LIGO SURF 2023 - Final Report

Caden Swain

Mentor: Louis Dartez

September 20, 2023

Demonstration of Bayesian Transfer Function Fitting Method – A Potential Tool for Estimating Interferometer Uncertainty

Abstract

The Response Function of the LIGO Interferometer is central to reconstructing the strain produced by incoming gravitational waves. A function of the interferometer's response to external stimuli, the Response Function is both analytically modeled and experimentally measured using excitations from the photon calibrator system at discrete frequencies. The uncertainty in each data point is propagated to the residual of the model and measurements, with both the uncertainty and residual being interpolated over a broadband frequency range. While valid, interpolation methods lack the accuracy to estimate measurement uncertainty at non-measured frequencies that fitting an analytical transfer function could provide. This project analyzes the results of fitting a series of transfer functions to the Response Function using Bayesian statistics as opposed to traditional transfer-function-fitting methods. We use data gathered from the OMC DCPD S2300004 whitening chassis at discrete frequency points, varying the Signal-to-Noise Ratio as a proxy for varying the uncertainty in the measurements, and compare the results of each method.

Contents

List of Figures

1 Introduction

This introduction provides critical background to this project as well as a summary of this project as a whole. We introduce the LIGO interferometer in addition to the Differential Arm control loop and discuss the application of Gaussian Process Regression to the Response Function. We also give a summary of this report, briefly discussing each section and the overall conclusion of the project.

1.1 Background

The most important data the LIGO interferometer is used to measure is the "strain" of external stimuli on the test masses in each arm of the interferometer, calculated using the equation

$$h = \frac{\Delta L_{\text{free}}}{L} = \frac{\Delta L_{\text{x}} - \Delta L_{\text{y}}}{L} \tag{1}$$

where h is the strain in question, ΔL_{free} is the total displacement of the arms in meters, L is the length of the arms (4 km), and ΔL_{x} and ΔL_{y} are the displacements of the x and y arms, respectively (see Figure 1 for a diagram of the LIGO Interferometer). Strain data is extremely important because it is the primary component in confirming binary collisions and other astrophysical inferences. As such, gathering accurate strain data is critical. In order to confirm the validity of the measured strain values, the interferometer employs a photon calibrator system (Pcal) in which radiation pressure from accessory lasers produces a directly observable strain on the interferometer. The physical Pcal displacement is calculated using equations relating the radiation pressure to the movement of the test masses and is confirmed using measurements of the total response of the interferometer.

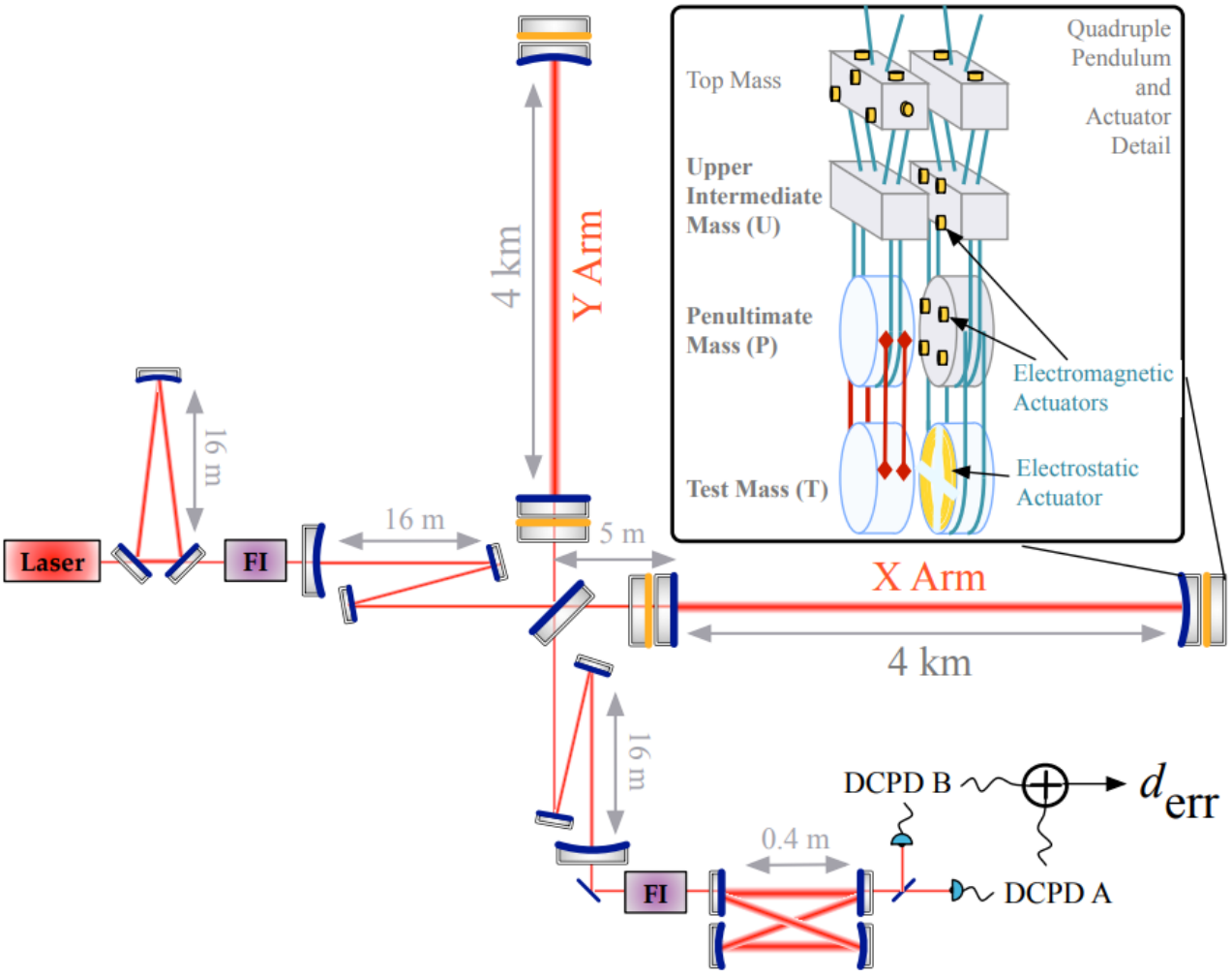

Figure 1: Conceptual diagram of the LIGO Interferometer. The pre-stabilized laser is sent through multiple resonant cavities before being split and sent into the two main Fabry-Perot cavities. The light then rejoins at the beamsplitter, interfering as a result of the oscillations of the end test masses, is sent through a final cavity, and is split and read into two photodiodes. For a more detailed description of the interferometer, see [1]

The interferometer is modelled by three different systems that compose the complete response of the interferometer: the Sensing Function (C), the Digital Filter (D), and the Actuation Function (A) (see Figure 2 for a model of the Differential Arm – DARM – Loop). Forming an open feedback loop, the Sensing Function records the residual displacement of the test masses, defined as $\Delta L_{\mathrm{res}} = \Delta L_{\mathrm{free}} - \Delta L_{\mathrm{ctrl}}$, where ΔL_{free} is the total displacement of the test masses and ΔL_{ctrl} is the control displacement produced by electromagnetic actuators to suppress ΔL_{free}. The Sensing Function then produces a digital error signal d_{err} which the Digital Filter subsequently converts to a digital

control signal d_{ctrl}. This Actuation Function then reads the control signal and produces an actuation displacement ΔL_{ctrl} to suppress ΔL_{free}. The composition of these three systems is represented by the Response Function, the entire interferometer's response to external stimuli. This Response Function is given by the equation

$$R = \frac{1 + ADC}{C} \tag{2}$$

where R is the Response Function model and A, D, and C are the Actuation Function, Digital Filter, and Sensing Function models, respectively. The strain incident on the interferometer is directly related to the Response Function through the relationship

$$h = \frac{R * d_{\text{err}}}{L} \tag{3}$$

where R is the Response Function model, d_{err} is the digital error signal produced by the Sensing Function, and L is the length of the interferometer's arms ($4\,km$). This equation displays the importance of forming an accurate model of the Response Function. For a detailed explanation of the systems composing the interferometer and the Response Function, see [1].

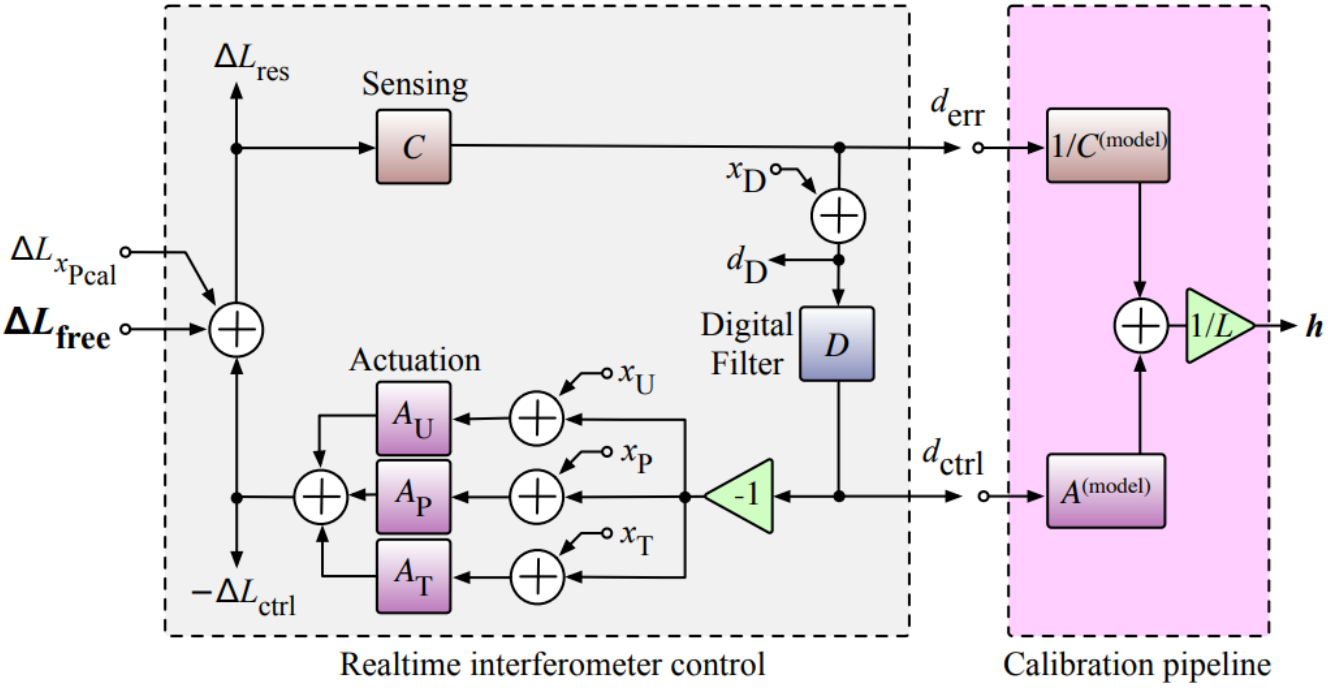

Figure 2: Conceptual diagram of the Differential Arm (DARM) Loop. For a detailed description of the DARM Loop, see [1]

The Response Function is measured using the Pcal system, inserting an excitation into the DARM loop, x_{Pcal}, that is propagated to DARM displacement, ΔL_{Pcal} and measuring the DARM loop error signal, d_{err}. By measuring the transfer functions of the Sensing and Actuation functions through injections of ΔL_{Pcal} at discrete frequencies, the Response Function measurements are reconstructed. A further explanation on the Sensing and Actuation Functions is given in [1]. Because these measurements are reconstructed at discrete frequencies, methods are used to interpolate the measured data over the frequency band. One very prominent interpolation method, Gaussian Process Regression (GPR), is currently utilized by the LIGO Scientific Collaboration (LSC) in data analysis. A statistical distribution of Response Function measurements constructed with GPR is used to calculate the systematic error in the Response Function, defined by $\frac{R^{(meas)}}{R^{(model)}}$. The median, 16th, and 84th percentiles of the systematic error distribution are then graphically displayed (see Figure 6). For more information on Gaussian Process Regression, see [2].

While viable, Gaussian Process Regression has shown a few important flaws, particularly, quoting Ethan Payne, the "mean model for Gaussian process cannot fit transfer function-like behavior," and the "measurements are not Gaussian distributed." These and more details concerning the flaws of GPR are discussed in his presentation, [3]. In this paper, we seek to examine an alternate data interpolation method in the attempt to account for the flaws in GPR and improve data analysis techniques used in the Calibration group. We test a python script denoted *BayesianTF* [4] that utilizes Bayesian statistics to create a distribution of transfer function fits directly to any set of transfer function measurement data, avoiding the many intermediate steps involved in GPR. We gather transfer function data from a spare OMC DCPD whitening chassis [5] and use both *BayesianTF* and the current standard transfer-function-fitting script *IIRRational* [6] to create fits for the data. By comparing the results of this new transfer-function-fitter to those of *IIRRational*, we roughly determine the accuracy of *BayesianTF* and present potential benefits the application of *BayesianTF* to the Response Function could produce.

1.2 Summary

In this project, we begin by gathering measurement data from a spare OMC DCPD whitening chassis. This data is both used to characterize the whitening chassis for active use in the interferometer as well as to test the transfer-function-fitters *IIRRational* and *BayesianTF*. We discuss the different strategies for gathering both transfer function and noise data from the whitening chassis, including a brief discussion on the measurement uncertainty calculations generated by unique scripts written for this project. We also discuss scripts written to confirm the accuracy of the gathered noise data.

We then analyze each transfer-function-fitter, with a brief description of the code and functional differences between the two. We also display and examine Bode plots of the resulting transfer function fits from each fitting method. We describe the potential benefits and pitfalls each fitting method displays based on these Bode plots.

Finally, we end this paper with a discussion of the final results of the project. In particular, we conclude that *BayesianTF* is an accurate transfer-function-fitter that, due to its consideration of measurement uncertainty, generation of error-bars, and overall excellence with high measurement uncertainty data sets, is a valid candidate for application to the interferometer's Response Function. We also find that the rough limit for *BayesianTF's* application is data sets with an SNR of $\sim 10^3$ and that the rough limit for *IIRRational's* application is similarly between SNR $\sim 10^4$ and $\sim 10^3$. Given that the limit points of each fitting method are not precisely defined, further research and experimentation with *BayesianTF* and *IIRRational* on data with Signal-to-Noise Ratios between 10^3 and 10^4 should be conducted. The results of this project may also be utilized in further research by the LIGO Calibration team on *BayesianTF's* application to the Response Function.

