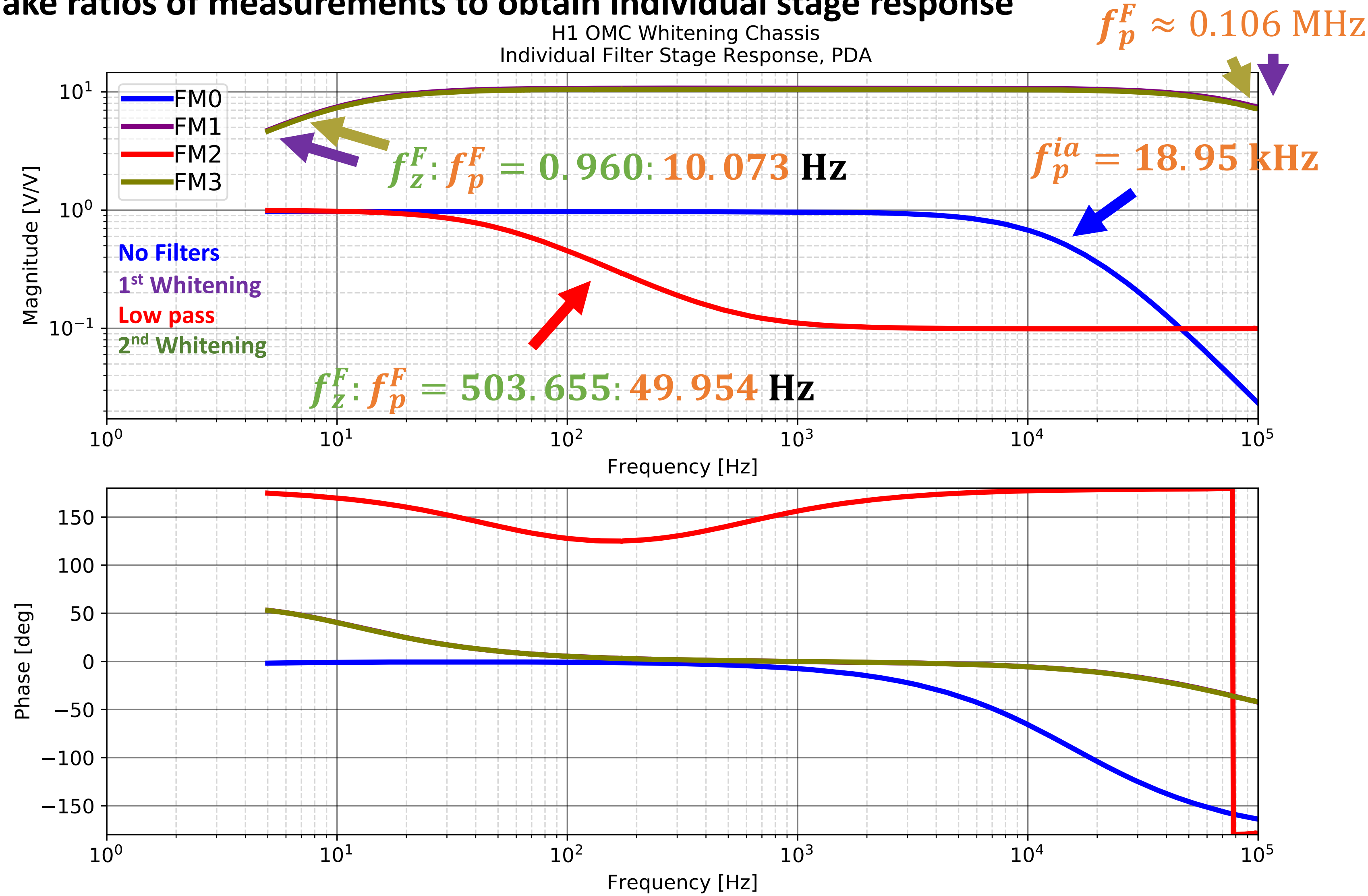
A.i. Import measurement (which are of cascaded application of filtering), divide by test box



This (the right panels) is the data that Lilli fit in [LHO:47358](). In fact, we really only took the results from the green curve (i.e. State 3). **One can see how a fitting program might get confused**, having to fit for the poles and zeros of all *four* states simultaneously… and **note the frequency vector only spans 5 Hz to 100 kHz**.

II.5 Fitting and Predicting Systematic Error

A.ii. Take ratios of measurements to obtain individual stage response

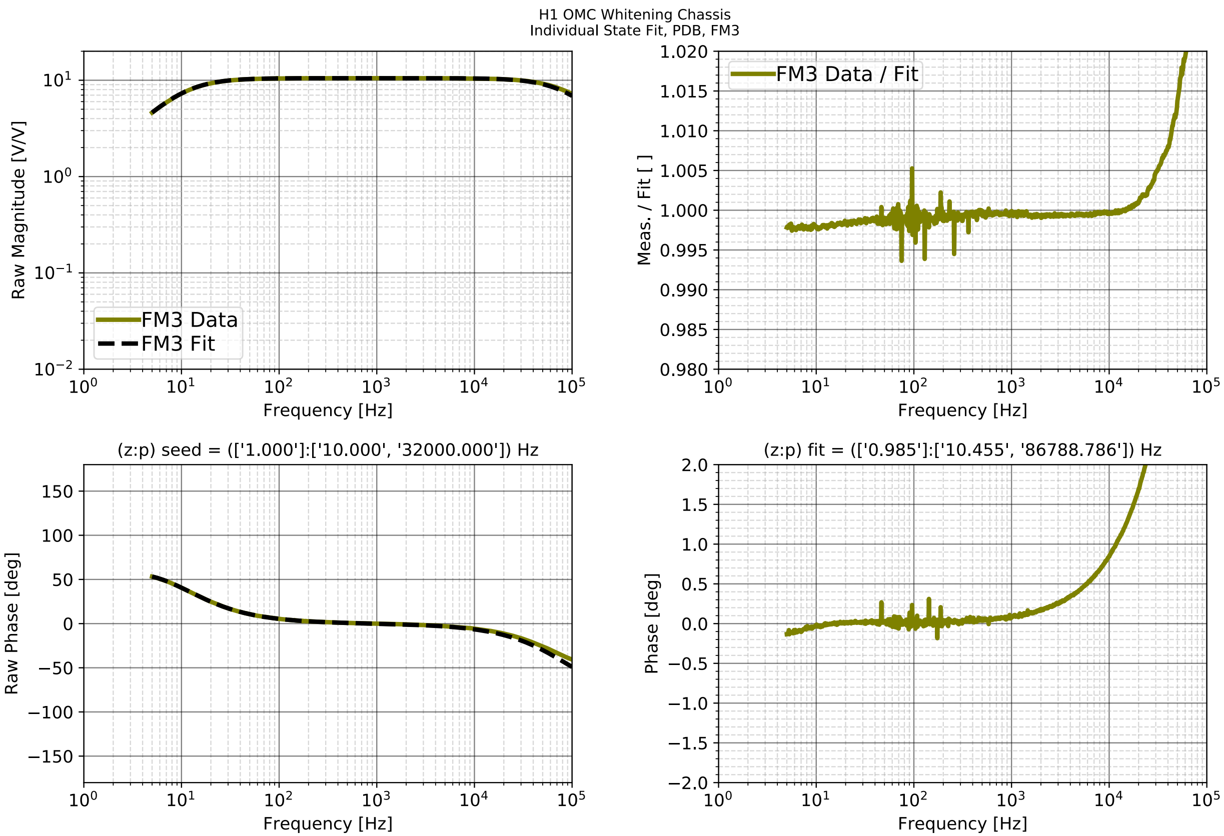


Ah, that's reassuringly simple. BUT – we see much more clearly that the data only goes down to 5 Hz and up to 100 kHz, so fitting for the 1 Hz zero and 0.1 MHz pole will be error prone.

So again, we need better data, but we'll proceed with what we have…

II.5 Fitting and Predicting Systematic Error

A.iii. Fit individual stages, obtain expected zeros and poles



2nd Whitening Stage, PDB

Expected:

$f_z^F : f_p^F = 0.960 : 10.073$ Hz

$f_p^F \approx 0.106$ MHz

Previous Results:

$f_z^F : f_p^F = 1.000 : 10.467$ Hz

$f_p^F = 0.0328$ MHz

$G = 0.9973$

Seed (PDB):

$f_z^F : f_p^F = 1.0 : 10.0$ Hz

$f_p^F = 0.032$ MHz

New Results (PDB):

$f_z^F : f_p^F = 0.985 : 10.455$ Hz

$f_p^F = 0.868$ MHz

Fits are now much more simple – but the fit for the two whitening stages are limited because they're trying to fit for zeros or poles outside the data's frequency region.

At the low end, **given incomplete data for the 1 Hz, the fitter yields a residual with ~0.25% error in magnitude.**

I think this is why Stefan felt the need to add a little gain modifiers to each compensation filter as reported and installed in LHO:47257.

II.5 Fitting and Predicting Systematic Error

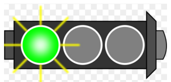
A.iii. Fit individual stages, obtain expected zeros and poles

PDA	New Result Zeros	Old Result Zeros	New Result Poles	Old Result Poles	Old Result Gain
Fixed Response	0.2057		0.0489, 13612.7, 17873.2	11346	
1^{st} Whitening Stage	0.9894	0.969	10.4215, 98277.1	10.440, 32875.3	G = 0.9734
Low Pass	501.119	497.5	49.7277	49.63	G = -0.9995
2^{nd} Whitening Stage	0.9677	0.9865	10.3377, 86258.7	10.372, 32875.3	G = 1.0020

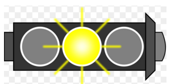
PDB	New Result Zeros	Old Result Zeros	New Result Poles	Old Result Poles	Old Result Gain
Fixed Response	0.2067		0.0505, 13707.1, 18046.0	11521	
1^{st} Whitening Stage	0.979	0.966	10.1639, 98413.7	10.160, 32863	G = 0.975458
Low Pass	501.790	497.7	49.8120	49.72	G = -1.0004
2^{nd} Whitening Stage	0.9845	1.000	10.4551, 86788.8	10.467, 32683	G = 0.9973

II.5 Fitting and Predicting Systematic Error

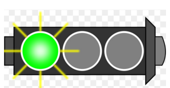
A.iii. Fit individual stages, obtain expected zeros and poles



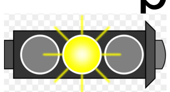
- Most low frequency zeros and poles are very similar,



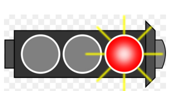
- BUT new fit takes out DC gain from State 0 with nearly cancelling zero:pole pair at ~0.20:0.05 Hz; old fit did it with a gain on each of the State 1, 2, and 3 fits.
  - Which is better? Dunno. Seems like new fit might be conceptually better, but here's where we'd need data down to (past) 1 Hz.
  - Preferably, get data down to 0.5, 0.2, 0.1… for whatever patience allows, and fit again.



- New fit does a *much* better job at fitting for the 0.1 MHz pole



  - (though the State 3 fit still reports ~20 kHz too low)
  - Would be nice to get data out to 200 kHz, or to see if "ratio of State 3 / State 2" data we used agrees with a new measurement of "only State 3 on" but probably not worth chasing



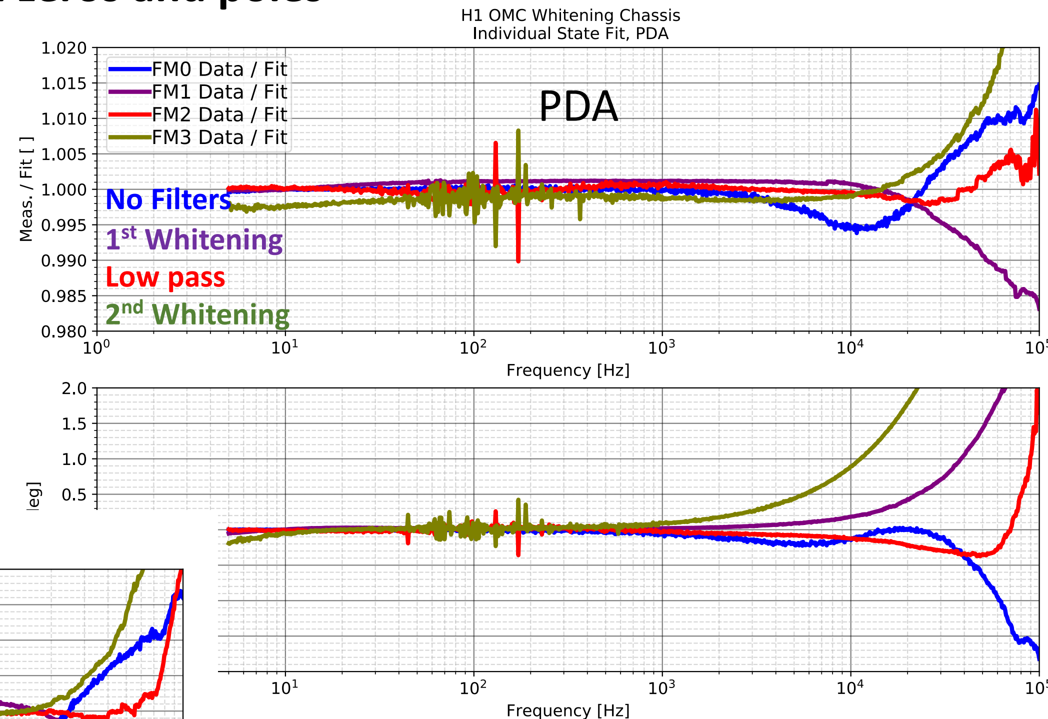
- Both old and new fit for State 0 have a high frequency pole at 11 or 13 kHz, respectively, where there shouldn't be
  - What's going on here? Should it be included? Dunno…

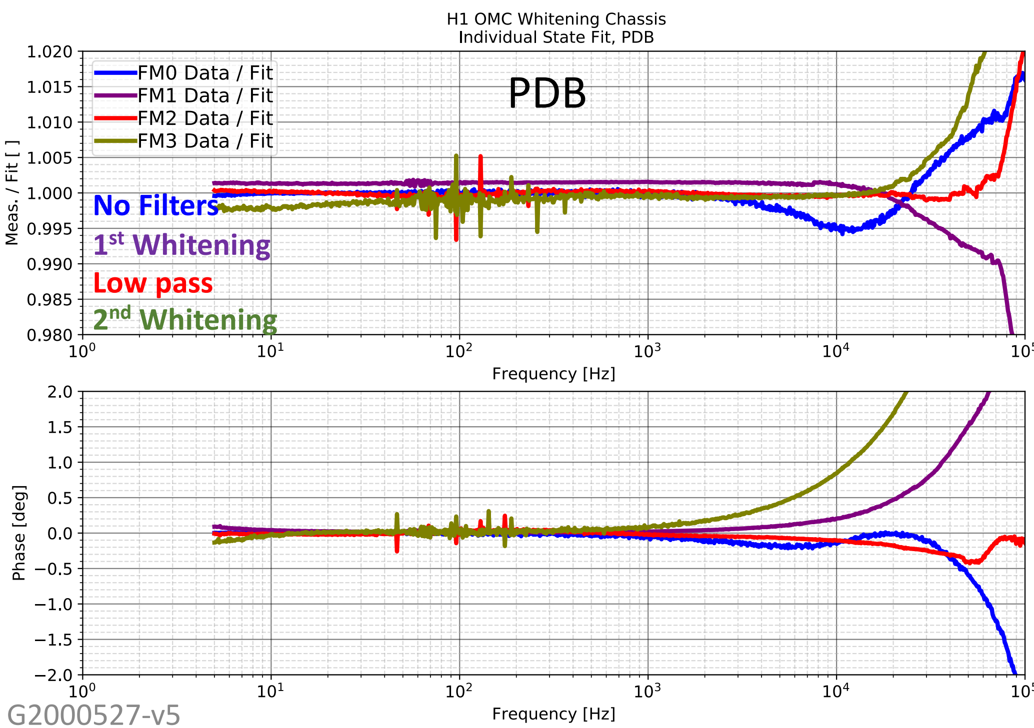
II.5 Fitting and Predicting Systematic Error

A.iii. Fit individual stages, obtain expected zeros and poles

Here's the residuals between each stage and the fit for that stage.