2 Methodology

This section details the techniques utilized in this project. We begin with a detailed explanation on the collection of data used to test the transfer-function-fitters *IIRRational* and *BayesianTF*. This subsection includes details concerning the physical collection of data and scripts used to validate the data. We then explain the data analysis techniques used to examine the results of each method. This subsection includes breakdowns of each method, including the functions, differences from the other, and the output when given input data. This section then discusses final results of this project, centering around the effectiveness of *BayesianTF* and its potential application to the Response Function. Finally, this section concludes with the future endeavors that should be conducted following this project.

2.1 Data Collection

Measurement data is gathered from a spare OMC DCPD S2300004 whitening chassis and is used to test the effectiveness of *BayesianTF* as compared to *IIRRational*. The whitening chassis is designed as a 1:10 filter using the zeros-poles-gain transfer function model. As such, the theoretical zeros, poles, and subsequent Bode plot displaying the transfer function are known *a priori*, allowing the effectiveness of each transfer-function-fitter to be accurately analyzed. Transfer function data is gathered from the whitening chassis using swept-sine measurements (see subsubsection 2.1.1) at varying input voltages, the data set with the highest input voltage being the most accurate with the highest Signal-to-Noise Ratio. Noise data from the whitening chassis is also collected using Fast-Fourier Transforms (see subsubsection 2.1.2) and used both to calculate the SNR of each measured data point as well as calculating the measurement uncertainty contained in each data set (see appendix A). In this subsection we describe the intricacies of gathering both the transfer function and noise data from the whitening chassis.

2.1.1 Transfer Function Data

The transfer function data was gathered from the whitening chassis using the SR785 Dynamic Signal Analyzer and its Accessory Suite (displayed in Figure 7). The Accessory Suite serves to change the singular ouptut of the SR785 into a differential, alternating current output. This differential output is sent through the whitening chassis and fed back into the SR785, where the Bode plot of the whitening chassis's transfer function is displayed. The whitening chassis is set up to be a device under test using the Chassis Schematic Print (displayed in Figure 8). For instance, when testing channel A of the whitening chassis, coaxial cables would connect the differential output of the Accessory Suite to the A+ and A- pins on the Preamp Chamber Interface on the Front Panel of the whitening chassis. Coaxial cables would also connect the subsequent differential output of the whitening chassis to the input of the SR785. For

a picture of the whitening chassis as a device under test, see Figure 9. The SR785 uses a swept-sine measurement to send a series of input signals at a constant amplitude but varying frequencies. For this experiment in particular, the data was collected at 200 frequency points ranging from 0.1 Hz to 102.4 kHz. The resulting data contains the shifts in the magnitude and phase of the signal that are caused by the whitening chassis. This transfer function data was collected for both Whitening ON and Whitening OFF configurations of each Channel of the whitening chassis at varying input voltages: 1 Vpk, 0.5 Vpk, 0.1 Vpk, 10 mVpk, 1 mVpk, and 0.1 mVpk. The reasoning behind collecting the data at varying input voltages is explained in subsubsection 2.1.2 when discussing the Signal-to-Noise Ratio.

2.1.2 Noise Data

The noise data of the whitening chassis is also gathered with the SR785 but using a Fast-Fourier Transform (FFT) as opposed to a swept-sine measurement. Using a similar setup for the whitening chassis, the input pins are short-circuited to each other rather than being connected to the SR785 to reduce the effect of floating input voltages. The noise floor of the SR785 is then gathered by taking FFT measurements with the whitening chassis turned completely off. Similarly, the noise floors of the Whitening ON and Whitening OFF configurations of both Channels of the chassis are gathered by taking FFT measurements with the chassis turned on. These Fast Fourier Transform measurements were taken at varying intervals: 0.5 Hz to 400 Hz, 4 Hz to 3.2 kHz, and 32 Hz to 25.6 kHz. This detail becomes critical in analyzing the transfer-function-fitting methods, as due to each FFT being taken at differing frequency intervals, the total amount of noise will differ depending on which FFT interval the data was taken within. Importantly, this means that data points at high frequency values with contain more noise data than those points at low frequency values. In order to confirm the noise data was gathered correctly, we created plots overlaying the measured noise data and noise data from the S2300003 whitening chassis previously gathered by Jeff Kissel [7]. These plots, as well as full details containing the complete characterization of the OMC DCPD S2300004 whitening chassis may be found in [8].

The noise data is critical to the application of the transfer function data to *BayesianTF*, as *BayesianTF*, relying on Bayesian statistics, requires measurement uncertainty in the data in order to function. The noise data allows for the Signal-to-Noise Ratio, calculated with $\frac{V_{(\text{signal})}}{V_{(\text{noise})}}$, and subsequent measurement uncertainty to be gathered. Notice that by varying the input signal voltage, the SNR may be modified as needed, as the noise voltage remains roughly static. This is critical to testing *BayesianTF*, as utilizing data sets with varying SNR's allows for an accurate assessment of *BayesianTF's* ideal SNR range and possibly even the SNR at which *BayesianTF* begins to fail. In order to calculate the Signal-to-Noise Ratio (SNR) of each data point, more FFT measurements were taken to span the 0.1 Hz to 102.4 kHz frequency band at which the transfer function data was gathered. The data sets for Channel A Whitening

ON were then appended together, and the data points for the set with the smaller range were utilized in the case of overlapping data points. The noise data was then interpolated using Python, and noise data corresponding to the particular frequency points at which the transfer function data was gathered was retrieved.

In order to calculate the uncertainty present in the measurements, equations relating the Signal-to-Noise Ratio to the coherence between the input and output signals and relating the subsequent coherence to measurement uncertainty are utilized. For a detailed breakdown of the equations used to calculate measurement uncertainty, see appendix A. We wrote Python scripts to use the interpolated noise data to calculate the measurement uncertainty for each data point and create a new file containing transfer function data with the frequency, magnitude, phase, and newly acquired measurement uncertainty. In particular, one function calculated the SNR, coherence, and measurement uncertainty of each data point in a particular data set (*Uncertainty Measurements.py*), and the other script called the function and appended the data to a new file (*Transfer Function Data with Uncertainty.py*). Both of these files may be found in a GitLab repository [9].

2.2 Data Analysis

In this subsection, we give a short description of the design and function of each transfer-function-fitting method in addition to present the results of each method. We particularly use data gathered for Channel A "Whitening ON," as Channel A and B are interchangeable and the Whitening ON configuration provides the necessary amount of data to analyze the methods. The 1 V input data set, along with the 1 V *IIRRational* transfer function fit, is considered the standard to which the transfer function fits to all other data sets will be compared. This specific data set and fit is the standard due to the extremely high SNR and subsequent nonexistent measurement uncertainty at which it was gathered. Each script produces an array of complex points modelling the transfer function of the whitening chassis. As such, the residuals between the standard 1 V input transfer function fit produced by *IIRRational* and all the transfer function fits of other sets of data may be constructed and analyzed.

2.2.1 *IIRRational*

The *IIRRational* code is the current standard transfer-function-fitter used in the LSC. As such, the *IIRRational* results from the data set with the lowest measurement uncertainty are extremely accurate with a residual of approximately unity. *IIRRational* itself fits a polynomial composed of zeros in the numerator and poles in the denominator to measured transfer function data. Using data analysis techniques for control systems, the transfer-function-fitter are able to fit high-order transfer function polynomials. In addition, *IIRRational* does take the SNR into account in order

to calculate residuals between the measurements and noise data. The code excels in its adaptability and versatility, being known for its accuracy with high SNR data. For presentations detailing the functionality of *IIRRational* in more detail, see [10], [11], and [12].

We used a modified version of *IIRRational* to generate transfer function fits to each of the data sets. The resulting fits for the Channel A Whitening ON 1 V, 0.1 mV, 0.05 mV, and 0.01 mV data sets are shown in Figures 10 – 13. The left plots in the figures are Bode plots displaying the transfer function data and resulting *IIRRational* transfer function fit, with the upper-left plot displaying the Magnitude and the lower-left plot displaying the Phase. The right plots in the figures are also Bode plots displaying the residuals between the data and resulting fit. The increasing chaos and deviation from unity in the residuals as the input voltage decreases shows the difficulties present in fitting a transfer function to high measurement uncertainty data without already knowing the theoretical transfer function. In this case, the transfer function of the whitening chassis is known *a priori*; however, the zeros-poles-gain model of the Response Function is not known *a priori*, so the fit generated by *IIRRational* in such situations cannot be considered definitive. The inaccuracy of *IIRRational* in these high measurement uncertainty scenarios is exemplified by Figure 14, in which the residuals between the 0.1 mV, 0.05 mV, and 0.01 mV fits and the standard 1 V fit are displayed. The more the residual between a specific fit and the standard strays from unity, more inaccurate that specific fit is. Note that the tail end of the residuals stray from unity to such an extent due to the increased amount of noise data in the high frequency range due to the construction of the FFT measurements used to gather the noise data. As shown, the residual for the 0.01 mV fit, the most important fit of the group due to its extremely low SNR/high measurement uncertainty, strays significantly from unity. This residual in particular implies that *IIRRational* is not reliable in generating a definitive, accurate transfer function fit for data with high measurement uncertainty.

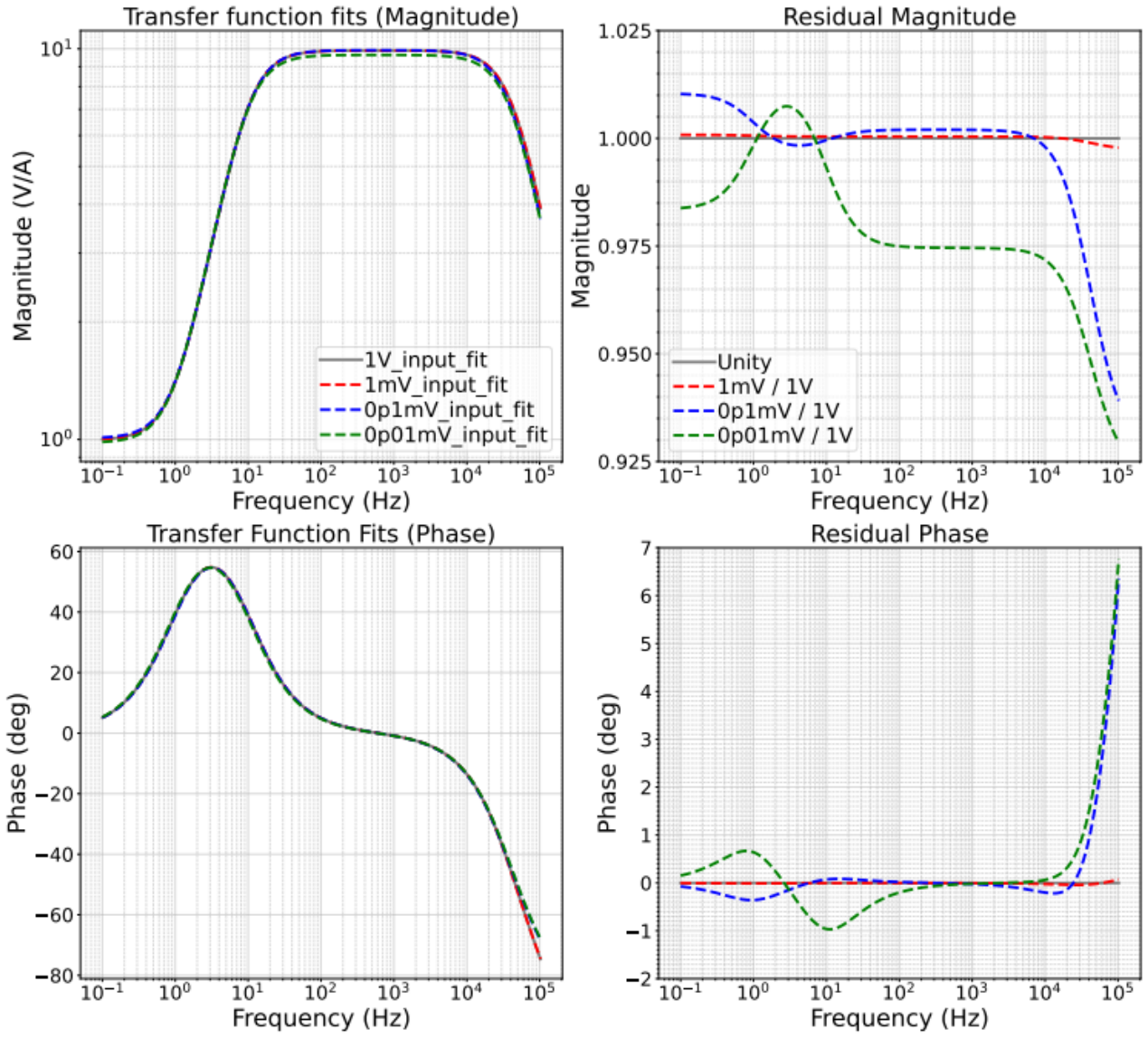

Figure 3: Overlaid residuals between the 0.1 mV, 0.05 mV, 0.01 mV and 1 V *IIRRational* transfer function fits. The top plots show Magnitude. The bottom plots show Phase. Left plots show each transfer function fit generated by *IIRRational*. Right plots show the residuals between each fit and the standard 1 V fit.

2.2.2 *BayesianTF*

The *BayesianTF* code utilizes Bayesian statistics to create a posterior distribution of transfer function curves characterizing any system that may be modelled by a transfer function. Utilizing a uniform distribution of parameters as priors, *BayesianTf* searches through the parameter space to find the parameter values that best match the measurement data. With a predetermined Gaussian likelihood function and the massive parameter space, *BayesianTF* constructs the posterior distribution of transfer function curves from the peaks of the distributions of each parameter following the inclusion of the likelihood of the data given the parameters. In addition, *BayesianTF* uses a residue-pole transfer function mode as opposed to the traditional zeros-poles-gain model used by *IIRRational* in order to enhance the speed of the code by reducing the complexity in the transfer function form. A complete explanation of the residue-pole transfer function model is found in [13]. *BayesianTF* takes the number of poles as a user input, fitting a transfer function to the user's specified number of poles. More analytical poles in the transfer function model inevitably results in a more accurate fit to the data at the cost of increased computing power. As such, using *BayesianTF* multiple times with an increasing number of poles on a specific set of data results in a distribution of increasingly complex distributions of transfer function fits. The evidence of each fit distribution may be compared to the evidences of the other fit distributions, allowing for the best fit distribution to be determined.

The same sets of data used to generate fits with *IIRRational* were then used to generate fits with *BayesianTF*. Because *BayesianTF* is designed specifically for data with high measurement uncertainties, the 0.01 mV and 0.1 mV data sets were specifically selected for testing. For the 0.01 mV data set (average SNR of 113), *BayesianTF* generated fits for transfer functions with 2, 3, and 4 poles with the poles-residue model. For the 0.1 mV data set (average SNR of 1131), *BayesianTF* generated fits for transfer functions with 2 and 4 poles with the pole-residue model. Bode plots showing the *BayesianTF* and *IIRRational* transfer function fits for each data set overlaid with the measurement data are displayed in Figures 15 – 18. All of the *BayesianTF* fits for the 0.01 mV data seem very accurate to the eye, but further examination of plots of the residuals between the *BayesianTF* fits and standard 1 V *IIRRational* fit (displayed in Figures 5 and 19 – 20) reveal that the most accurate *BayesianTF* fit is the 3 pole transfer function model fit (displayed in Figure 4). This result is consistent with the designed whitening chassis transfer function. Plots of the residuals for the 0.1 mV data set are displayed in Figures 21 – 22 and further examined in section 2.3 along with the plots of the fits themselves.

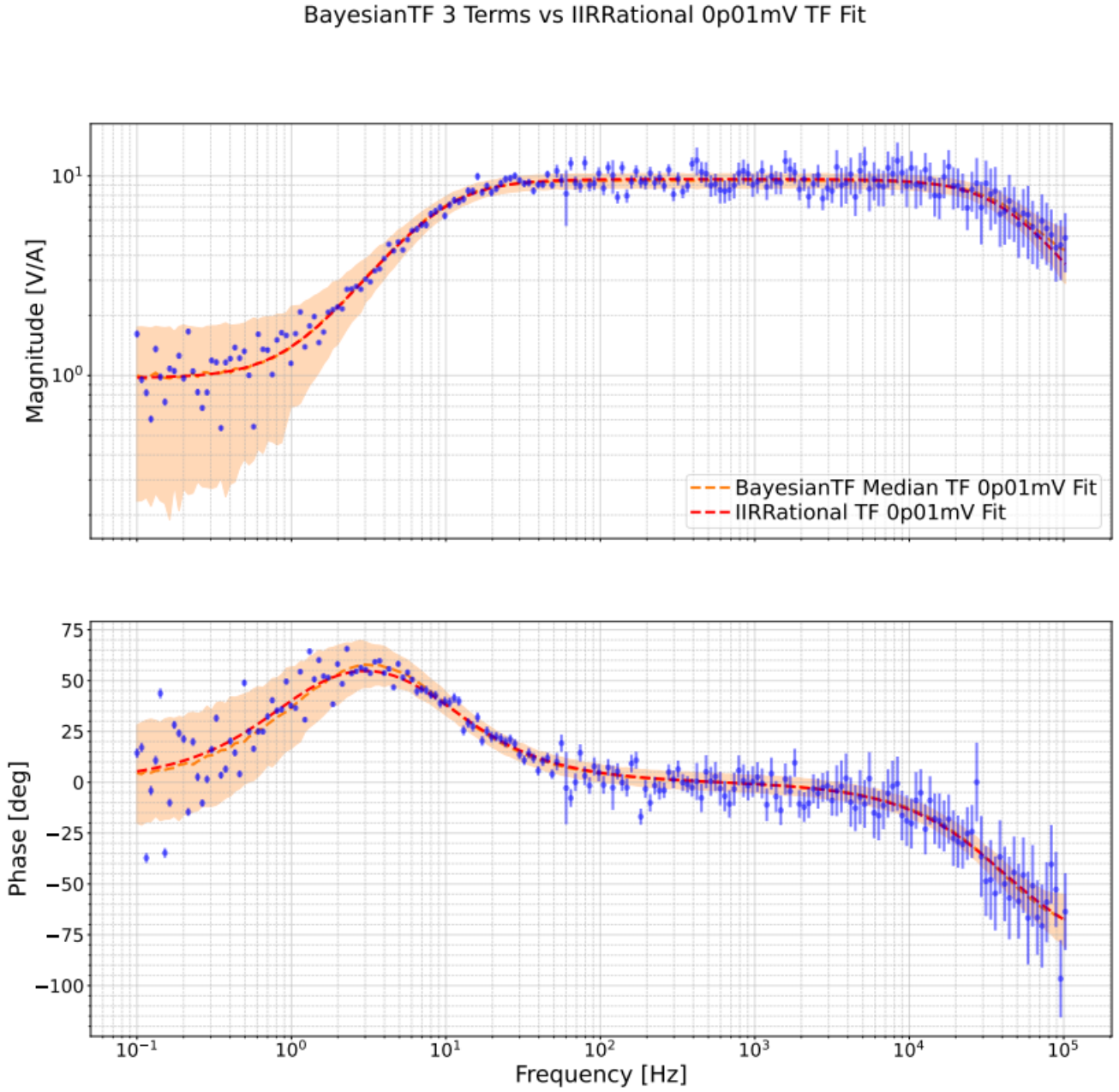

Figure 4: Bode plot showing the comparison between the 3-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. Error bars for the *IIRRational* fit are constructed with the 5th and 95th percentile fits from the posterior distribution.

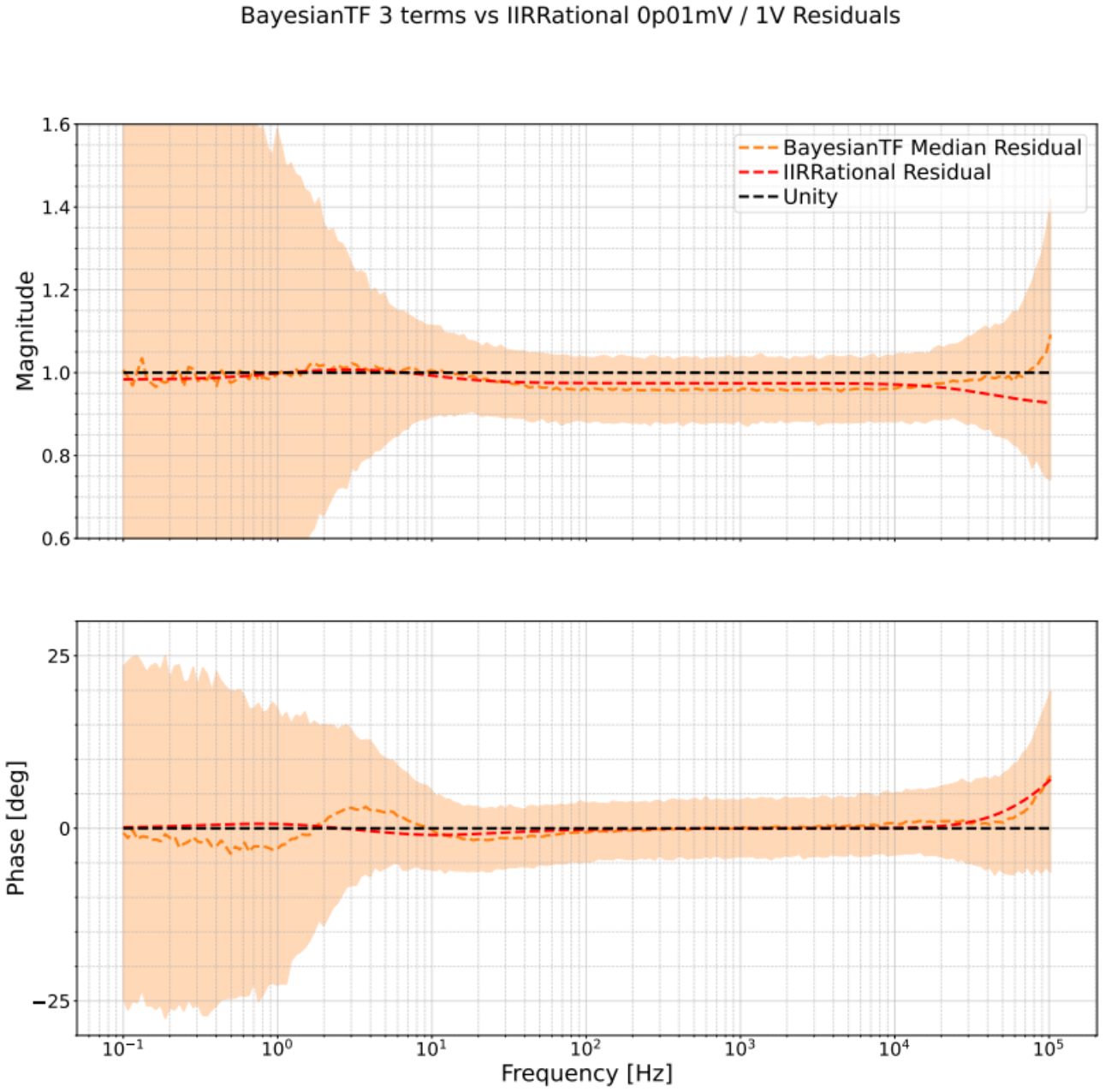

Figure 5: Bode plot showing the residuals between the 3-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set with the standard 1 V *IIRRational* fit. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. The error bars for the *BayesianTF* residual are constructed with the 5th and 95th percentile residuals from the posterior distribution of residuals.

2.3 Results

Finally, *BayesianTF* has enormous potential to be a valid replacement to Gaussian Process Regression in constructing a fit for the interferometer's Response Function. The bode plots overlaying the best fit distribution from

BayesianTF and the fit from *IIRRational* for each data set shows the flexibility of the *BayesianTF* fits at extremely low SNR's. Specifically viewing the 0.01 mV input voltage data set, the residual between the median *BayesianTF* fit and the standard 1 V *IIRRational* fit deviates from unity slightly more than the 0.01 mV *IIRRational* fit in certainty frequency ranges. However, the 90th percentile error bars always spans unity, meaning that the fit closest to unity, the best fit, is contained in the *BayesianTF* fit distribution. In contrast, the single *IIRRational* fit is not close enough to unity to justify the lack of measurement uncertainty in the fit. As such, *BayesianTF* is a significantly better transfer-function-fitter than *IIRRational* in low SNR/high measurement uncertainty scenarios. In addition, *BayesianTF's* implementation of error bars is extremely useful for its potential application to the Response Function. Because the Response Function data is often gathered at an SNR less than or equal to 10, the measurement uncertainty is high enough that error bars are essentially critical to the accurate estimation of the transfer function. *BayesianTF's* usefulness is refined even further by the accuracy of its results in this test using a filter designed with a specific, known transfer function.

The other noteworthy result is the evaluation of the points at which each transfer-function-fitter begins to fail. As demonstrated with Figure 14, *IIRRational* begins to fail significantly at some point between SNR $\sim 10^3$ and SNR $\sim 10^2$. Similarly, *BayesianTF* also begins to fail at an SNR of $\sim 10^3$, as shown by its inability to accurately fit the 0.1 mV data set. The 2-pole fit shown in Figure 17 is obviously completely incorrect, and the 4-pole fit shown in Figure 18, while rather accurate, is more inaccurate than any of the 0.01 mV fits and reaches the peak of a reasonable computing time. For these reasons, *BayesianTF* should not be applied to data sets with an average SNR of above $\sim 10^3$; *BayesianTF* should ideally be applied to data sets with an average SNR of between $\sim 10^2$ and ~ 1. This limit on *BayesianTF* is a result of the design of the code, as *BayesianTF* functions by searching for a peak in the massive, high-dimensional parameter space to fit the data. In high SNR scenarios, the region of the parameter space corresponding to the posterior becomes to narrow to be sufficiently found, and *BayesianTF* fails. While this is a pitfall of *BayesianTF*, its effectiveness with high measurement uncertainty data sets is critical to its potential application to the Response Function. Because the Response Function data is usually gathered at an SNR of between ~ 10 and ~ 1, the application of *BayesianTF* to better estimate the uncertainty and systematic error present in the Response Function might not only be valid but an overall more accurate method than Gaussian Process Regression.

In conclusion, *BayesianTF* is an accurate transfer-function-fitter in low SNR/high measurement uncertainty scenarios as shown in the comparison with the currently accepted transfer-function-fitter, *IIRRational*. The benefit of including error-bars also makes *BayesianTF* a valid candidate for application to the interferometer Response Function, which is critical to confirming accurate strain data used in astrophysical inference.

2.4 Future Goals

This project may be furthered in quite a few places. For instance, we could only gather the rough SNR limit on *BayesianTF* and *IIRRational*. Further experimentation and analysis using data sets between an SNR of $\sim 10^4$ and $\sim 10^3$ should be conducted to determine the exact point at which each transfer-function-fitting method breaks down. In addition, an analytical determination of *BayesianTF's* effectiveness, such as a reduced chi-squared statistic, could be applied to more accurately compare *BayesianTF* to *IIRRational*. Importantly, *BayesianTF* should be compared with Gaussian Process Regression using the same data sets as in this project in order to determine whether *BayesianTF* could be the best application to estimate the uncertainty and systematic error in the Response Function. Finally, general analysis and updates to the *BayesianTF* method itself should be conducted in the hopes of being utilized for the Response Function.

3 Acknowledgements

Thank you to everyone at the LIGO Hanford site for a warm and supportive work environment. Special thanks to the SURF mentors at the Hanford site – Louis Dartez, Ansel Neunzert, Rick Savage, Camilla Compton, and Michael Landry. Special thanks also to my mentor, Louis Dartez, Ethan Payne, Marc Pirello, Jeff Kissel, Rick Savage, and Ansel Neunzert for personal assistance with this project. Finally, thank you to the LIGO Scientific Collaboration and the National Science Foundation for this opportunity.

References

[1] L. Sun et al. (LIGO Scientific Collaboration), (2021).

[2] J. Wang, *An intuitive tutorial to gaussian processes regression*, https://doi.org/10.48550/arXiv.2009.10862, 2022.

[3] E. Payne, *Bayesian transfer function fitting for gravitational-wave detector calibration*, https://dcc.ligo.org/LIGO-G2300833-v1, version G2300833-v1, 2023.

[4] E. Payne, *Bayesiantf*, https://git.ligo.org/ethan.payne/bayesiantf/-/tree/experimental?ref_type=heads, 2023.