Notice that the fits are pretty dang awesome – but the 1:10 Hz whitening stages have low-frequency gain issues. **Again: this is the result of incomplete data.**



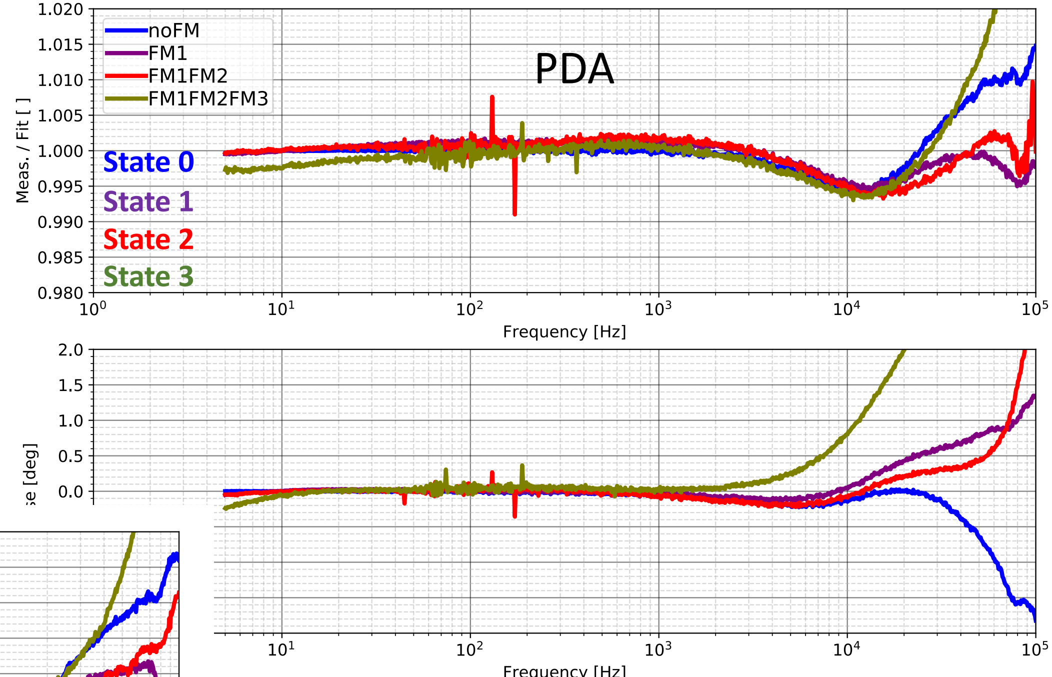


But alas. We move on with the plan, 'cause maybe "perfect is the enemy of good enough."

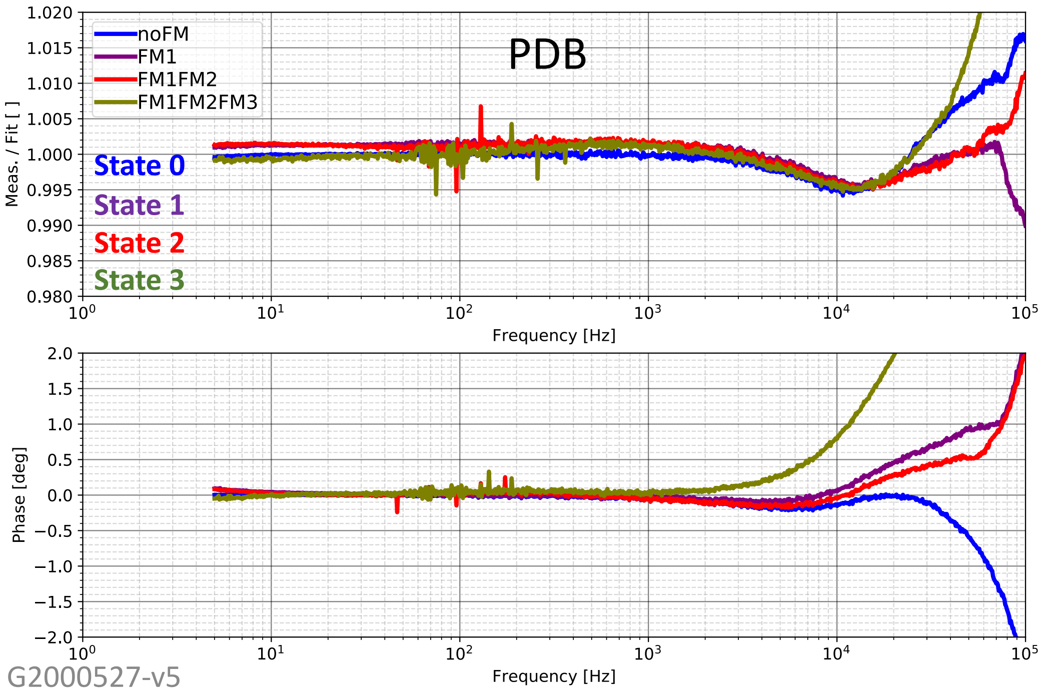
II.5 Fitting and Predicting Systematic Error

A.iv. Stack fit zeros:poles to re-create cascading application of filtering, compare against measurement to create residuals

Take the poles and zeros from the fit each stage, collect them, successively adding them to a new model filter of each state, and divide that cascaded model against the original data



We see how the lack of low frequency data for the whitening stages impacts this residual – especially for state 3, which has the error from both the 1st and 2nd whitening stage



… but the magnitude residual is still only at most 0.0025 away from unity at 10 Hz, and closer than that for most frequencies until 5 kHz.

So now we have a good model of the measurement.
Is it better than the old model that Stefan and Lilli created and installed?

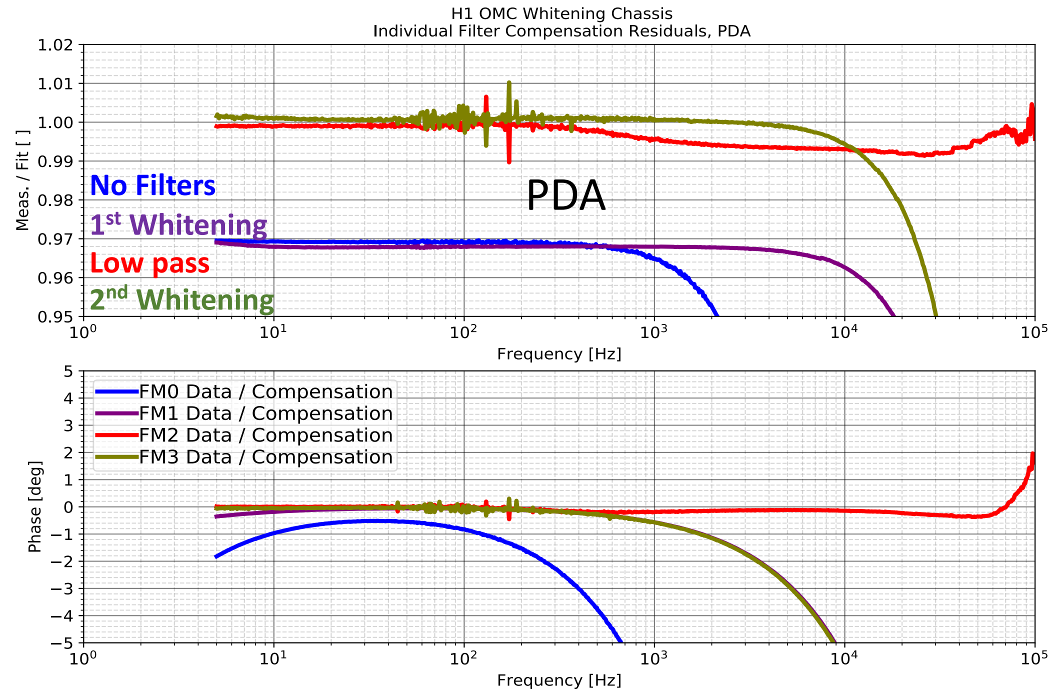
II.5 Fitting and Predicting Systematic Error

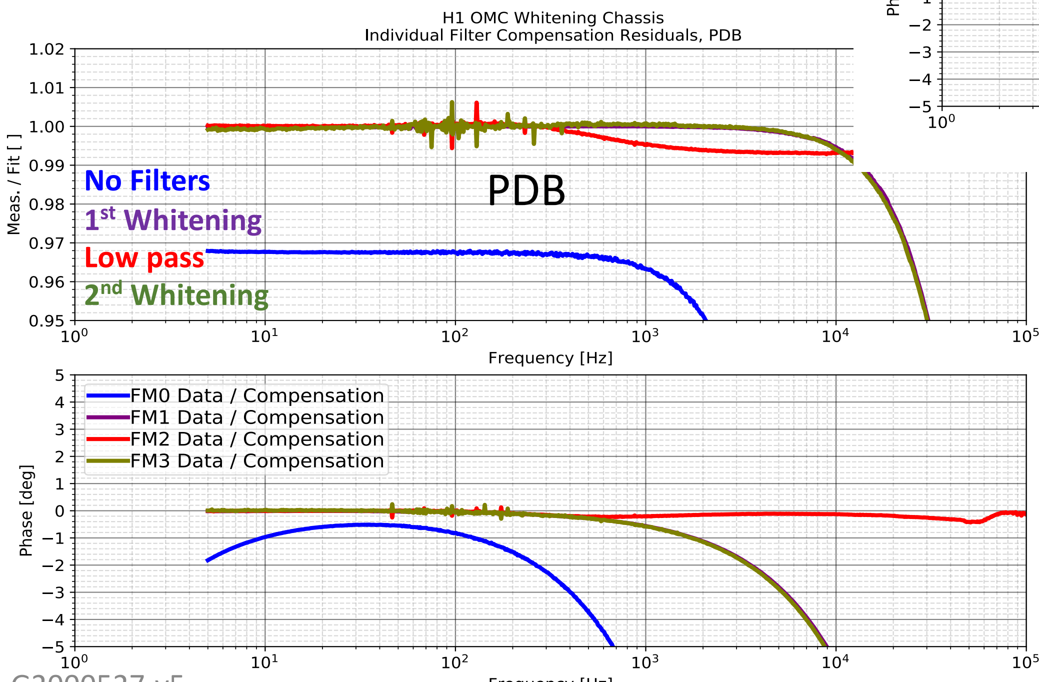
A.v. Reconstruct residuals of compensation we actually used (both in front-end for LF and GDS for HF)

This pulls the LF zeros and poles from Stefan's work.

It does *not* yet include HF poles

The residual is each individual stage's measured response / compensation FM response



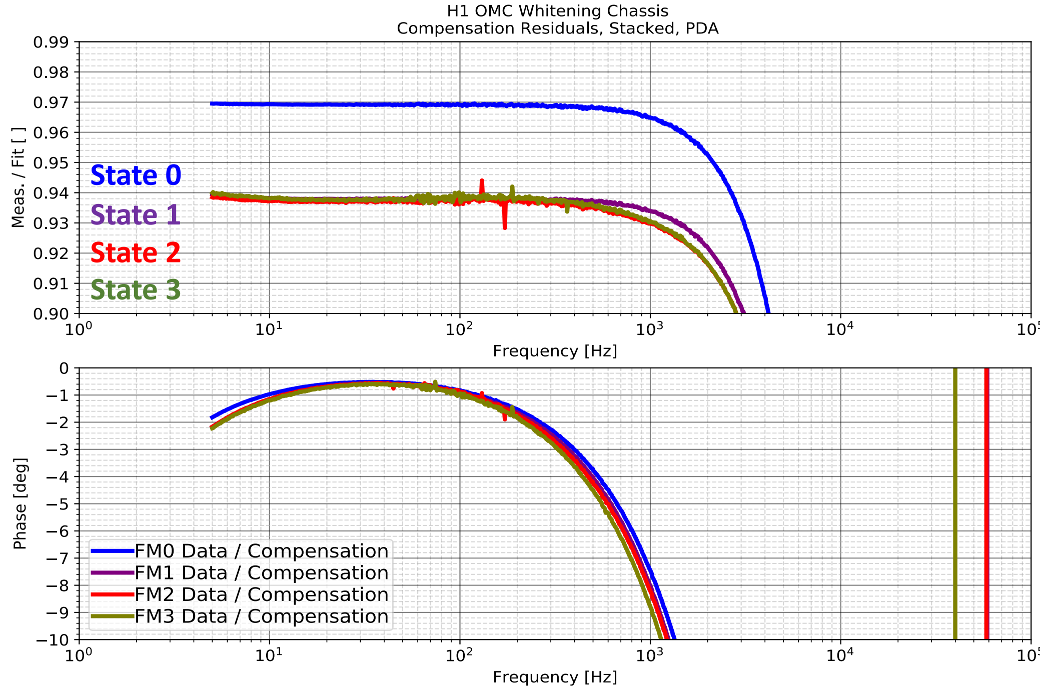


NOTE THE YLIM SCALE CHANGE

- **PDA has a 3% gain error in it's 1st whitening stage compensation, see rabbit hole**
- "No filters" response is clearly missing the 19 kHz pole, fine – we compensate that later. Fine.
- 2nd stage compensation is < 0.1% away from unity magnitude. Interesting.