[5] D. Schaetzl, *Omc dcpd whitening filter chassis assembly (2x channel, 1x omc version)*, https://dcc.ligo.org/LIGO-S2300004, version S2300004-v3, 2023.

[6] L. McCuller, *Wield-iirrational*, https://github.com/wieldphysics/wield-iirrational, 2023.

[7] D. Schaetzl, *Omc dcpd whitening filter chassis assembly (2x channel, 1x omc version)*, https://dcc.ligo.org/LIGO-S2300003, version S2300003-v5, 2023.

[8] C. Swain, *Characterization of the spare omc dcpd whitening chassis - s2300004*, https://alog.ligo-wa.caltech.edu/aLOG/index.php, 2023.

[9] C. Swain, *Data uncertainty*, https://git.ligo.org/caden.swain/data-uncertainty, 2023.

[10] L. McCuller, *Overview and demos of iirrational for system identification*, https://dcc.ligo.org/LIGO-G2000855, version G2000855-v2, 2020.

[11] L. McCuller, *A fitting explanation of iirrational*, https://dcc.ligo.org/LIGO-G2000636, version G2000636-v1, 2020.

[12] L. McCuller, *Iirrational fitter exploration (lvc talk)*, https://dcc.ligo.org/LIGO-G1800571, version G1800571-v1, 2018.

[13] B. Gustavsen and A. Semlyen, IEEE Transactions on Power Delivery **14**, https://doi.org/10.1109/61.772353 (1999).

[14] D. Schaetzl, *Omc dcpd whitening filter chassis assembly (2x channel, 1x omc version)*, https://dcc.ligo.org/LIGO-D2200215, version D2200215-v1, 2023.

A Measurement Uncertainty Calculations

In order to calculate the measurement uncertainty in the transfer function data gathered using the SR785, we had to calculate the Signal-to-Noise Ratio (SNR) of each measurement in each data set. The relevant equations used for this process are:

$$SNR = \frac{P_{\text{signal}}(f)}{P_{\text{noise}}(f)} \tag{4}$$

$$= \frac{V_{\text{signal}(f)}}{V_{\text{noise}}(f)} \tag{5}$$

where P_{signal} and V_{in} are the power and voltage, respectively, of the input signal and P_{noise} and V_{noise} are the power and voltage, respectively, of the noise signal. The SNR is then used to calculate the coherence between the input and output signals with $SNR = \frac{C(f)}{1-C(f)}$, as the input signal is completely coherent with itself and the noise is completely incoherent with the input signal. From here, the coherence may be derived and used to further derive the uncertainty in each measurement with

$$C(f) = \frac{SNR}{1 + SNR} \tag{6}$$

$$\sigma(f) = \sqrt{\frac{1 - C^2(f)}{2N_{\text{avg}}C^2(f)}} \tag{7}$$

where $C(f)$ is the coherence between the input and output signals, $\sigma(f)$ is the measurement uncertainty of each data point, and N_{avg} is the number of data points averaged per discrete frequency [1].

B Figures

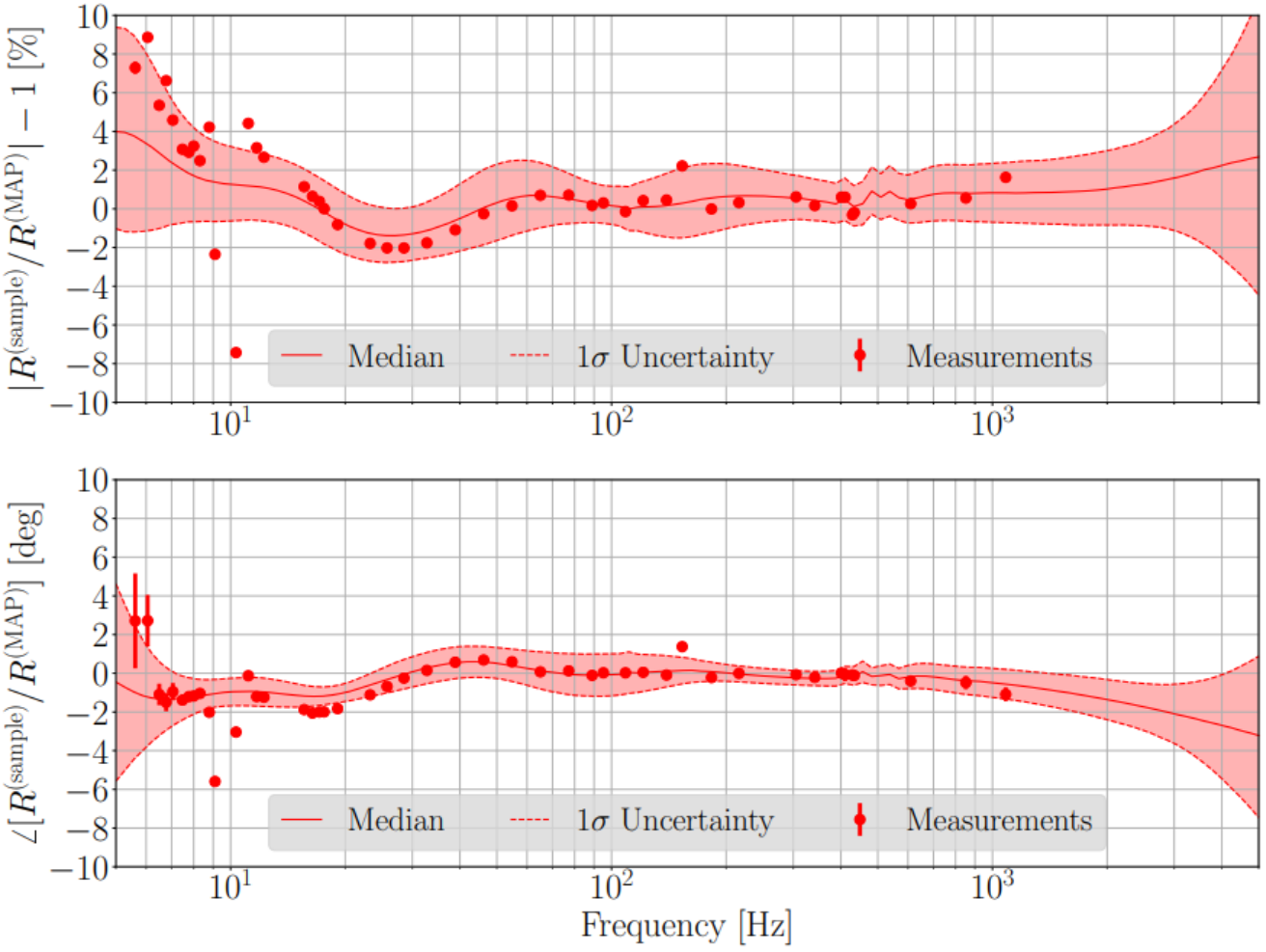

Figure 6: Bode plot displaying the systematic error in the Response Function. The top plot shows Magnitude and the bottom plot shows Phase. Each of the red dots are data points from a swept sine measurement on the Response Function. The solid and dashed lines represent the median, 16th, and 84th percentiles, respectively, of the distribution of transfer functions created with Gaussian Process Regression. For a detailed description of this plot and the use of GPR in its creation, see [1]

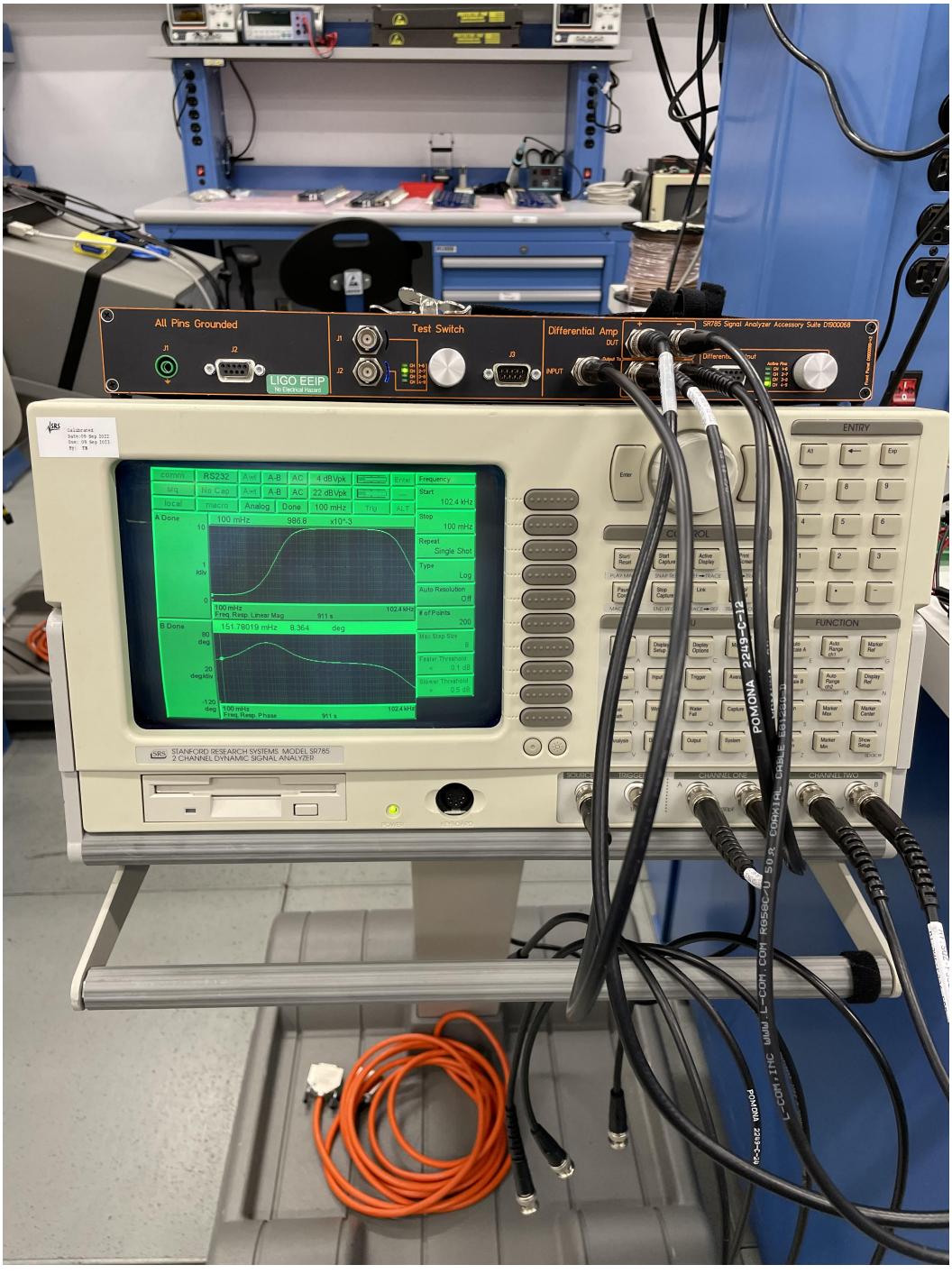

Figure 7: Picture of SR785 Dynamic Signal Analyzer (bottom) and Accessory Suite (top) during Data Collection

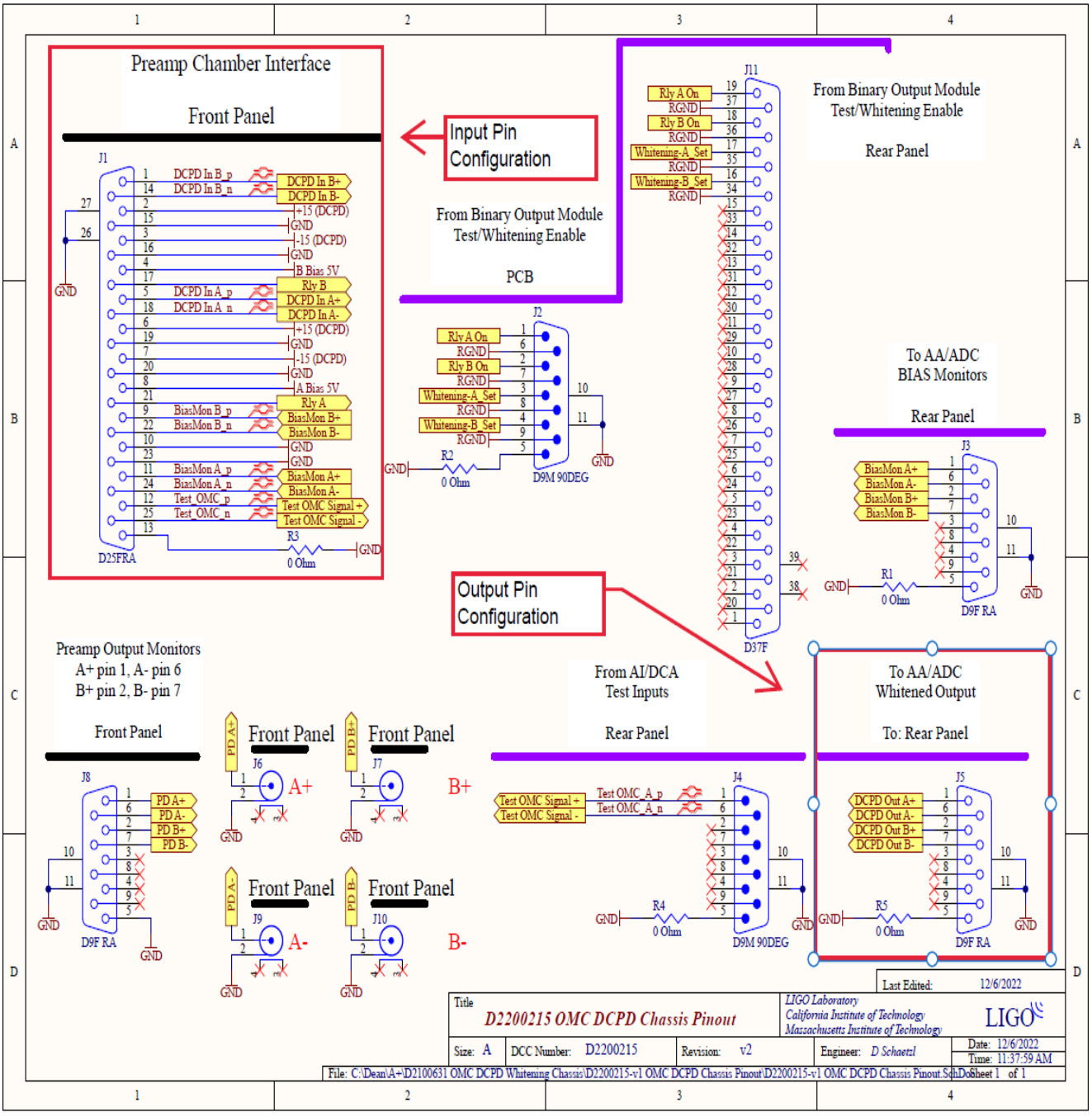

Figure 8: Annotated OMC DCPD Chassis Schematic Print. Focus particularly on the Preamp Chamber Interface Front Panel and To AA/ADC Whitening Output To: Rear Panel sections of the schematic. See [14] for the original whitening chassis schematic.

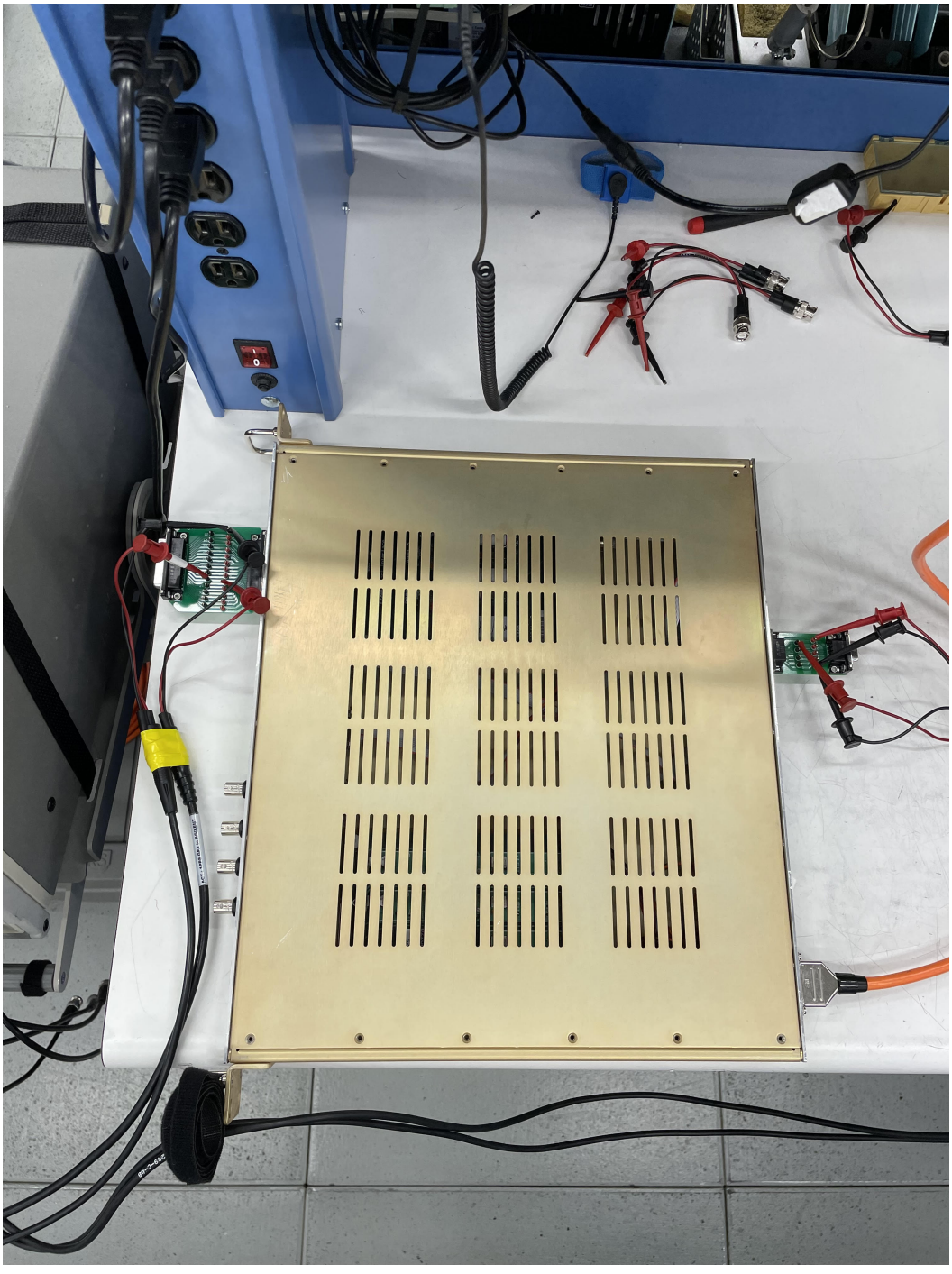

Figure 9: Picture of OMC DCPD S2300004 Whitening Chassis during Data Collection

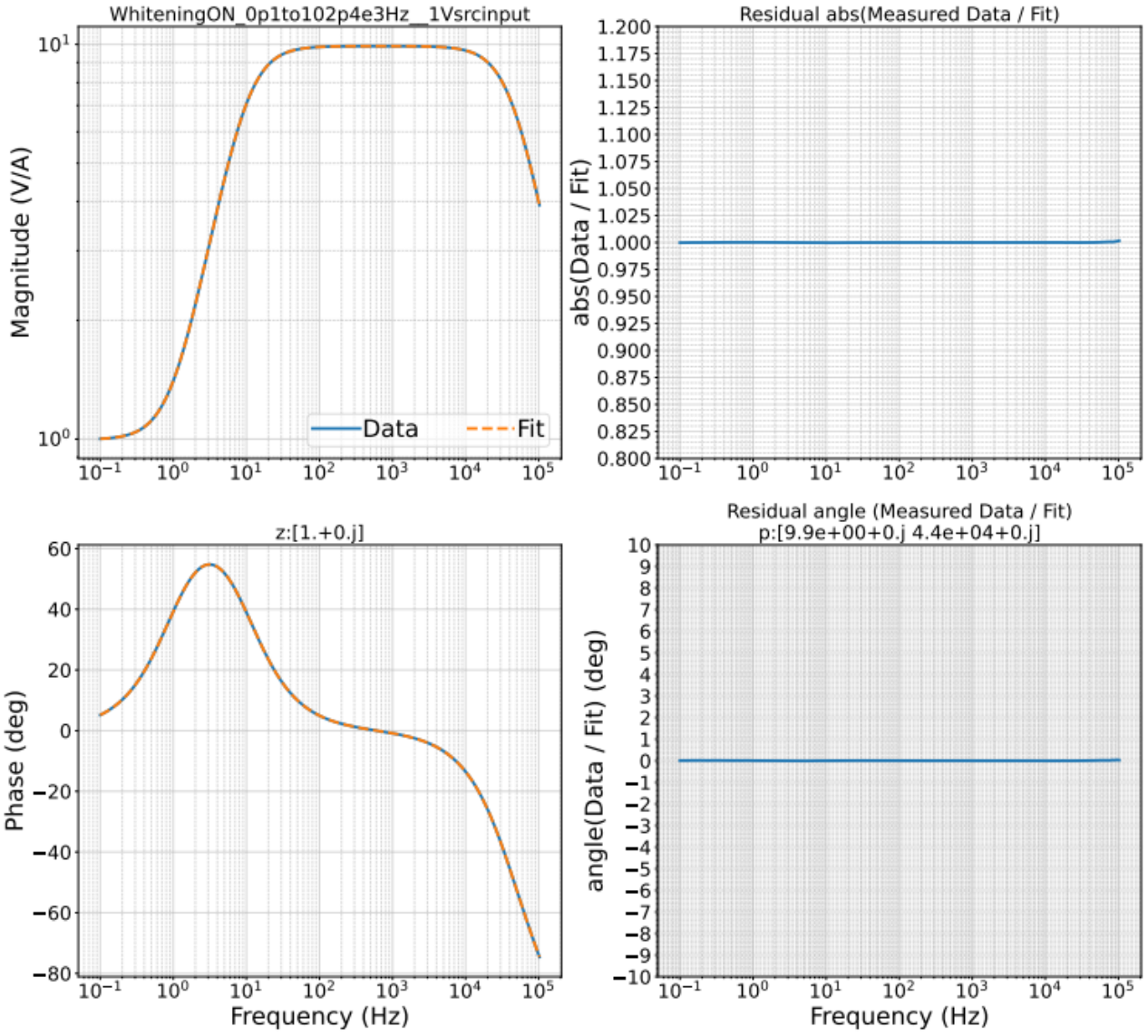

Figure 10: *IIRRational* transfer function fit and residual for the 1 V input data set. The top plots show Magnitude. The bottom plots show Phase. Left plots show the transfer function data overlaid with the generated fit. Right plots show the residual between the data and fit.

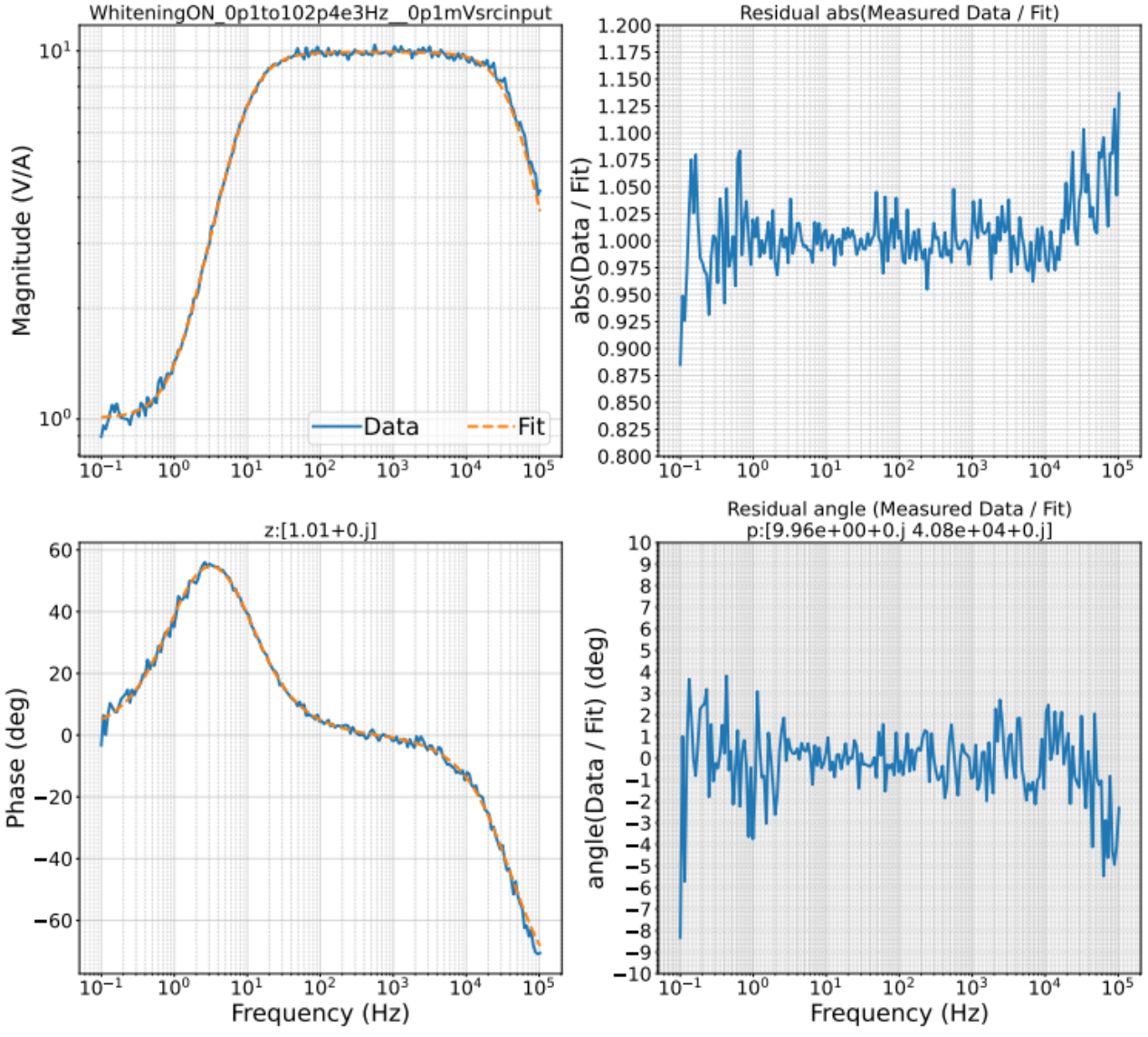

Figure 11: *IIRRational* transfer function fit and residual for the 0.1 mV input data set. The top plots show Magnitude. The bottom plots show Phase. Left plots show the transfer function data overlaid with the generated fit. Right plots show the residual between the data and fit.