II.5 Fitting and Predicting Systematic Error

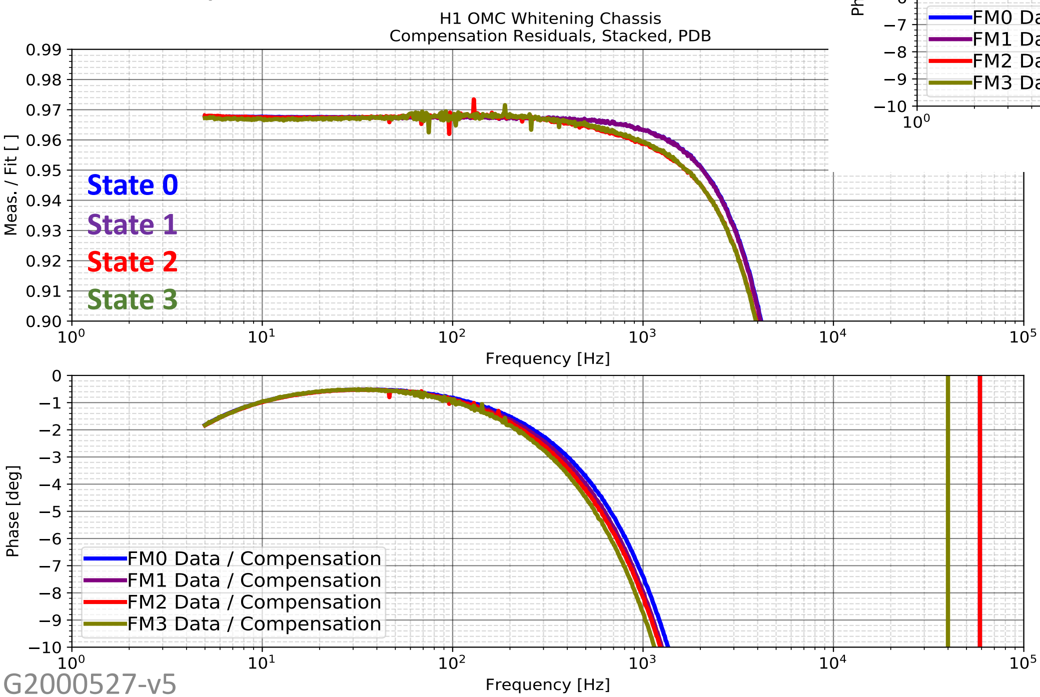
A.v. Reconstruct residuals of compensation we actually used (both in front-end for LF and GDS for HF)



Here's the "stacked" version…

One can see the "DC Gain" error is stacking up – **especially, strangely the 1st whitening stage of PDA**

but the magnitude response is still "flat" up to ~1kHz



Phase is rolling off quickly, but that's the missing compensation for 19 kHz pole.

This gets applied when I create the sum (because that's how it's done in the current calibration scheme)

II.5 Fitting and Predicting Systematic Error

B.i. For each (stacked) filtering stage, sum each paths residuals (use “balance matrix” for compensation version and apply average HF poles)

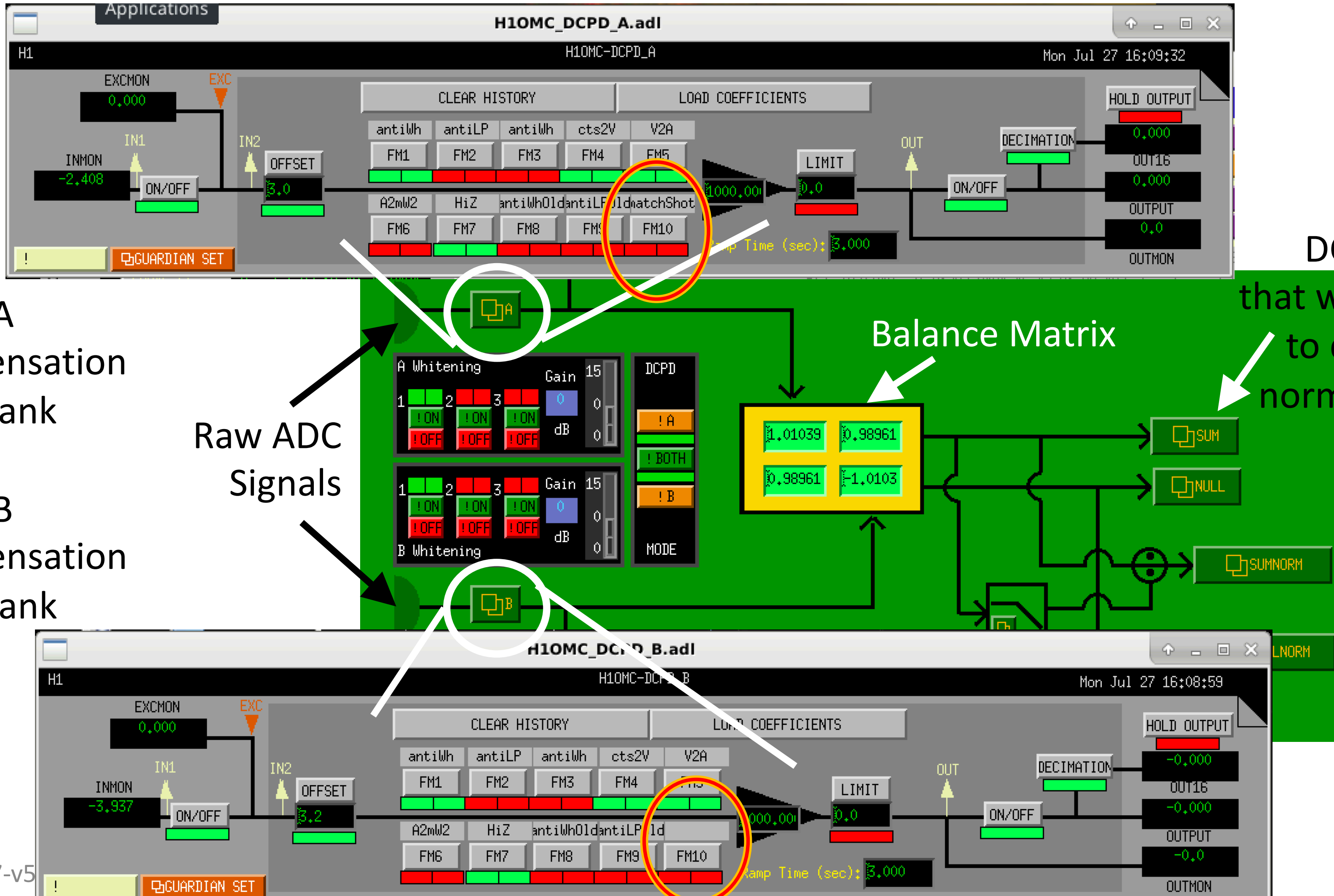
Dude – you don’t think the rabbit hole is deep enough yet???

Let’s talk about the balance matrix and how the A and B paths are matched.

DCPD A Compensation filter bank

DCPD B Compensation filter bank

Raw ADC Signals



Balance Matrix

DCPD SUM that will be fed to quadratic normalization

II.5 Fitting and Predicting Systematic Error

B.i. For each (stacked) filtering stage, sum each paths residuals (use "balance matrix" for compensation version and apply average HF poles)

- Wednesday, Dec 2018, Stefan develops methodology to populate the balance matrix: LHO:45734
- Friday, Mar 01 2019, Stefan updates and augments his methodology documentation (nice!): LHO:47217
- Mar 01 2019, (same Friday) Stefan updates the balance matrix **and turns on FM10 = gain of 0.977**
  - 2019-03-01 22:08 UTC FM10 turned on (Friday at 2pm)
- Saturday, Mar 02 2019 Craig & Stefan discover that the balance is (ah!) DCPD light level dependent! LHO:47247
- Sunday Mar 03 2019, Rich and Peter debug and fix the OMC Whitening Chassis, LHO:47254
- 2019-03-04 02:35 UTC FM10 is turned off and *never used again* (Sunday at 5:30 pm).
- Monday Mar 04, Stefan aLOGs that he's updated the compensation after Rich/Peter chassis fix: LHO:47257, but says:
  - "Finally, we still will have to re-match the photo diode light levels in lock (alog 47217)."
- **…. but I don't think this "re-match" to the photodiode light level -- i.e. re-compensate for the DCPDA channel 0.977 errant gain was ever done – or maybe it was encorporated in to the first FMs?!**

II.5 Fitting and Predicting Systematic Error

B.i. For each (stacked) filtering stage, sum each paths residuals (use "balance matrix" for compensation version and apply average HF poles)

So, in conclusion, what the compensation scheme really should be is:

$$\boldsymbol{Sum_{real} = G^{A}_{overall} * meas^{A}_{LF} * meas^{A}_{HF} * comp^{A}_{LF} * comp^{A}_{HF}\ G^{A}_{bal}}$$

$$\boldsymbol{+\ G^{B}_{overall} * meas^{B}_{LF} * meas^{B}_{HF} * comp^{B}_{LF} * comp^{B}_{HF}\ G^{B}_{bal}}$$

But what we've modelled in O3 assumes that in the above equation,

$$G^{A}_{bal} = G^{B}_{bal} = 1/2$$ Bad assumption

$$G^{A}_{overall} = G^{B}_{overall} = G_{overall} = G_{meas}$$ Pulled out / Folded in to H_C

$$meas^{A}_{LF} * comp^{A}_{LF} = meas^{B}_{LF} * comp^{B}_{LF} = 1$$ Bad assumption

$$comp^{A}_{LF} = comp^{B}_{LF} = comp_{HF} = (meas^{A}_{HF} + meas^{B}_{HF})/2$$ Bad assumption

$$\boldsymbol{Sum_{model} = comp_{HF}}$$ (without allowing for switchable response) Bad assumption

So let's review the *values* for the existing compensation, so we can compare it against the fit.

II.5 Fitting and Predicting Systematic Error

B.i. For each (stacked) filtering stage, sum each paths residuals (use "balance matrix" for compensation version and apply average HF poles)

```
# DCPD SUM Balance Matrix Elements
# LHO aLOG 47228
# https://alog.ligo-
wa.caltech.edu/aLOG/index.php?callRep=47228
        #    PDA       PDB
balance = [1.01039,0.98961]

# Existing compensation:
# Low frequency --
# From LHO aLOG 47257
# and cross checked to be present in
# ^/trunk/Common/H1CalFilterArchive/h1omc/
# v    H1OMC_1254266332.txt
#                          "FM0"   FM1        FM2       FM3
compOMCzeros_whitening = [[ 1.0,  10.440,    49.63, 10.372], #PDA
                          [ 1.0,  10.160,    49.72, 10.467]] #PDB
compOMCpoles_whitening = [[ 1.0,   0.966,    497.5,  0.9865], #PDA
                          [ 1.0,   0.969,    497.7,  1.0000]] #PDB
compOMCgain_whitening  = [[ 1.0, 0.973400, -0.9995,  1.0020], #PDA
                          [ 1.0, 0.975458, -1.0004,  0.9973]] #PDB

# From LHO aLOG 47377
# which has been copied to, e.g. modelparams_H1_20200103.py
#                              PDA          PDB
uncompOMCpoles_whitening = np.array([(11.346e3 + 11.521e3)/2,
                                     (32.875e3 + 32.863e3)/2,
                                     (32.875e3 + 32.863e3)/2])
```