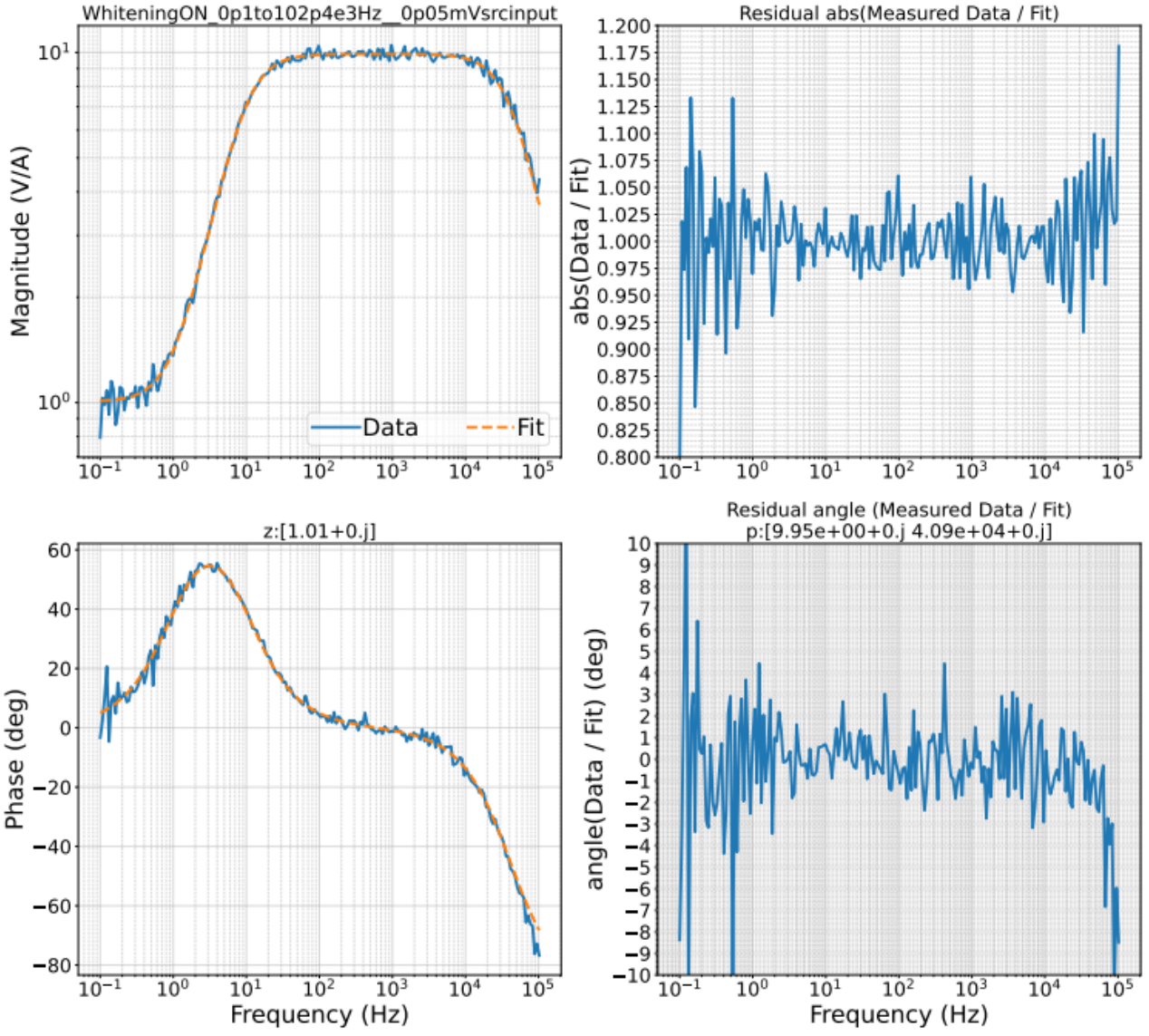

Figure 12: *IIRRational* transfer function fit and residual for the 0.05 mV input data set. The top plots show Magnitude. The bottom plots show Phase. Left plots show the transfer function data overlaid with the generated fit. Right plots show the residual between the data and fit.

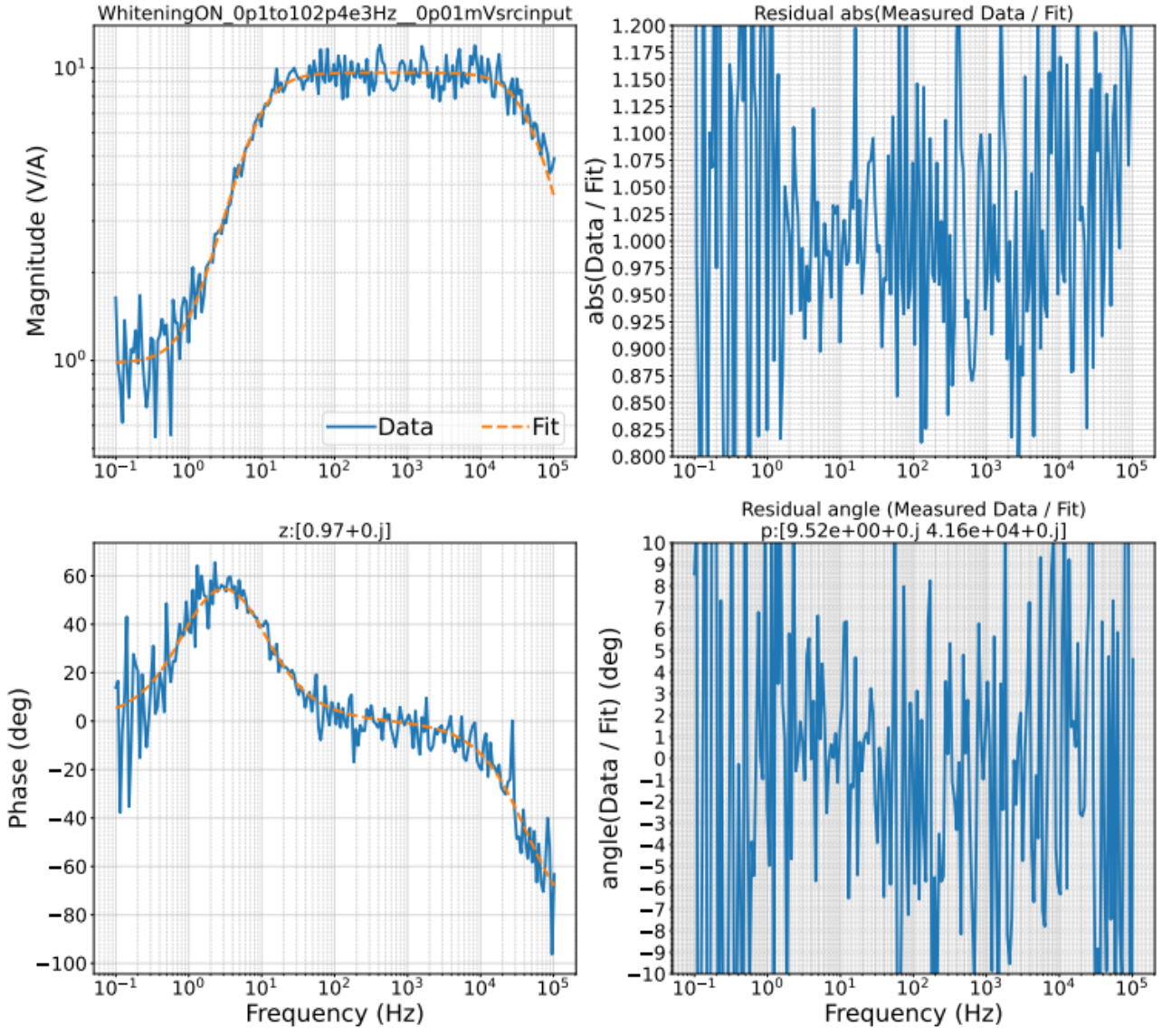

Figure 13: *IIRRational* transfer function fit and residual for the 0.01 mV input data set. The top plots show Magnitude. The bottom plots show Phase. Left plots show the transfer function data overlaid with the generated fit. Right plots show the residual between the data and fit.

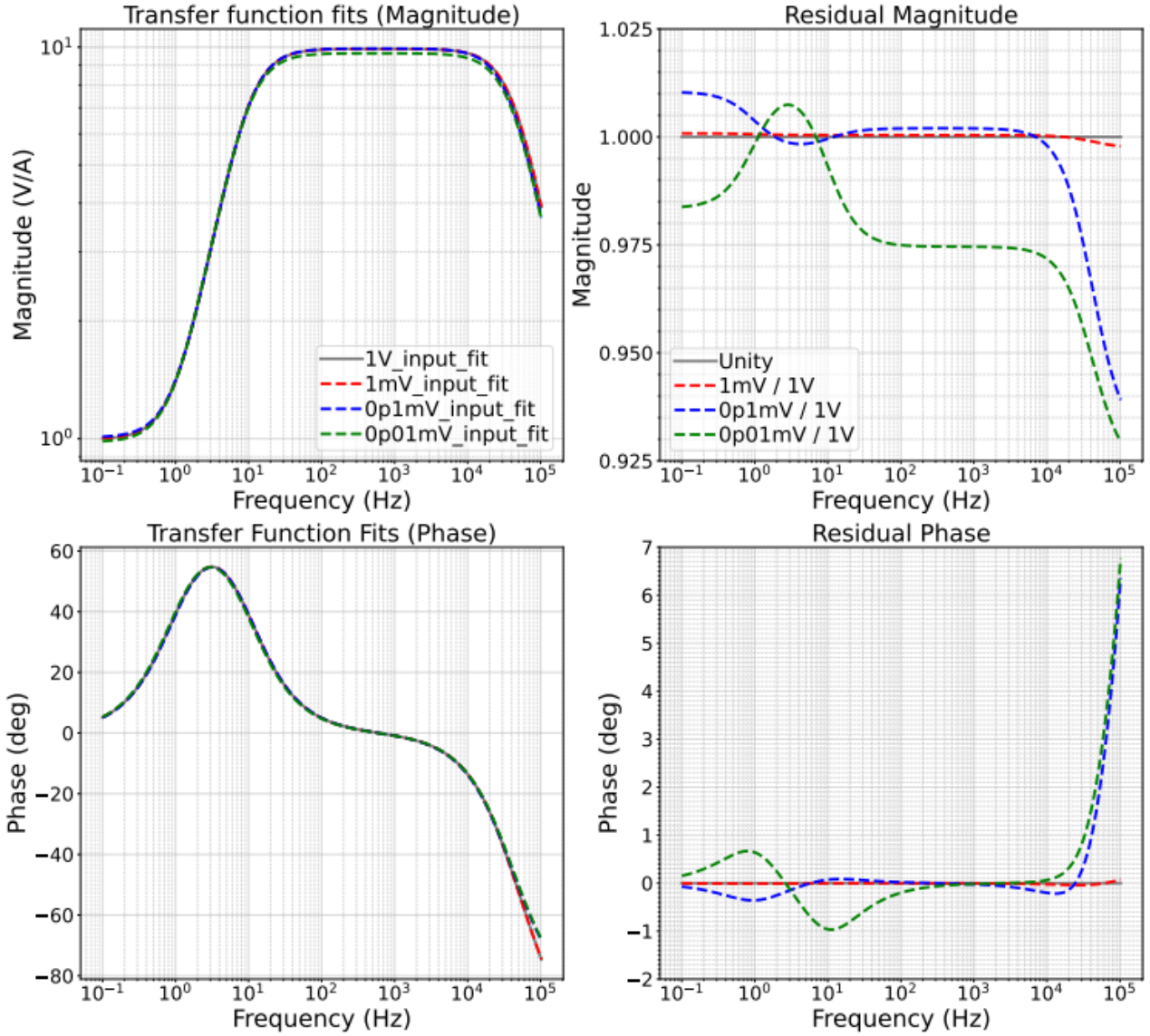

Figure 14: Overlaid residuals between the 0.1 mV, 0.05 mV, 0.01 mV and 1 V *IIRRational* transfer function fits. The top plots show Magnitude. The bottom plots show Phase. Left plots show each transfer function fit generated by *IIRRational*. Right plots show the residuals between each fit and the standard 1 V fit.

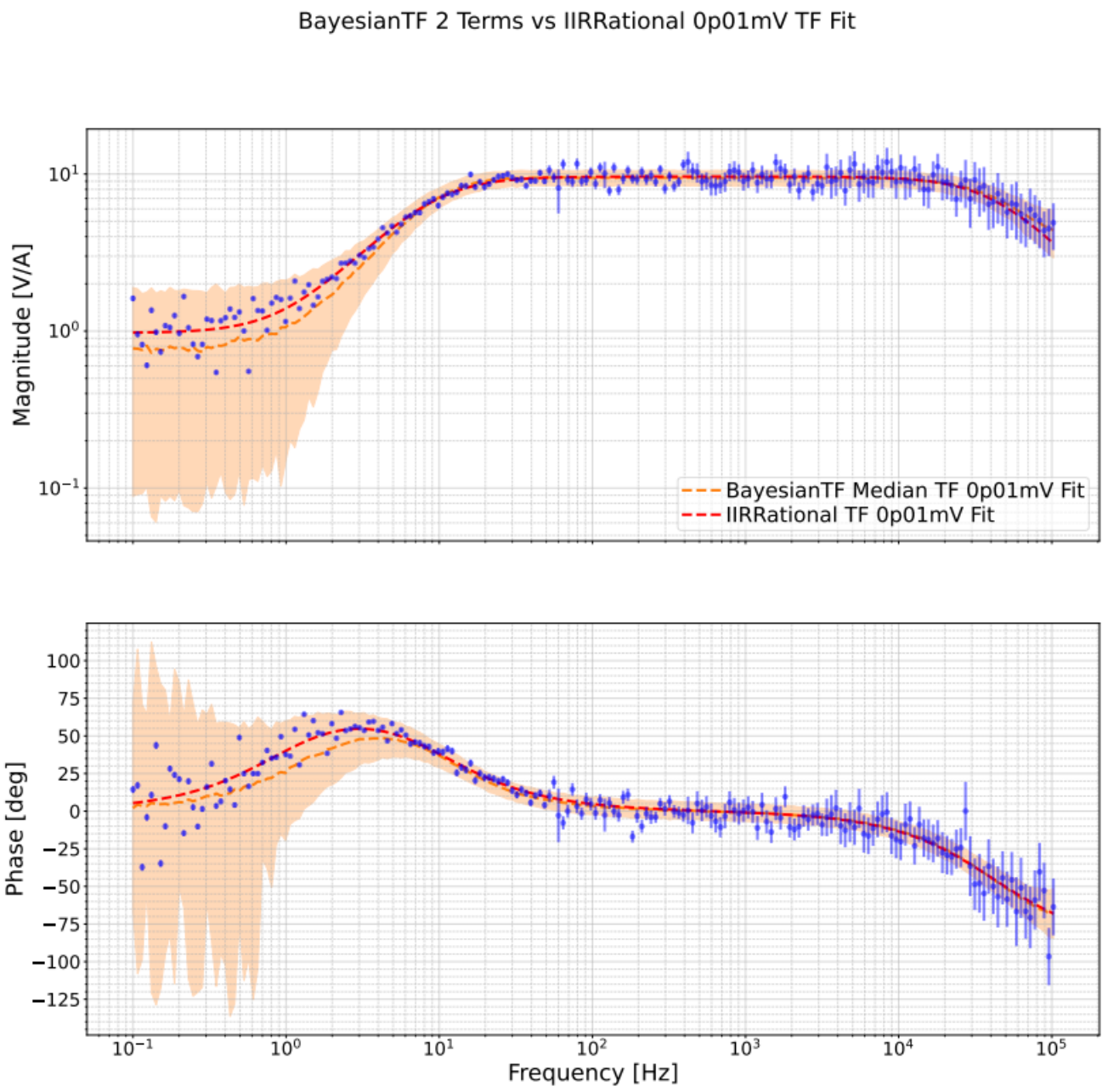

Figure 15: Bode plot showing the comparison between the 2-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. Error bars for the *IIRRational* fit are constructed with the 5th and 95th percentile fits from the posterior distribution.