1st Whitening FM1 $f_z^F : f_p^F = 0.960 : 10.073$ Hz

Low pass FM2 $f_z^F : f_p^F = 503.655 : 49.954$ Hz

2nd Whitening FM3 $f_z^F : f_p^F = 0.960 : 10.073$ Hz

$f_p^{ia} = 18.95$ kHz

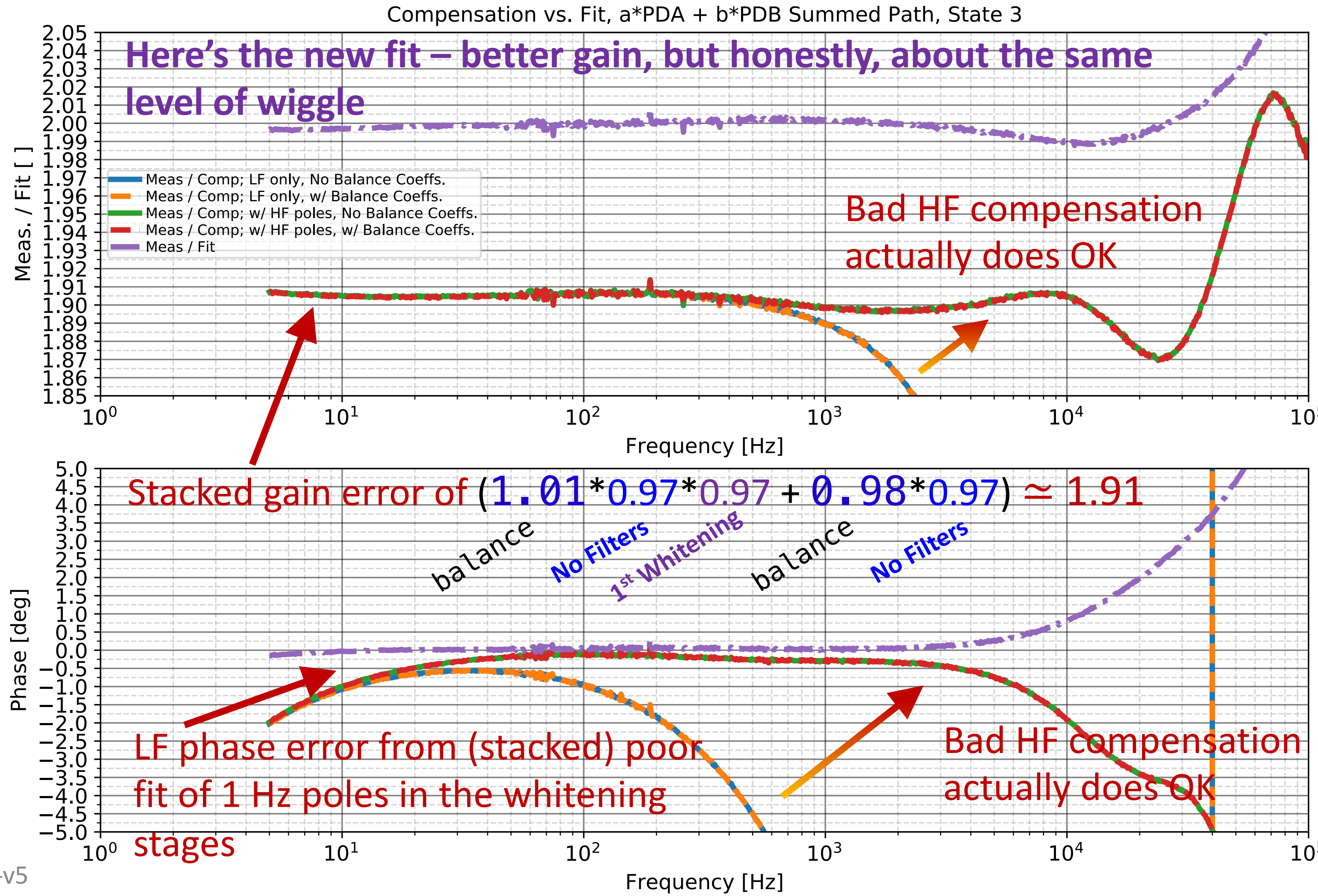
$f_p^F \approx 0.106$ MHz

$f_p^F \approx 0.106$ MHz

II.5 Fitting and Predicting Systematic Error

B.ii. For Each State create a stacked summed residual (both with existing compensation, and with fit)

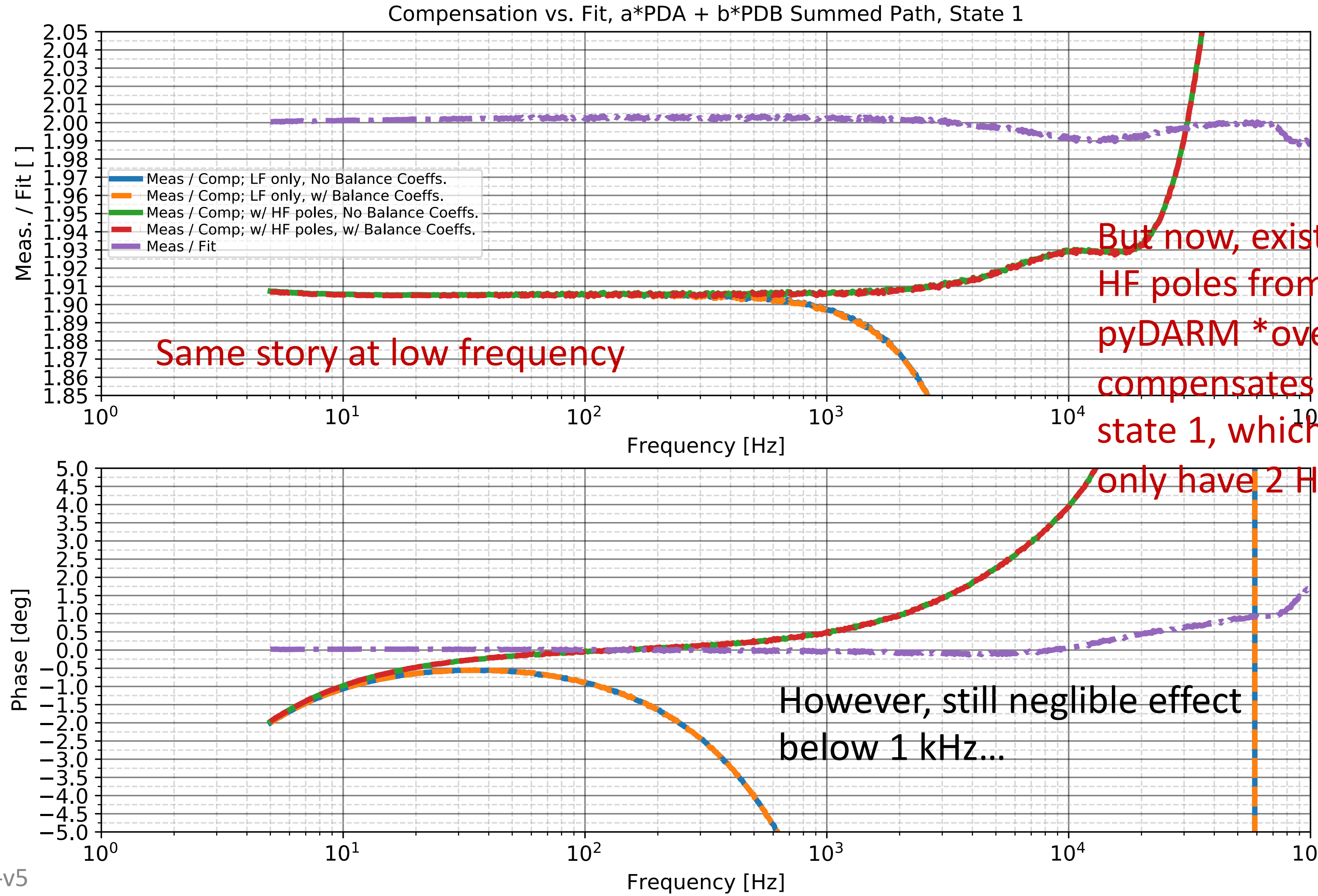
So here's where we were throughout most of O3: State 3, All three filter stages on, the correct number of HF poles, a *gain* flaw, but again is absorbed in G_{meas} (i.e. H_C)



II.5 Fitting and Predicting Systematic Error

B.ii. For Each State create a stacked summed residual (both with existing compensation, and with fit)

So here's where we ended O3: State 1, only one low-pass on



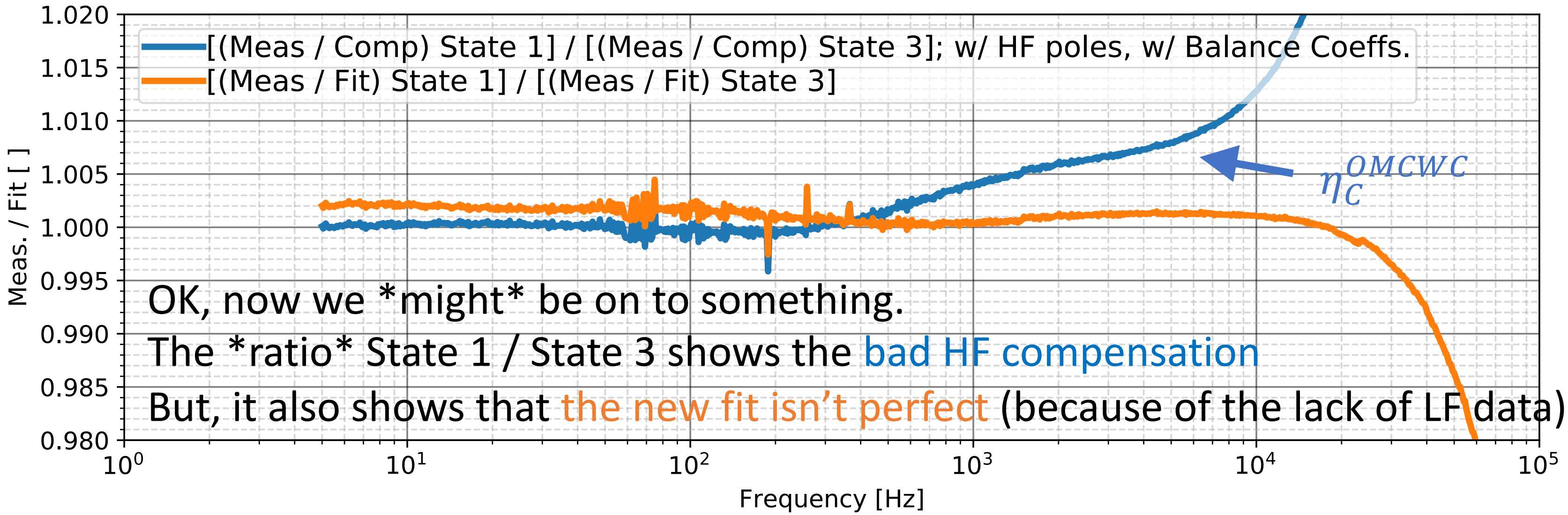
Same story at low frequency

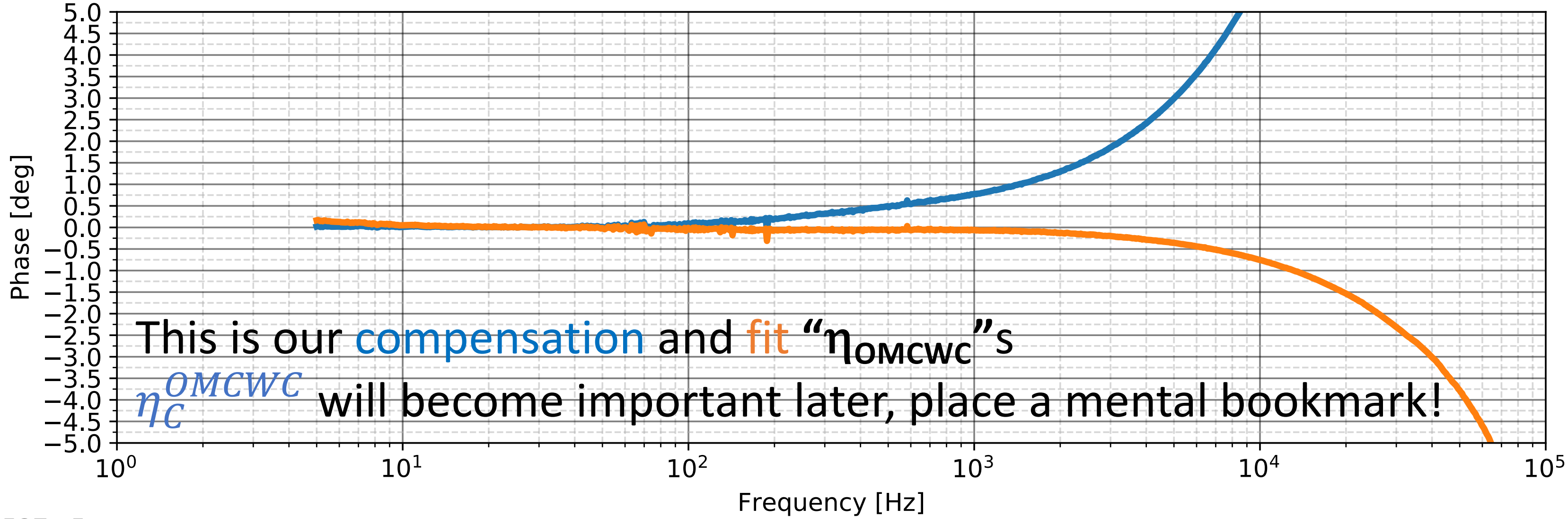
But now, existing 3 HF poles from pyDARM *over* compensates for state 1, which should only have 2 HF poles

However, still neglible effect below 1 kHz…

II.5 Fitting and Predicting Systematic Error

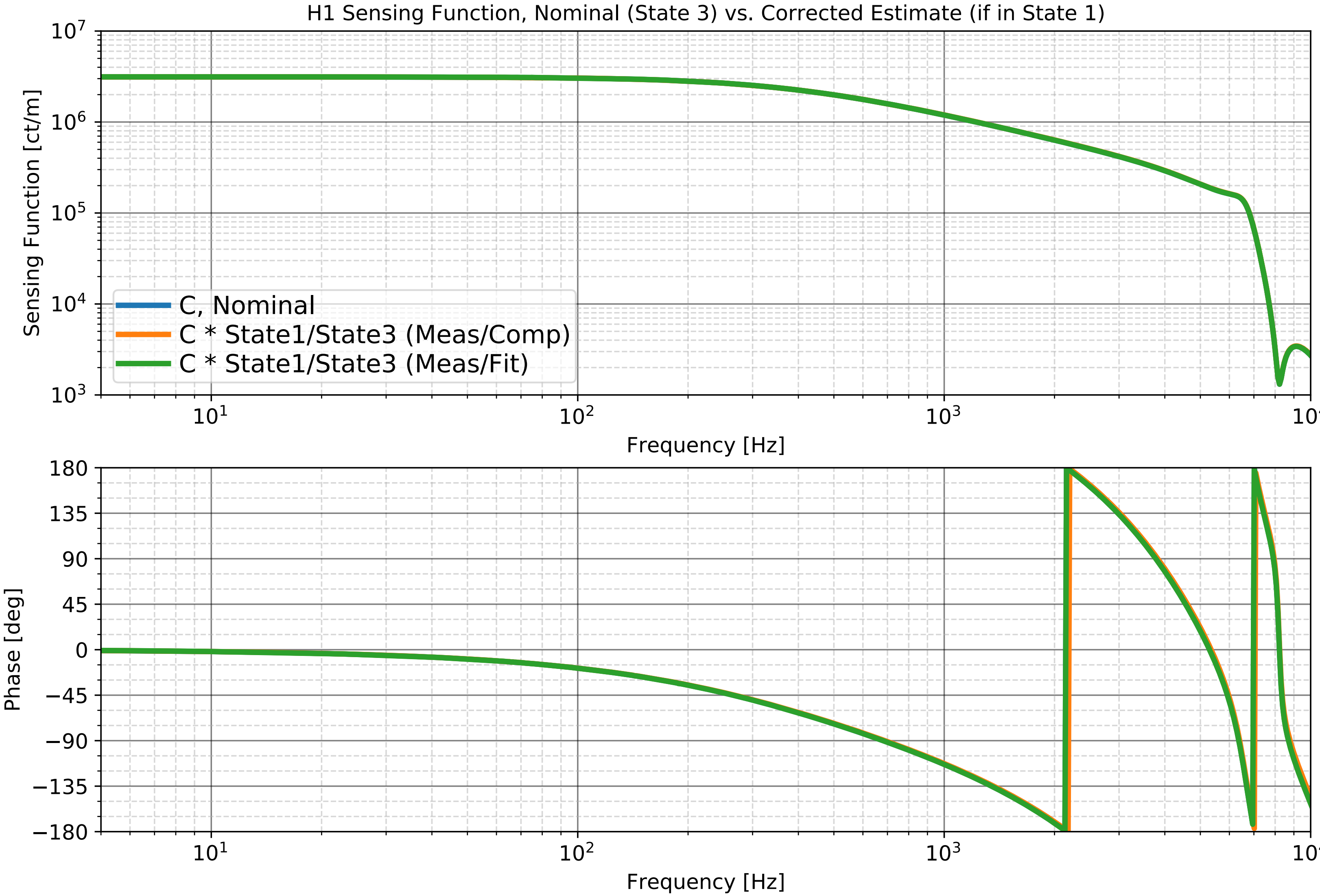
B.ii. Take the ratio of State 1 (one Whitening Stage – i.e. configuration after Mar 10), and State 3 (both whitening stages and the low pass – i.e. what we had for all prior times in O3)





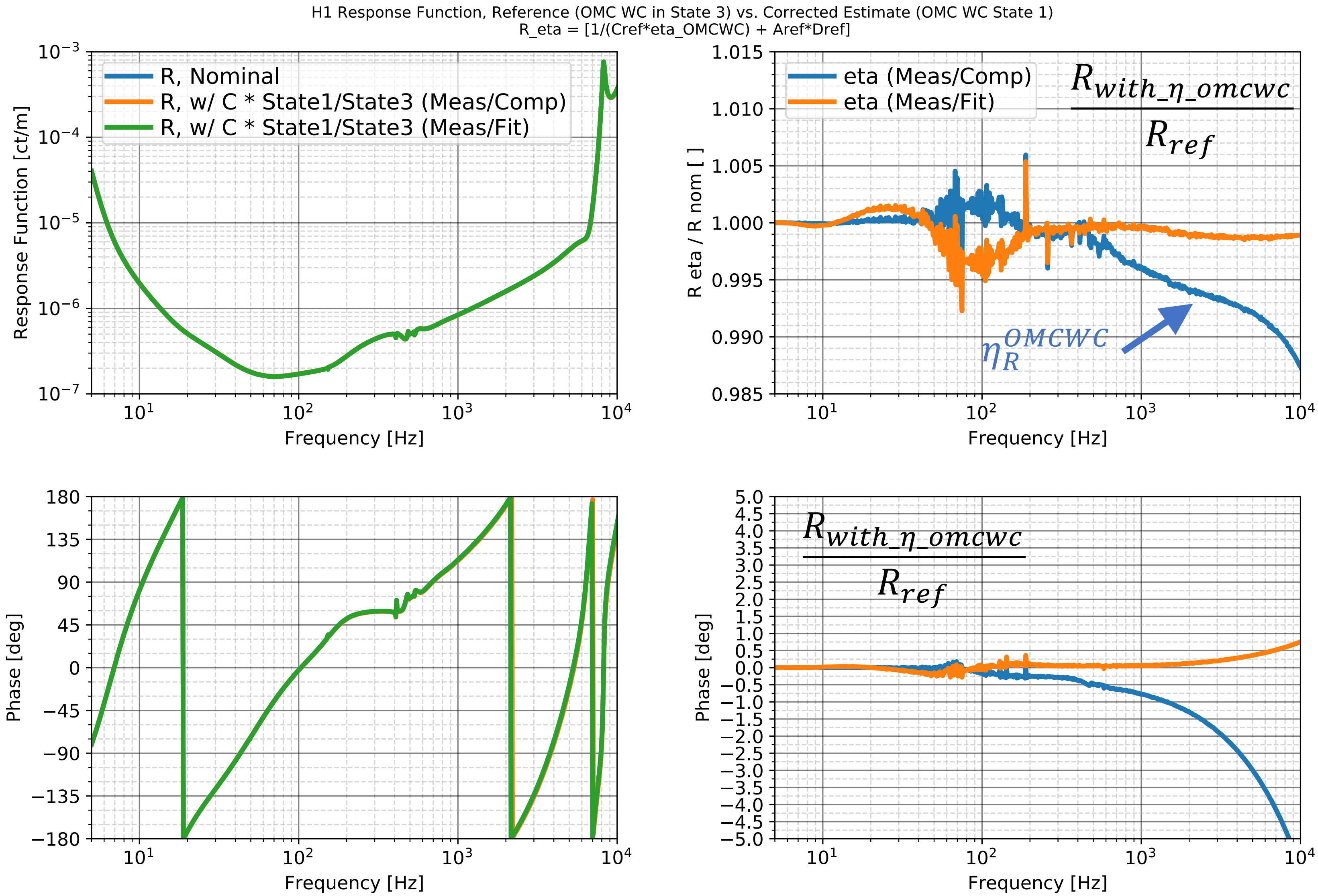
II.5 Fitting and Predicting Systematic Error

C.i. Use ratio of State 1 to State 3 as "η_{OMCWC}" for C



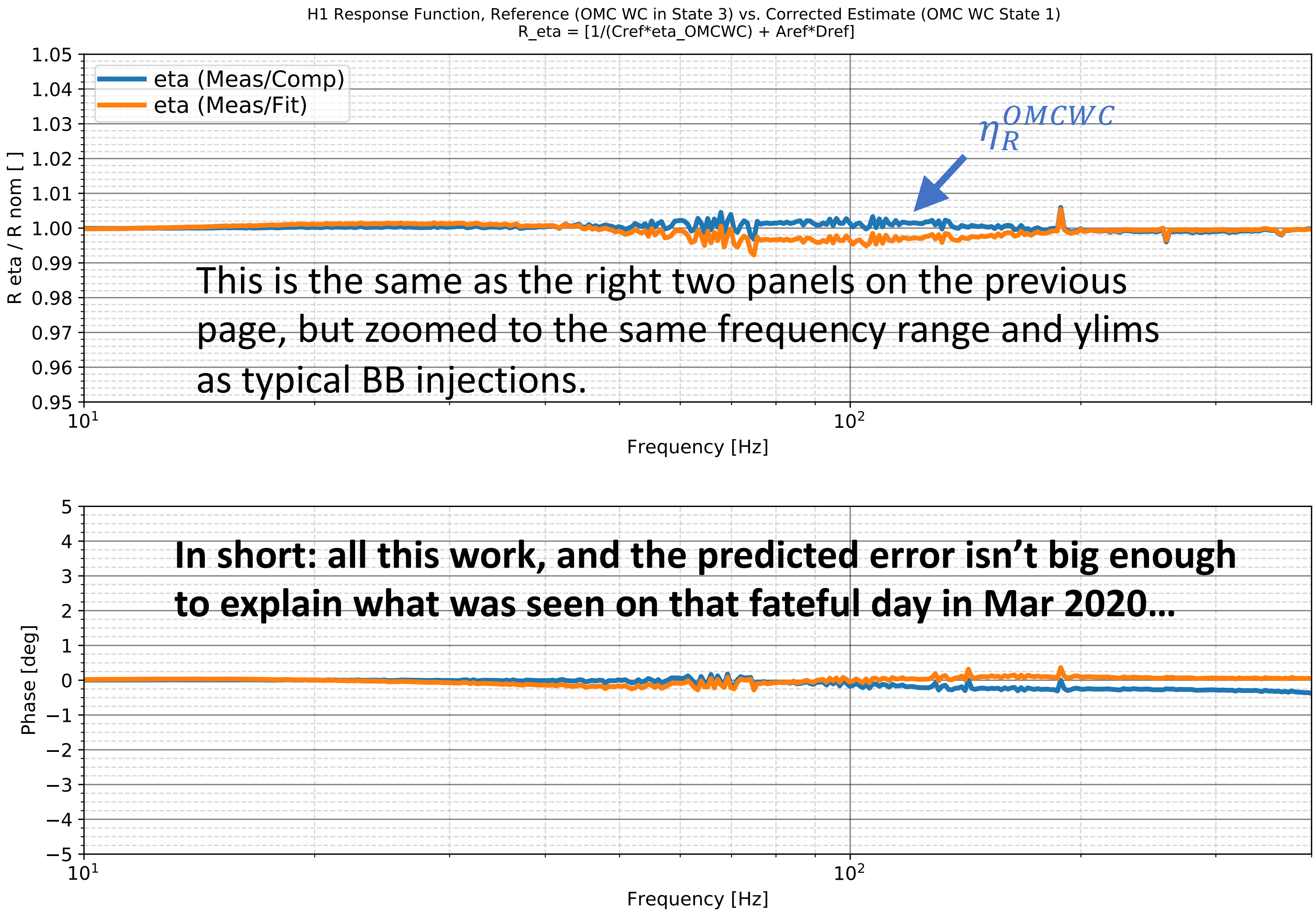
II.5 Fitting and Predicting Systematic Error

C.ii. Compute reference R from 20200103 model, then a modified R using C* η_{OMCWC}



II.5 Fitting and Predicting Systematic Error

C.ii. Compute reference R from 20200103 model, then a modified R using C* η_{OMCWC}



Outline

Two Parts, each quite long. *sigh*

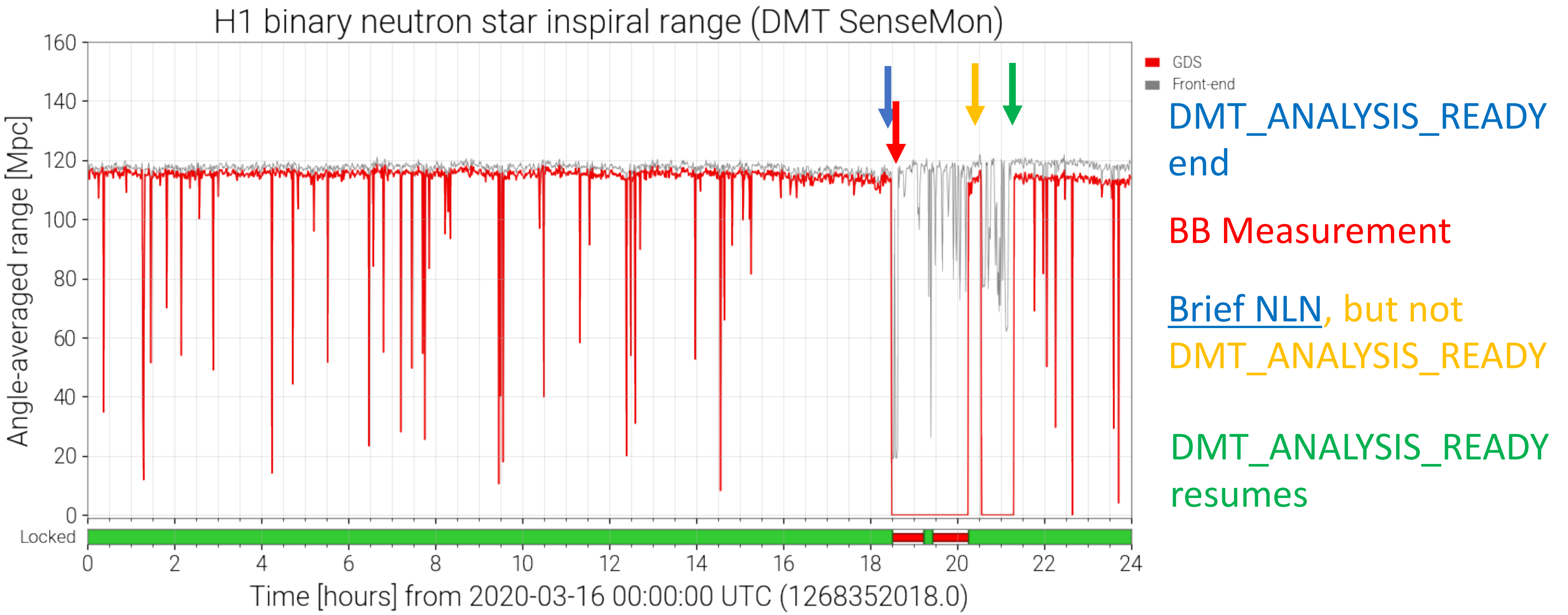
PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.6 Compare Against Measurement

Let's review what before vs. after broadband injection data is available.