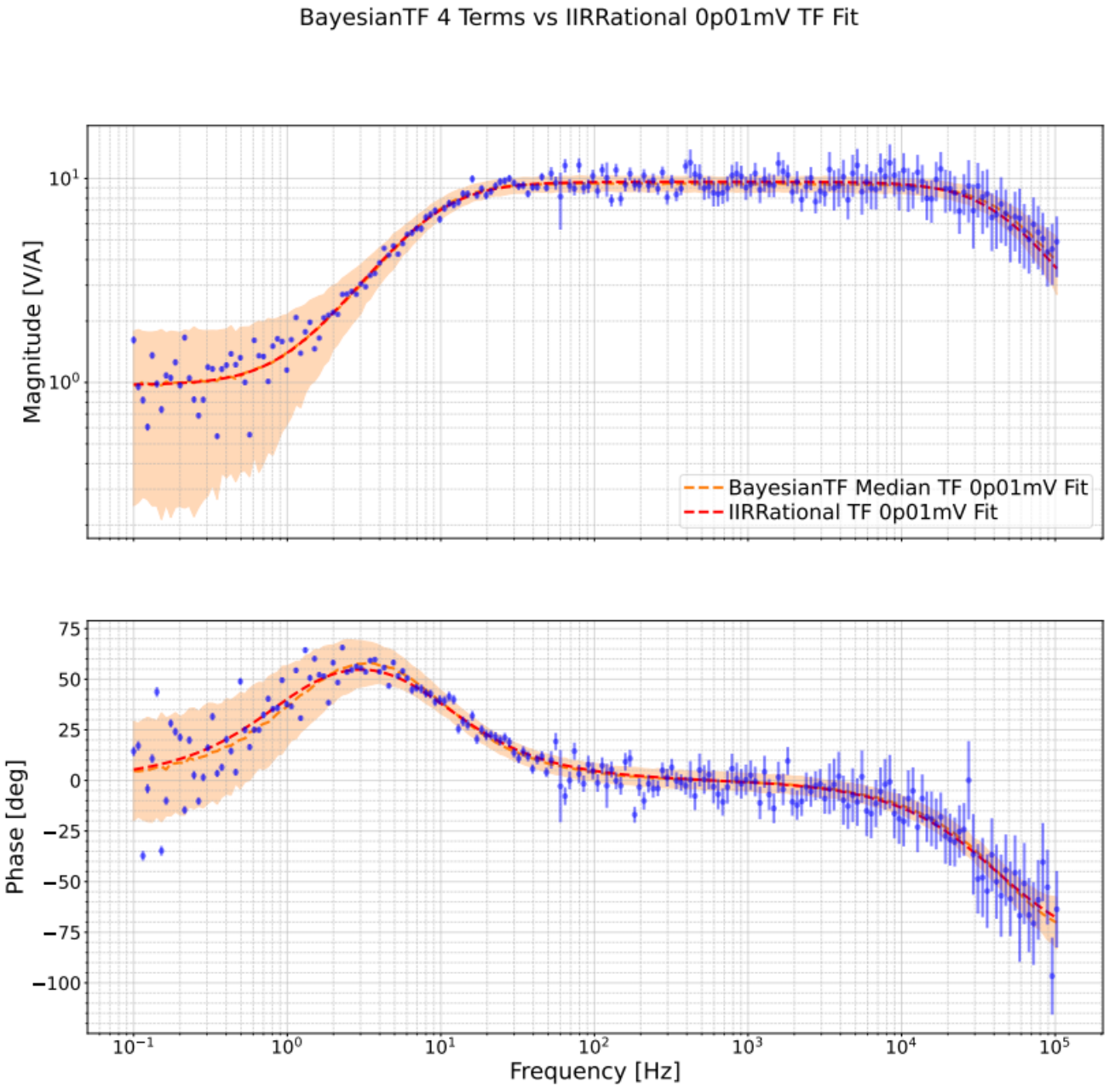

Figure 16: Bode plot showing the comparison between the 4-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. Error bars for the *IIRRational* fit are constructed with the 5th and 95th percentile fits from the posterior distribution.

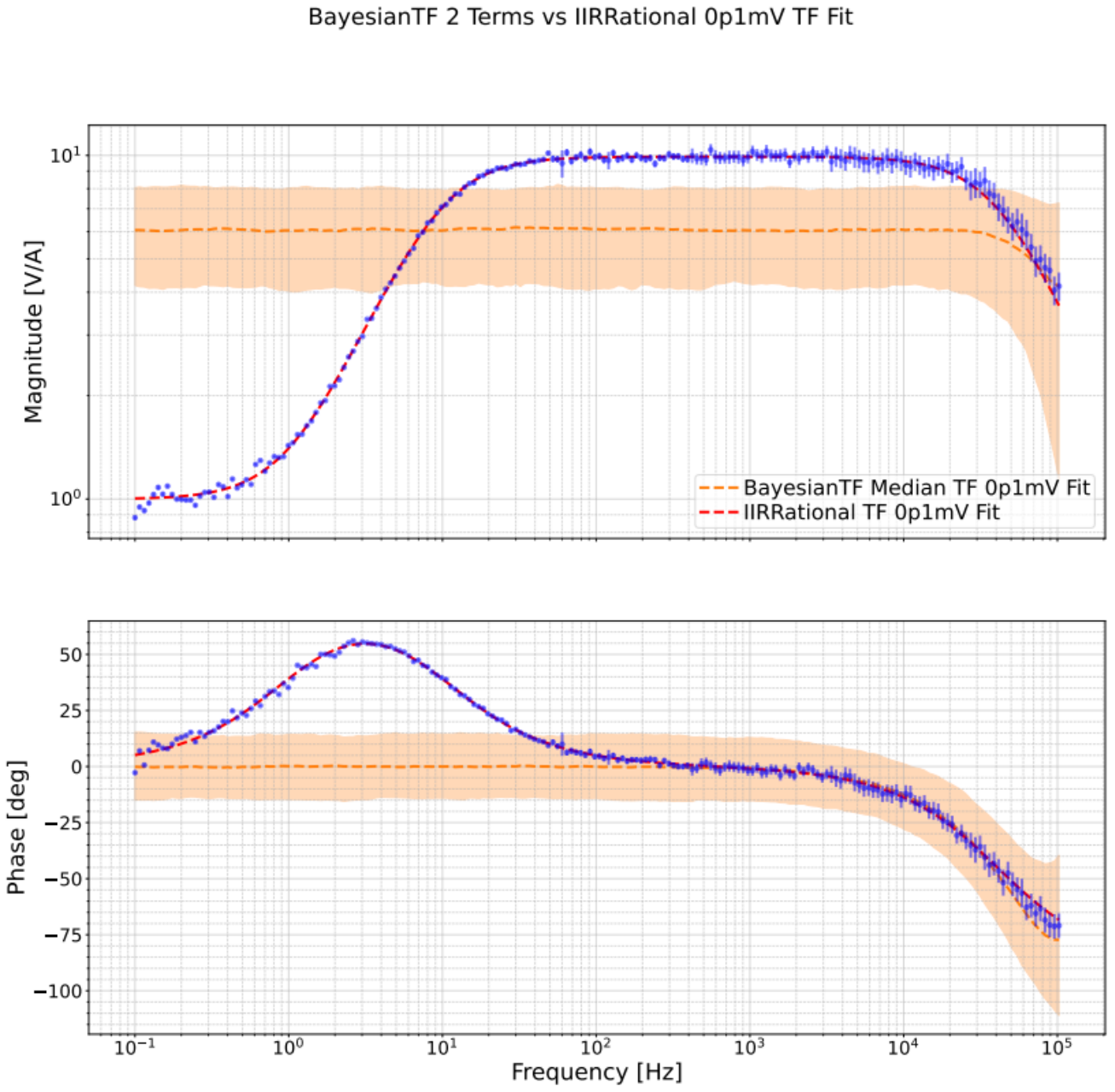

Figure 17: Bode plot showing the comparison between the 2-pole *BayesianTF* fit and *IIRRational* fit for the 0.1 mV input data set. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. Error bars for the *IIRRational* fit are constructed with the 5th and 95th percentile fits from the posterior distribution.

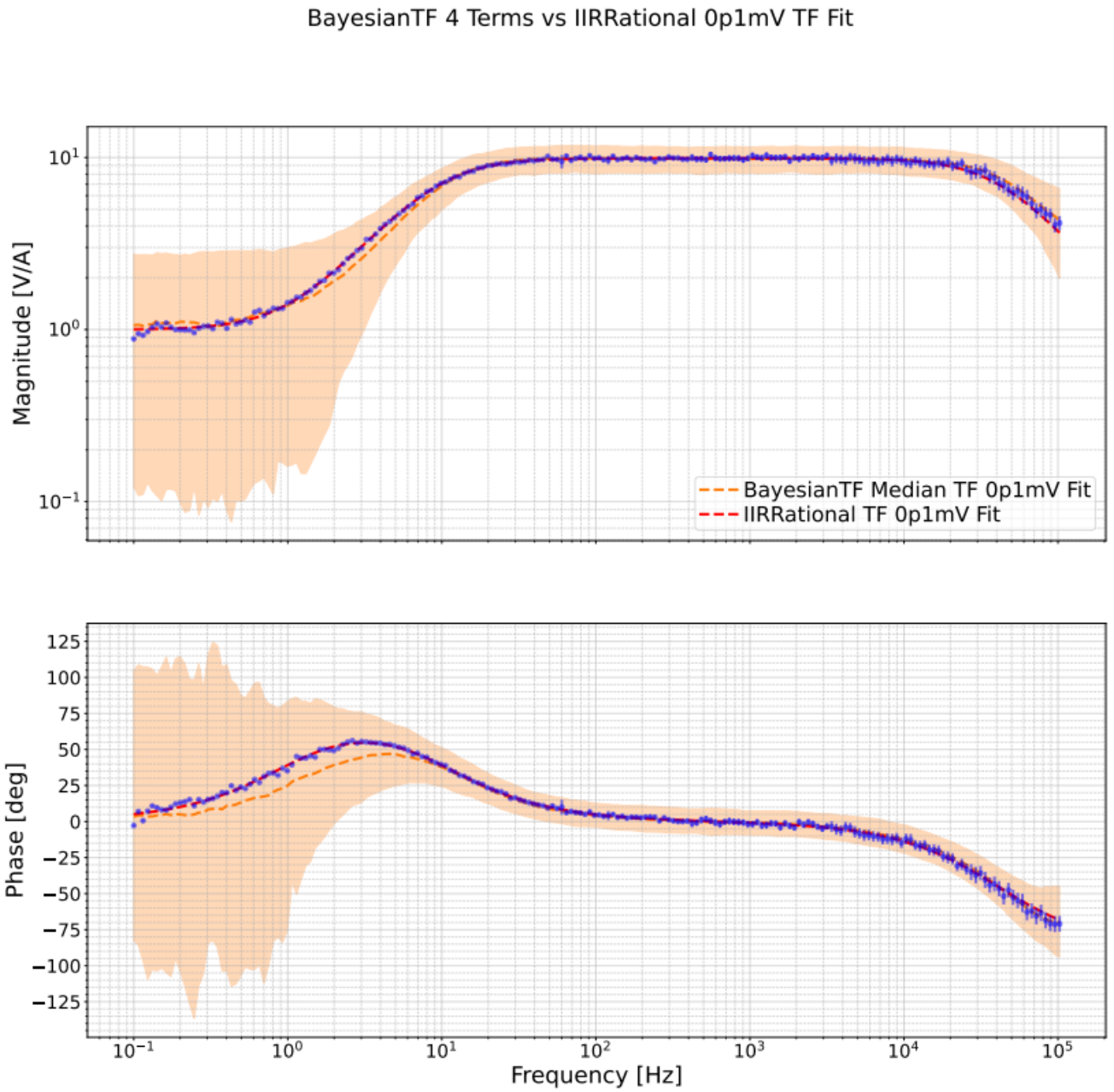

Figure 18: Bode plot showing the comparison between the 4-pole *BayesianTF* fit and *IIRRational* fit for the 0.1 mV input data set. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. Error bars for the *IIRRational* fit are constructed with the 5th and 95th percentile fits from the posterior distribution.

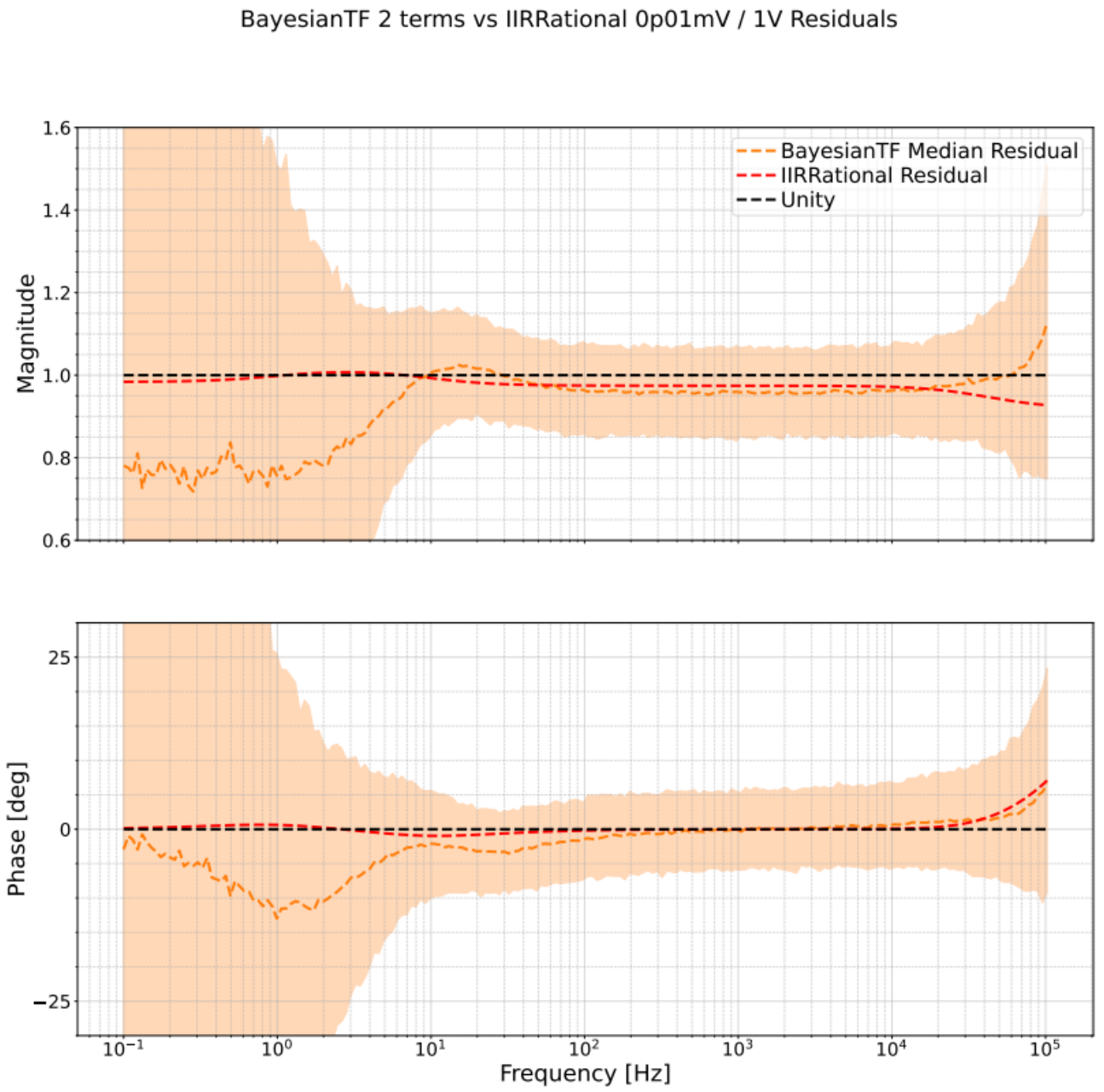

Figure 19: Bode plot showing the residuals between the 2-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set with the standard 1 V *IIRRational* fit. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. The error bars for the *BayesianTF* residual are constructed with the 5th and 95th percentile residuals from the posterior distribution of residuals.