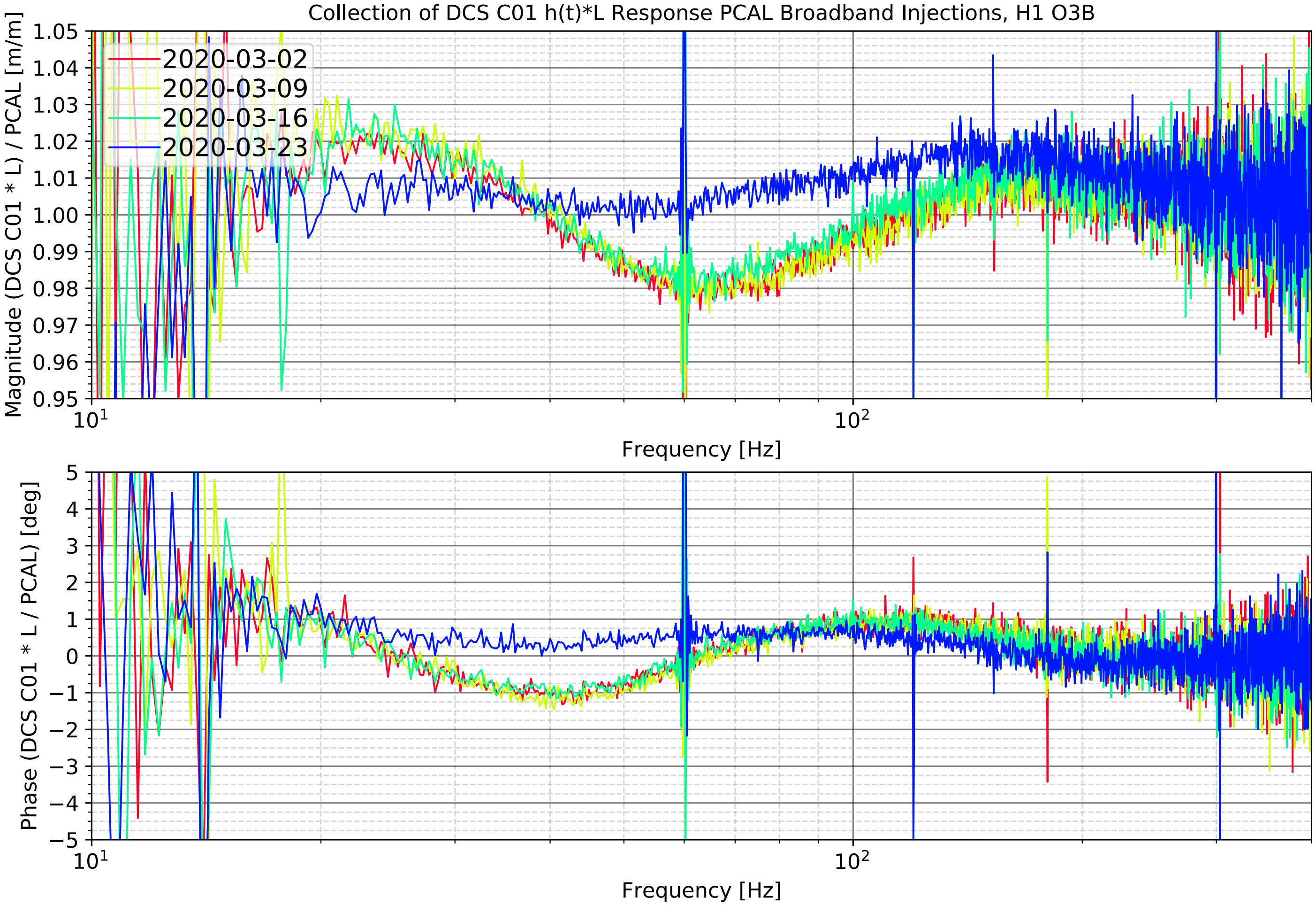


Detector was locked and happy for ~19 hours. Went out of OBS_READY at Mar 16 2020 18:29:59 UTC, switched whitening config, and measured broadband 30 seconds afterword.

- Pre
  - 2020-03-02_H1_PCALY2DARMTF_BB_3min.xml: 2020-03-02 19:00:32 UTC
  - 2020-03-09_H1_PCALY2DARMTF_BB_3min.xml: 2020-03-09 18:00:33 UTC
- Post
  - 2020-03-16_H1_PCALY2DARMTF_BB_3min.xml: 2020-03-16 18:30:31 UTC
  - 2020-03-23_H1_PCALY2DARMTF_BB_3min.xml: 2020-03-23 18:01:20 UTC

II.6 Compare Against Measurement

But somethings already fishy here…

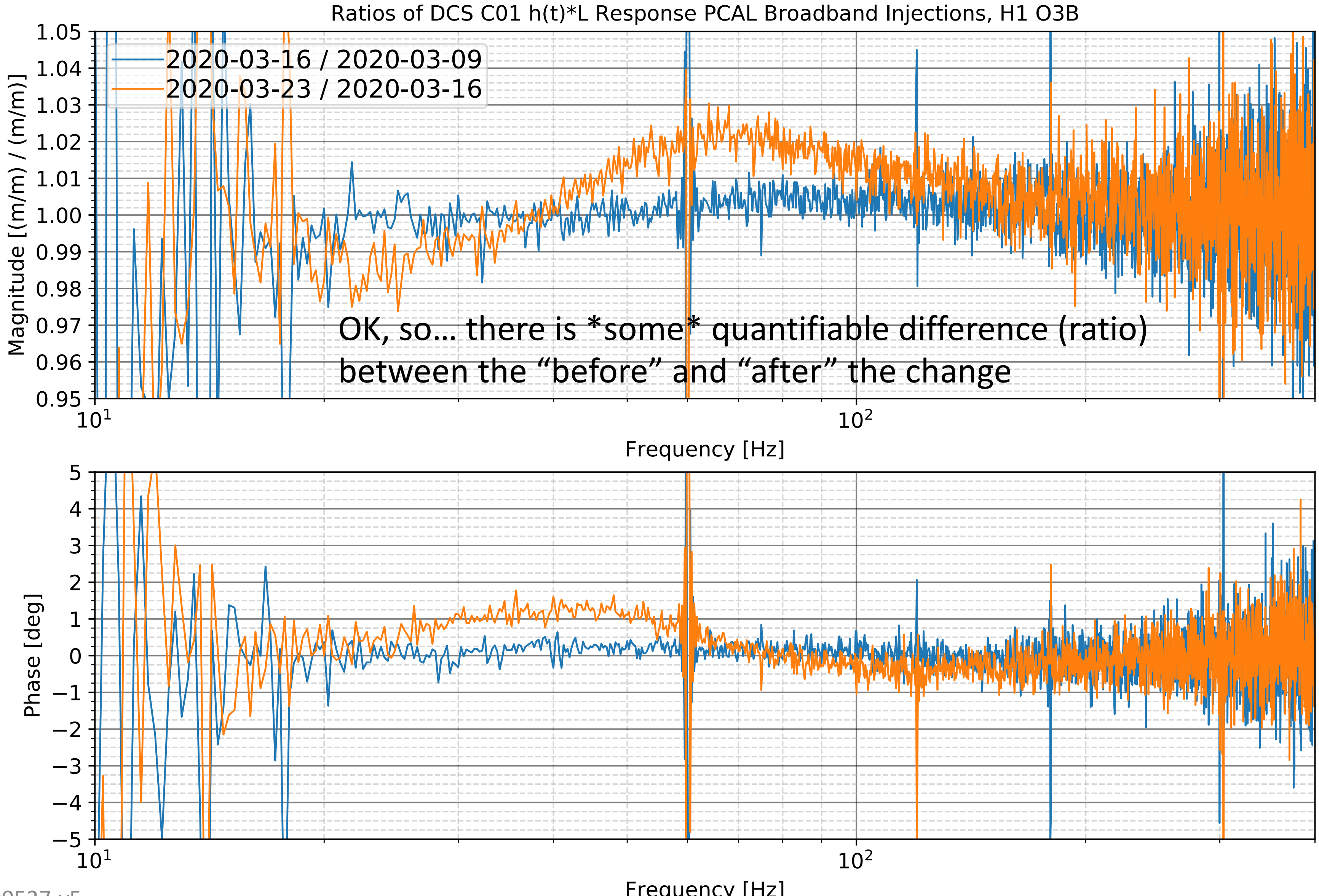


If the whitening configuration changed *before* the 2020-03-16 measurement, shouldn’t there be a difference?

Ok – maybe if you squint… you can see a change between 2020-03-09 and 2020-03-16… but what the heck is going on with 2020-03-23?

II.6 Compare Against Measurement

Maybe we can make sushi out of the fish?



OK, so… there is *some* quantifiable difference (ratio) between the "before" and "after" the change

II.6 Compare Against Measurement

Now, before we compare slide 122 with 118 this is *always* super duper confusing, let's make sure we plot the right thing, and compare apples to apples instead of apples to (1/apples).

On page 115, we've defined the systematic error in C as

$$\eta_{omcwc} \equiv \frac{C_{state\ 1}}{C_{state\ 3}} = \frac{C_{after}}{C_{before}}$$

Then, on page 117 and 118 we show

$$R_{reference} = \frac{1}{C_{before}} + AD$$

$$R_{with\_\eta\_omcwc} = \frac{1}{C_{after}} + AD = \frac{1}{C_{before} * \eta_{omcwc}} + AD$$

$$\eta_R = \frac{R_{with\_\eta\_omcwc}}{R_{reference}}$$

For the measurements on page 121 and 121, remember from T1900169

$$\frac{\Delta L}{PCAL} = \frac{R_{DCS}}{R_{PCAL}}$$

DCS response function isn't changing – the PCAL excitation is measuring the change,

$$\left.\frac{\Delta L}{PCAL}\right|_{2020-03-09} = \frac{R_{DCS}}{R_{reference}}$$

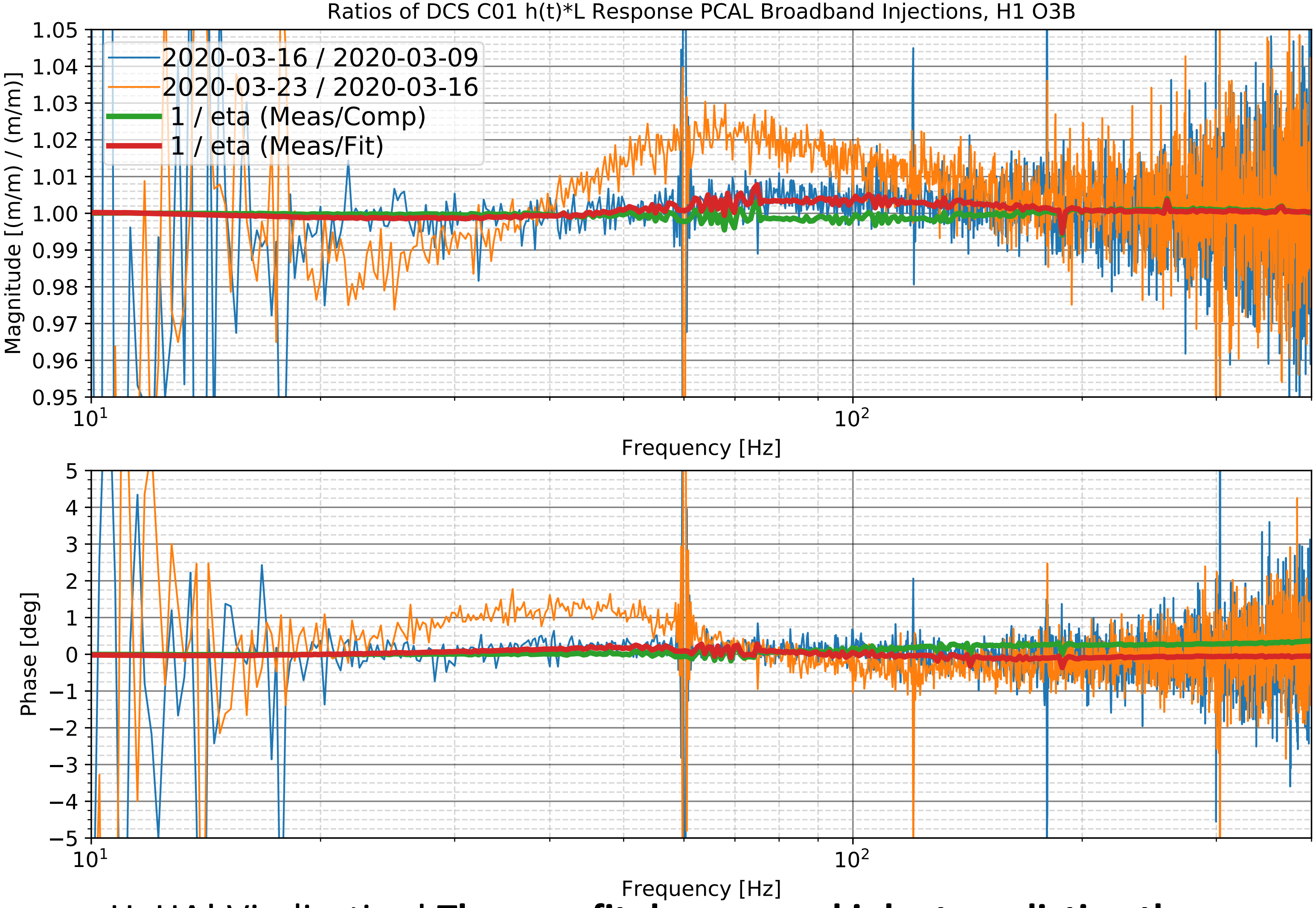
$$\left.\frac{\Delta L}{PCAL}\right|_{2020-03-16} = \frac{R_{DCS}}{R_{with\_\eta\_omcwc}}$$

Which means

$$\frac{\boldsymbol{\Delta L/PCAL|_{200316}}}{\boldsymbol{\Delta L/PCAL|_{200309}}} = \frac{\boldsymbol{R_{reference}}}{\boldsymbol{R_{with\_\eta\_omcwc}}} = \frac{\mathbf{1}}{\boldsymbol{\eta_R}}$$

II.6 Compare Against Measurement

Sushi go!



HaHA! Vindication! **The new fit does a good job at predicting the *miniscule* change in the response function. Great!**

So… what the heck is going on with the 2020-03-23 measurement??

Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What’s the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.7 Collateral Damage on/from TDCFs

- G2001293 started out as an unrelated investigation in to why the cavity pole changed by 7 Hz after the OMC whitening chassis change.

- However, the result of that investigation concludes that **the OMC whitening chassis error** (predicted in this talk) **at 410.3 Hz actually causes the sensing function TDCFs to falsely report a change in the optical gain and cavity pole.**

- These poorly informed TDCFs were then applied to h(t), and thus create a self-inflicted systematic error. In G2001293, we show that **accounting for this error almost entirely accounts for the change between the 2020-03-16 measurement and 2020-03-23 measurement.** Mystery solved!

- But, one last thing (picking up after the conclusion of G2001293): to completely account for impact, we will apply both the (negligible) η_C^{OMCWC} and the (more impactful) η_C^{TDCFs} to the Chunk 2, Period c systematic error and uncertainty budget.
  - (where the application of η_C^{OMCWC} will mostly just account for the small amount of error above 1 kHz)

II.7 Collateral Damage on/from TDCFs

The **maaaaaath** covered on page 14 of [G2001293] reminds us that, because of the "1+" part of the response function, the modification to C is not so straightforwardly propagated to a modification in R. So let's pick up where we left off, but include η_C^{OMCWC} as well as η_C^{TDCFs}.

$$\eta_C^{total} \equiv \eta_C^{OMCWC} \eta_C^{inc \gg cor} \eta_C^{ref \gg inc}$$

$$R_{ref} = \frac{1+ADC}{C}$$

$$R_{incorrect} = \frac{1+\kappa_C' \frac{(1+if/f_{cc}^{ref})}{(1+if/f_{cc}')} ADC}{\kappa_C' \frac{(1+if/f_{cc}^{ref})}{(1+if/f_{cc}')} C} = \frac{1+\eta_C^{ref \gg inc} ADC}{\eta_C^{ref \gg inc} C}$$

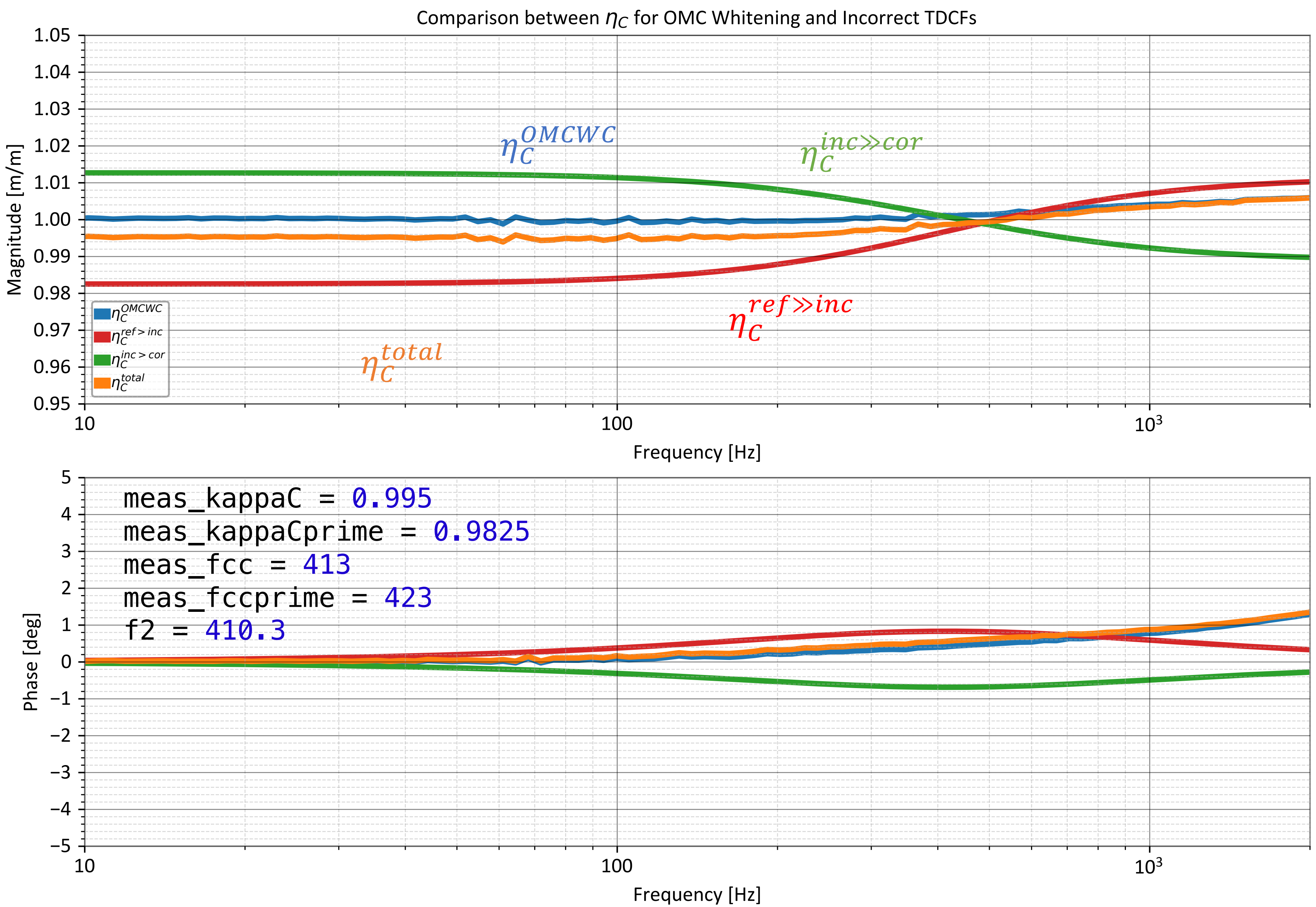
$$\eta_R^{ref \gg inc} \equiv \frac{R_{incorrect}}{R_{ref}}$$

$$\eta_R^{ref \gg cor} \equiv \frac{R_{correct}}{R_{ref}}$$

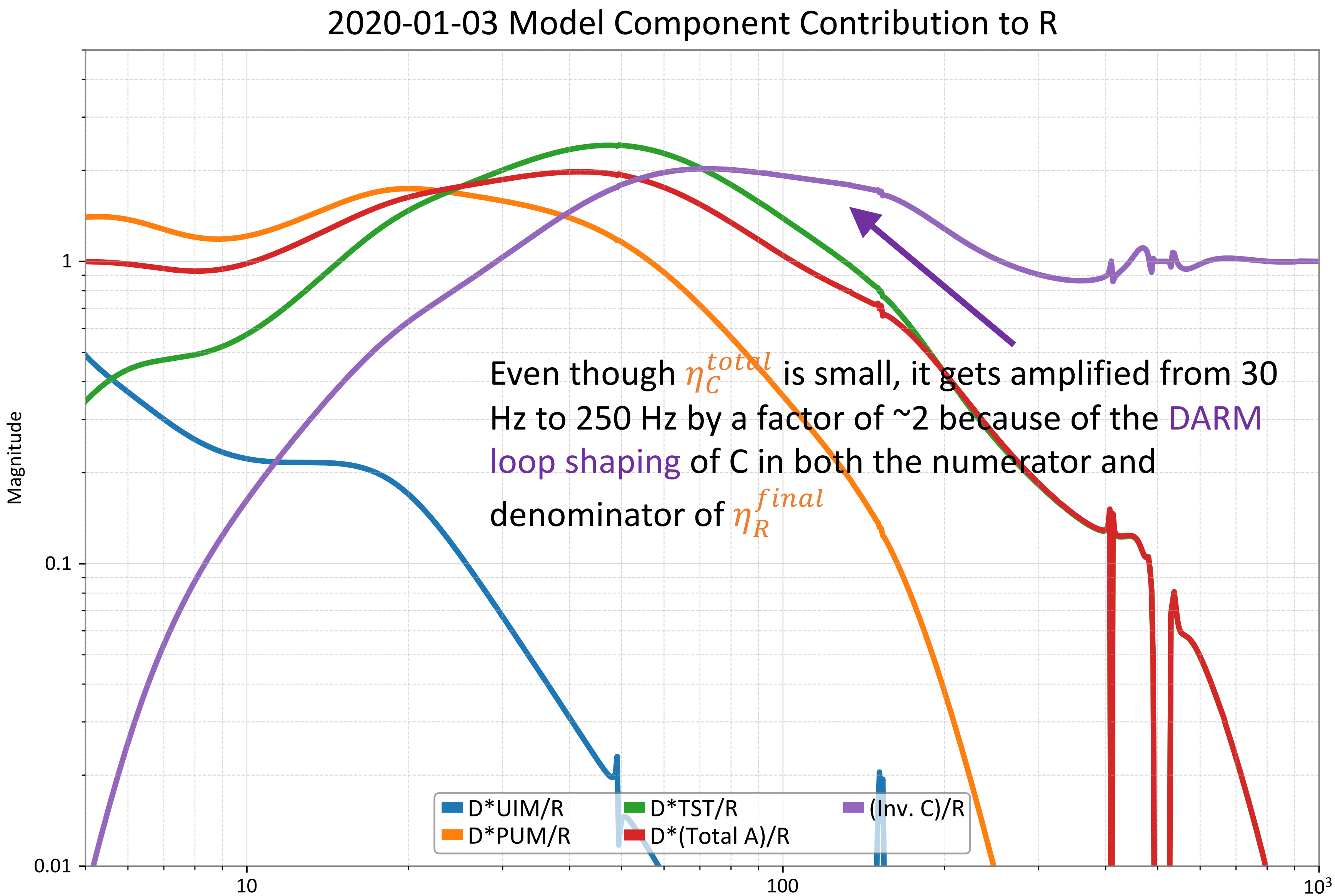
$$R_{correct} = \frac{1+\eta_C^{OMCWC} \frac{\kappa_C}{\kappa_C'} \frac{(1+if/f_{cc}')}{(1+if/f_{cc})} \kappa_C' \frac{(1+if/f_{cc}^{ref})}{(1+if/f_{cc}')} ADC}{\eta_C^{OMCWC} \frac{\kappa_C}{\kappa_C'} \frac{(1+if/f_{cc}')}{(1+if/f_{cc})} \kappa_C' \frac{(1+if/f_{cc}^{ref})}{(1+if/f_{cc}')} C} = \frac{1+\eta_C^{OMCWC} \eta_C^{inc \gg cor} \eta_C^{ref \gg inc} ADC}{\eta_C^{OMCWC} \eta_C^{inc \gg cor} \eta_C^{ref \gg inc} C}$$

$$\eta_R^{final} \equiv \frac{R_{correct}}{R_{incorrect}} = \frac{1+\eta_C^{OMCWC} \eta_C^{inc \gg cor} \eta_C^{ref \gg inc} ADC}{1+\eta_C^{ref \gg inc} ADC} \frac{1}{\eta_C^{OMCWC} \eta_C^{inc \gg cor}}$$

II.7 Collateral Damage on/from TDCFs



II.7 Collateral Damage on/from TDCFs



Outline

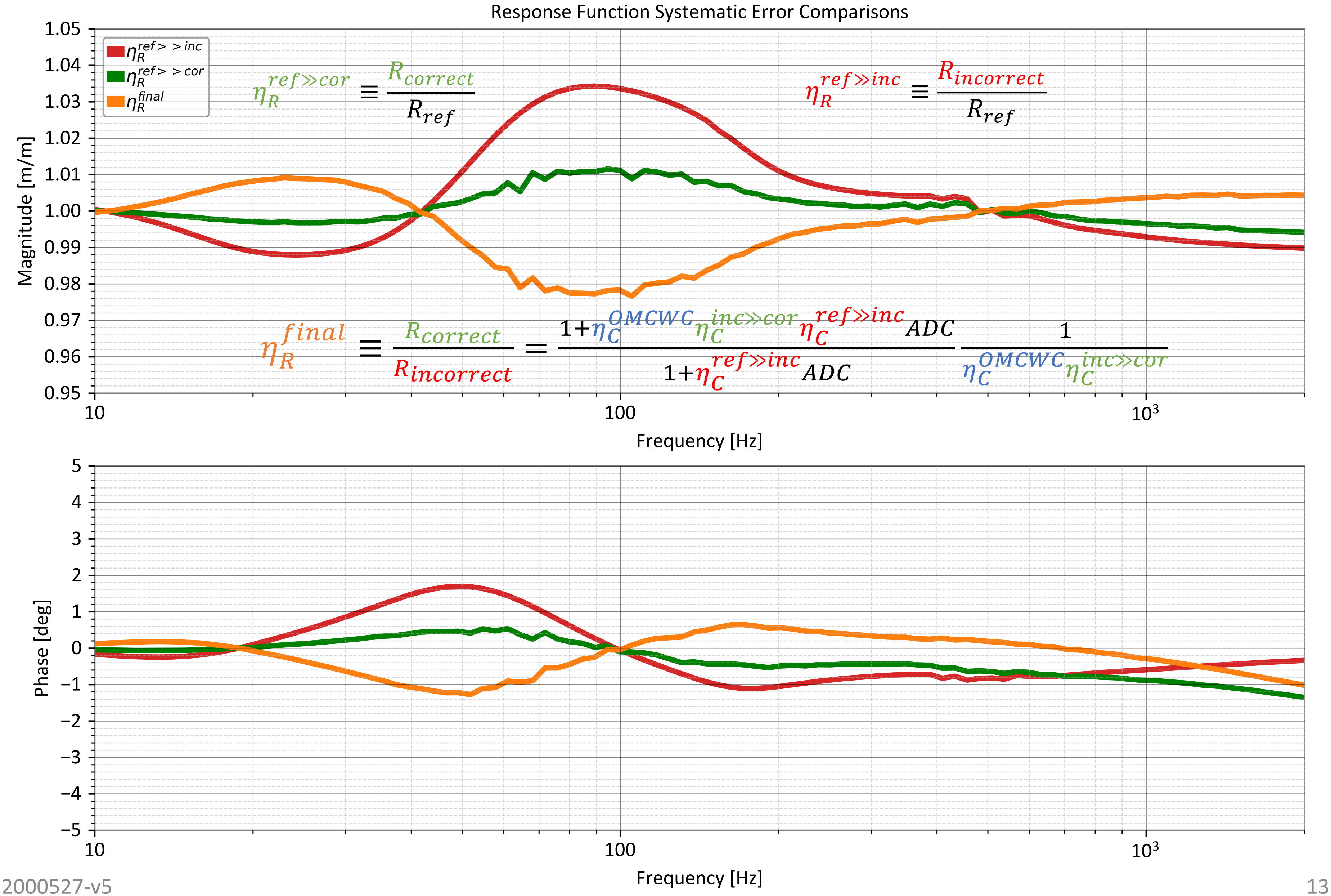
Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

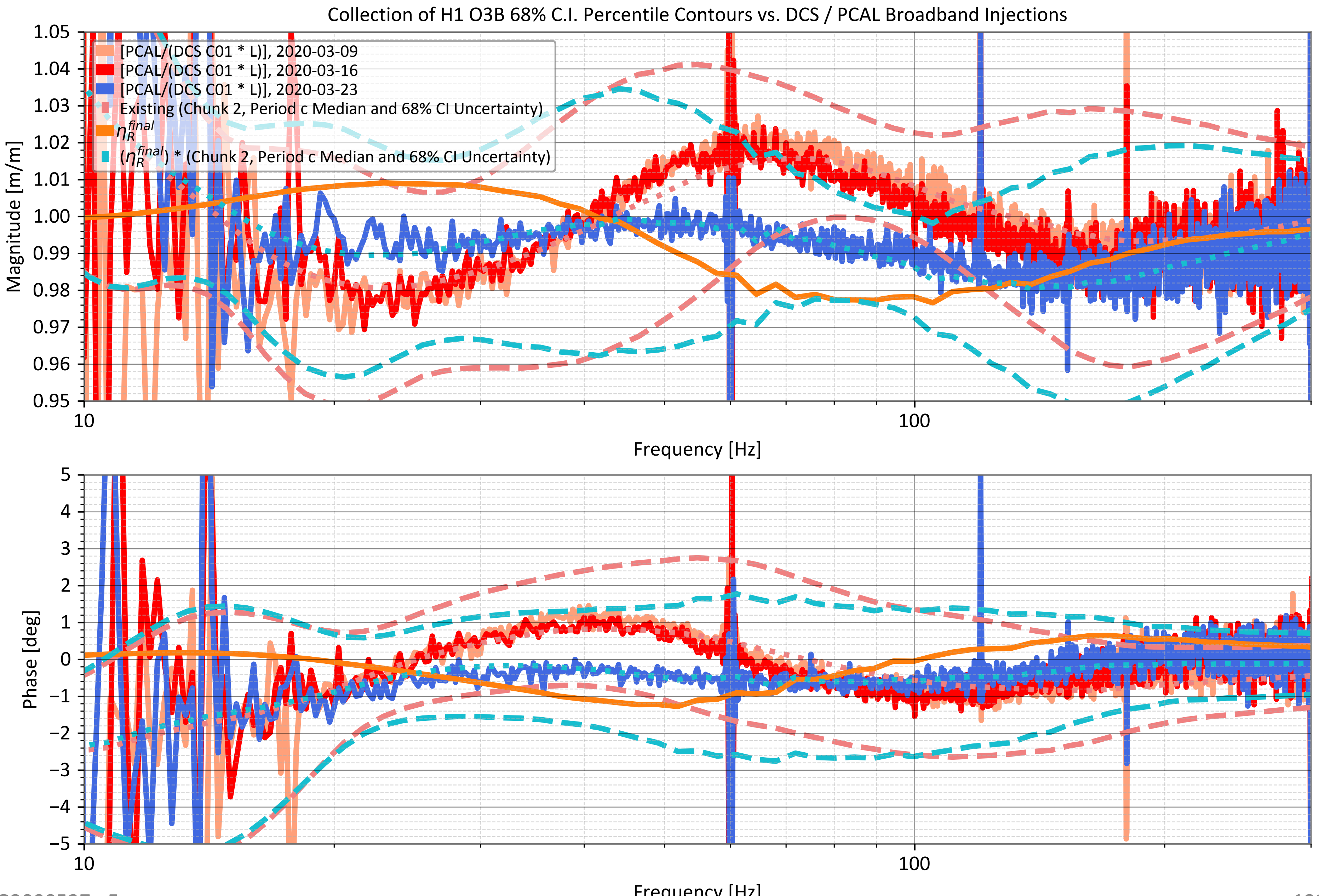
PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What's the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

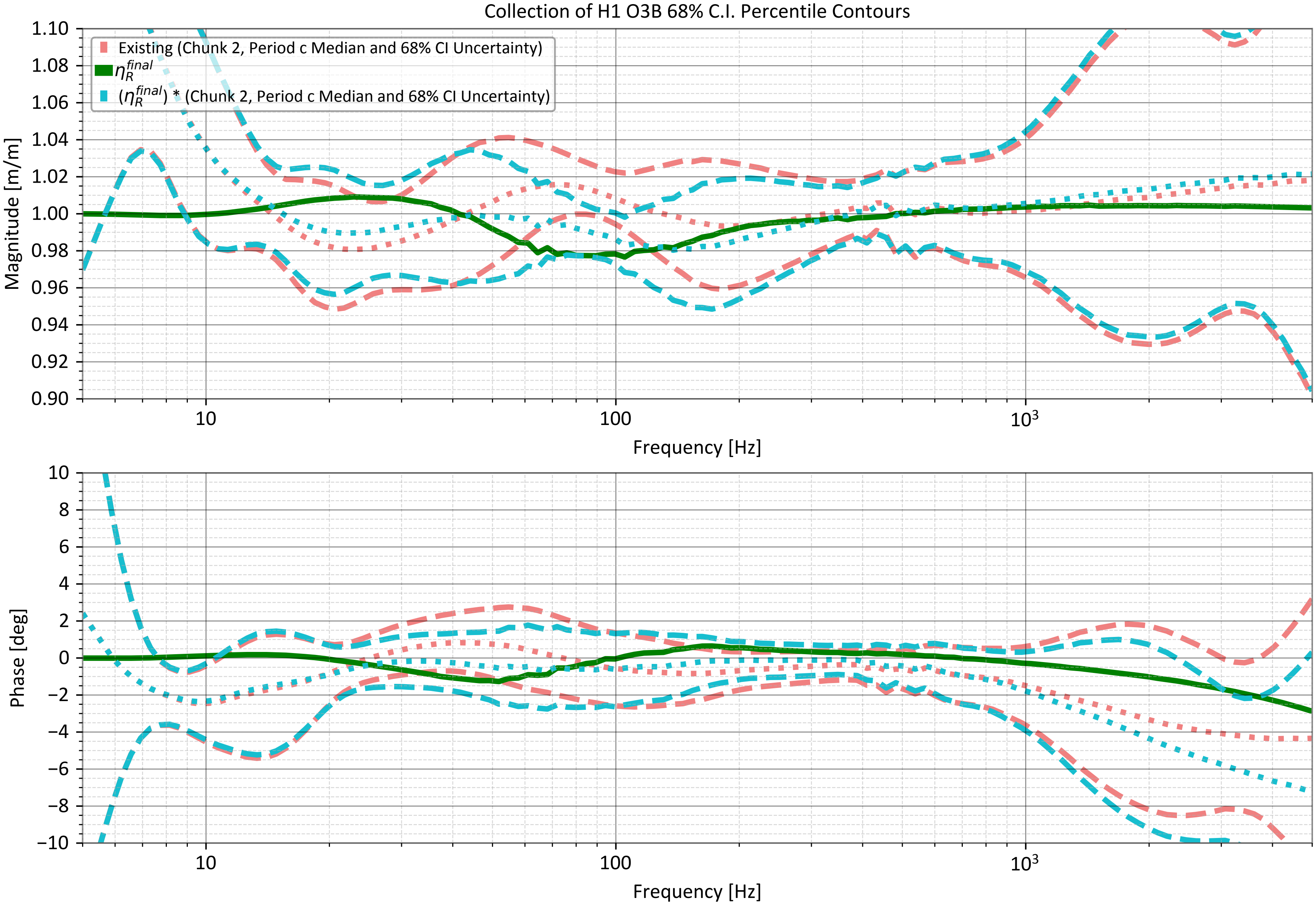
II.8 Final Answer: The construction of η_R^{final}



II.8 Final Answer: succession of BB vs. Modified Percentile for Period c



II.8 Final Answer: Predicted Impact across the Entire Band



II.8 Final Answer

- η_C^{OMCWC} and the bulk of the fitting/modeling plots was produced by

`/ligo/svncommon/CalSVN/aligocalibration/trunk/Common/Electronics/H1/Scripts/`

`fit_OMCDCPDWhiteningChassis_20190304_forG2000527.py`

and exported to

`/ligo/svncommon/CalSVN/aligocalibration/trunk/Runs/O3/H1/Results/Uncertainty/`

`2020-08-25_measDate2019-03-04_H1OMC_WhiteningChassisTFs_eta_C_omcwc.txt`

- $\eta_C^{inc \gg cor} \eta_C^{ref \gg inc}$ and the work presented in [G2001293] was produced by

`/ligo/svncommon/CalSVN/aligocalibration/trunk/Common/Documents/G2001293_O3BChunk2Periodc_OMCWhitening_and_TDCF_SysError/`

`showimpact_omcwhiteningchassiserror_on_sensingTDCFs_and_R.py`

- η_C^{OMCWC}, $\eta_C^{inc \gg cor}$, and $\eta_C^{ref \gg inc}$ were all combined as described on page 127 by

`/ligo/svncommon/CalSVN/aligocalibration/trunk/Runs/O3/H1/Scripts/Uncertainty/`

`combine_OMCWC_and_BadSensingTDCF_syserror_H1O3BChunk2Periodc_20200316-20200327.py`

- The resulting η_R^{final} produce by that script was exported as

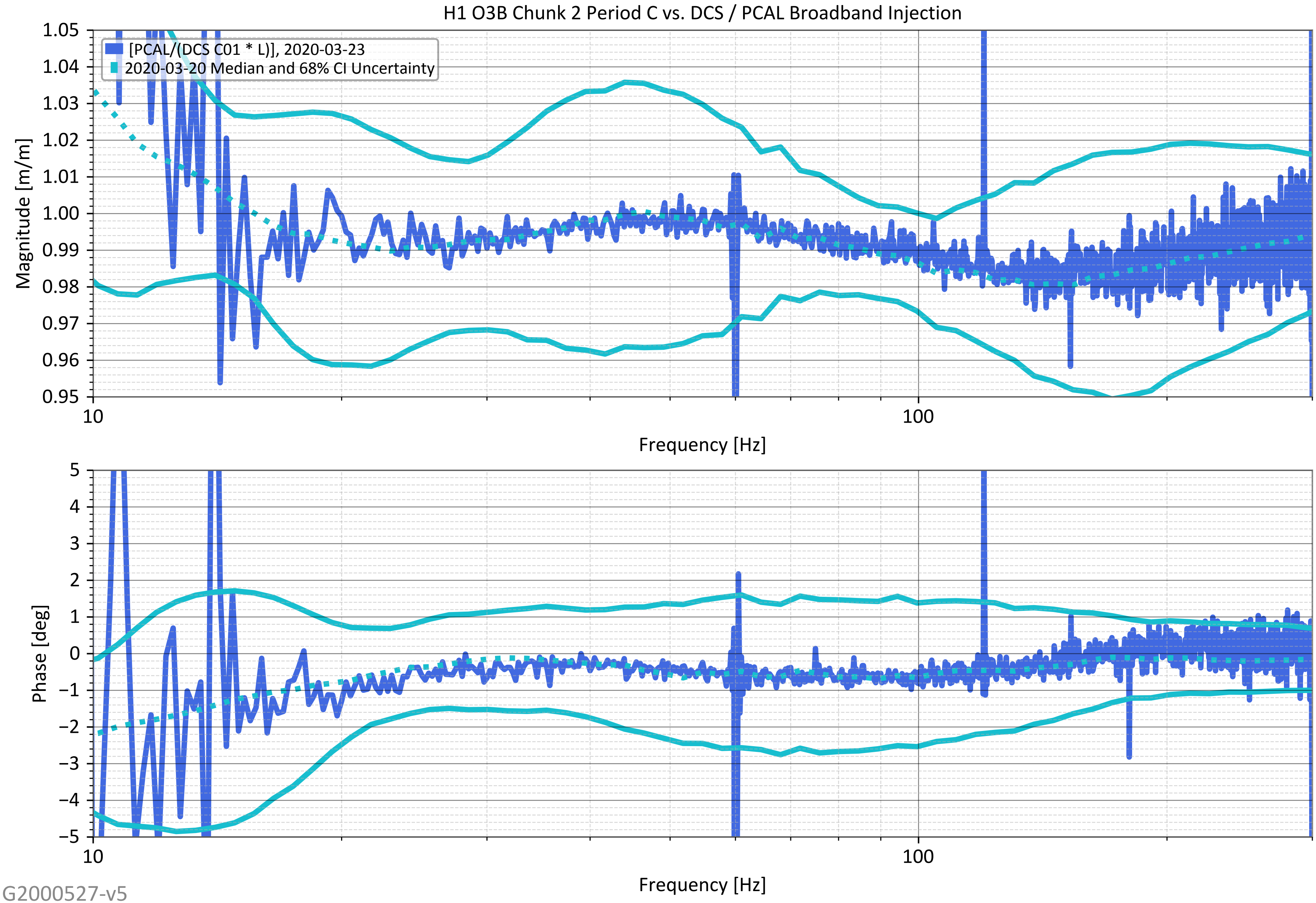
`/ligo/svncommon/CalSVN/aligocalibration/trunk/Runs/O3/H1/Results/Uncertainty/`

`2020-08-25_H1O3BChunk2Periodc_ResponseFunctionSysError_OMCWC_and_BadTDCFs_eta_R_omcwc_badtdcfs.hdf5`

- RRNom was modified in rev 11150

II.8 Final Answer

LHO aLOG 56582 details this, the 2020-03-23 broadband injection against the official, RRNom produced budget



Outline

Two Parts, each quite long. *sigh*

PART I: The ETMX UIM Driver, from Nov 27 to Dec 03 2019

PART II: The OMC Whitening Chassis, from Mar 16 to 27 2020

1. What's the issue?
2. Game Plan / Review of the Original Data
3. Review of the Circuit
4. Review of the Existing Compensation Scheme
5. Fitting and Predicting the Response Function Systematic Error
6. Comparison against measured Response Function Error
7. Collateral Damage on TDCFs
8. Final Answer
9. Conclusions

II.8 Conclusions

- Lessons learned:
  - We really need to take the time to get to know all of the circuits that matter, and understand how to best measure them
  - Yes, we can predict the frequency-dependent change caused by electronics errors in the sensing function.
  - We should take a new measurement of the OMC Whitening Chassis,
    - be sure to get data down to at least 0.25 Hz (and if patience allows 0.1 Hz), and update the compensation filters.
    - be sure to take both cascading response and single filter response measurements.
- We've found some insidious collateral damage of the OMC whitening chassis configuration change itself
  - the sensing function TDCFs were impacted by this, incorrectly reporting plant change, and cause *much worse* systematic error
  - Discovered and reviewed in G2001293

- Final answer (where we include OMC error itself *and* the correction for bad TDCFs) agrees nicely with 2020-03-23 measurement.

FIN