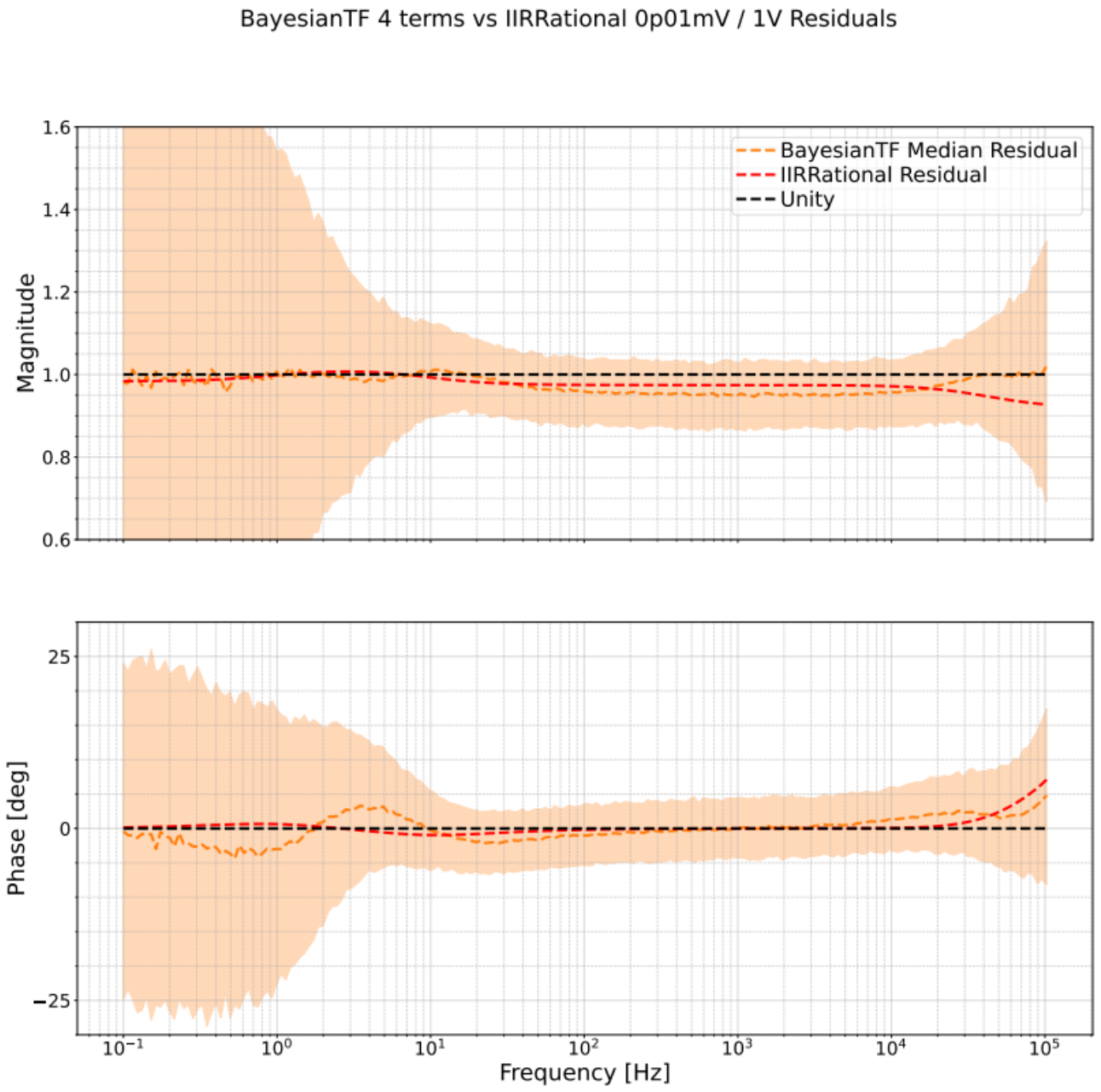

Figure 20: Bode plot showing the residuals between the 4-pole *BayesianTF* fit and *IIRRational* fit for the 0.01 mV input data set with the standard 1 V *IIRRational* fit. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. The error bars for the *BayesianTF* residual are constructed with the 5th and 95th percentile residuals from the posterior distribution of residuals.

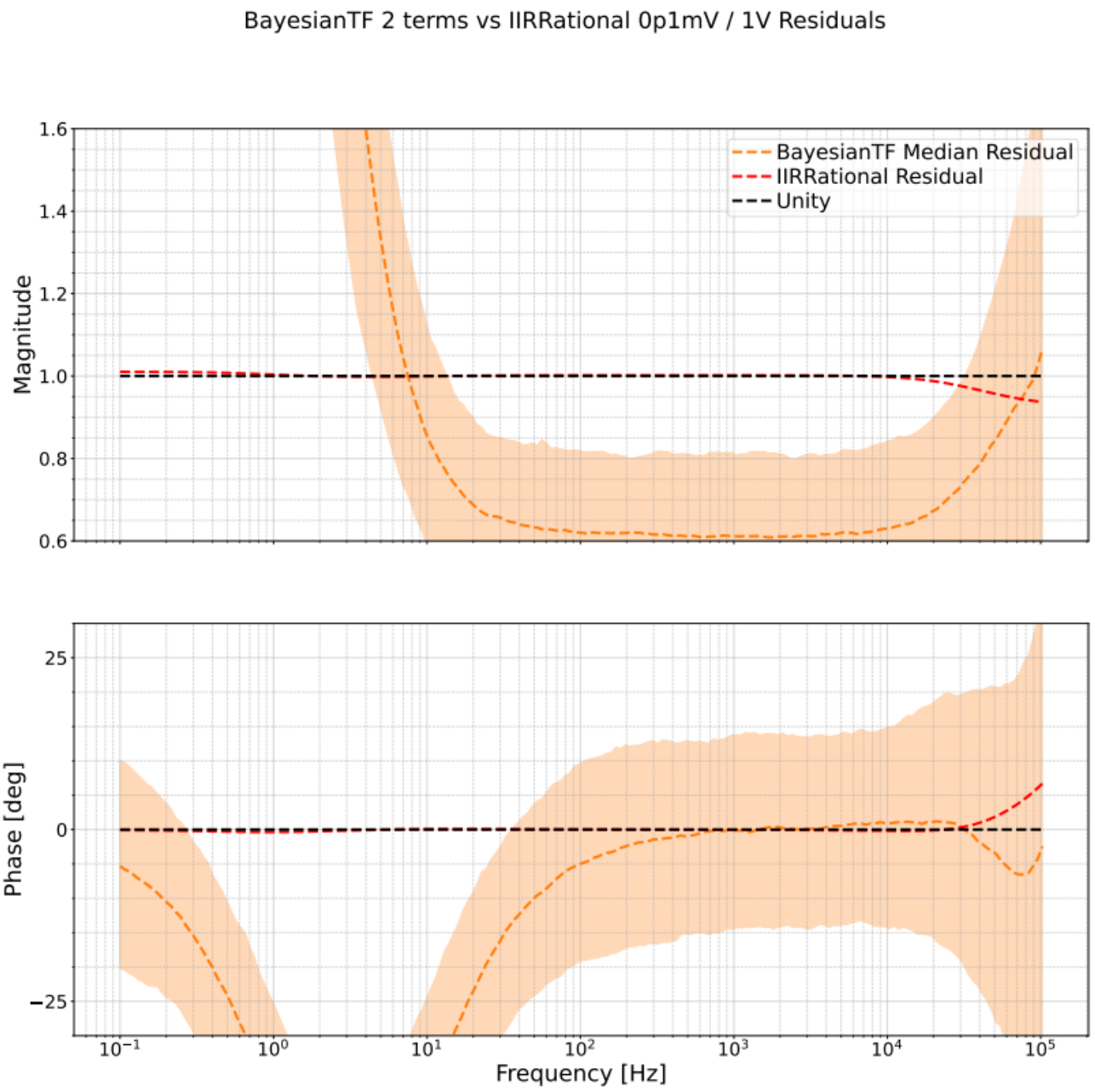

Figure 21: Bode plot showing the residuals between the 2-pole *BayesianTF* fit and *IIRRational* fit for the 0.1 mV input data set with the standard 1 V *IIRRational* fit. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. The error bars for the *BayesianTF* residual are constructed with the 5th and 95th percentile residuals from the posterior distribution of residuals.

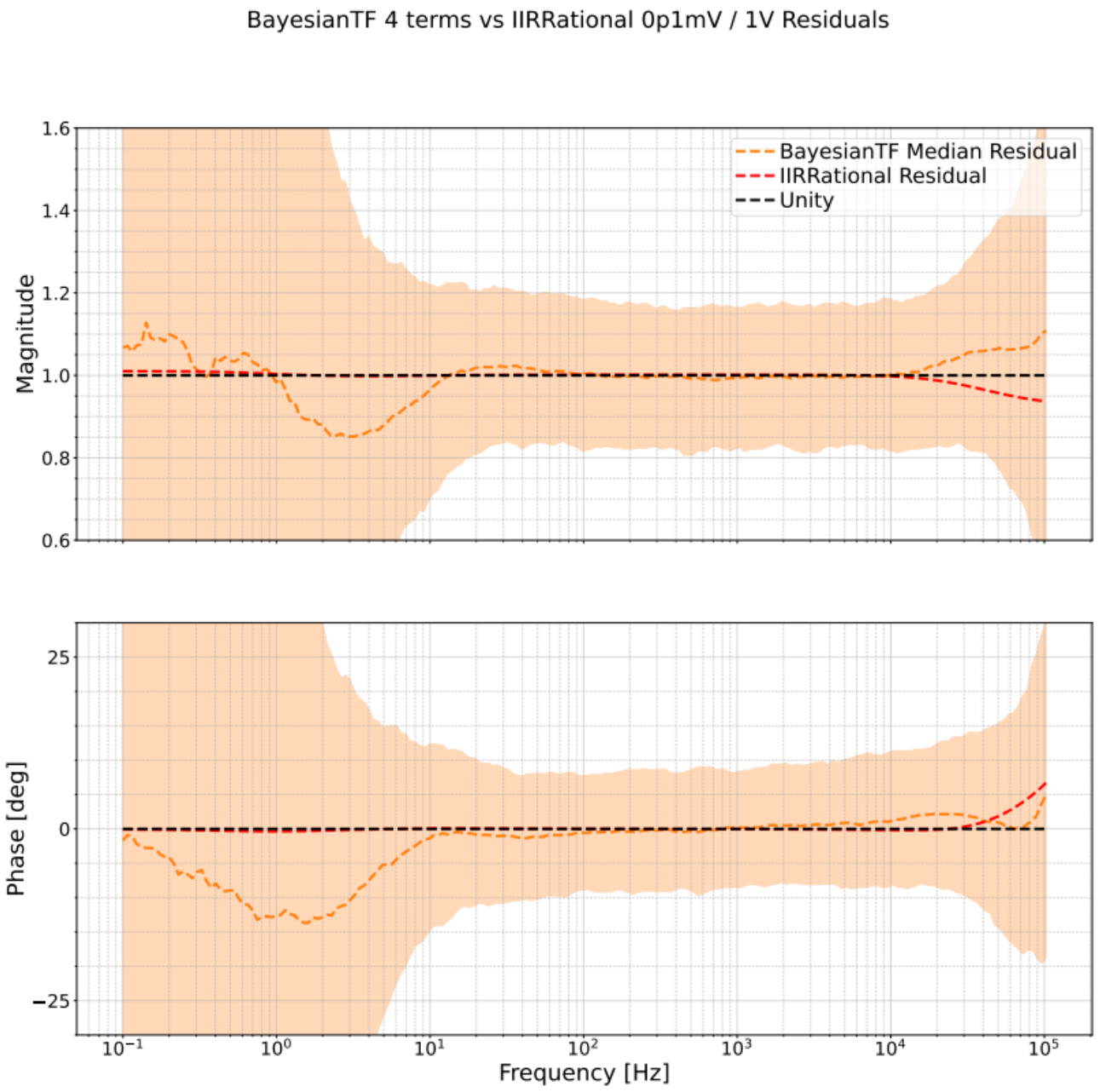

Figure 22: Bode plot showing the residuals between the 4-pole *BayesianTF* fit and *IIRRational* fit for the 0.1 mV input data set with the standard 1 V *IIRRational* fit. The top plot displays the Magnitude and the bottom plot displays the Phase. The measured data with measurement uncertainty is overlaid in blue. The error bars for the *BayesianTF* residual are constructed with the 5th and 95th percentile residuals from the posterior distribution of residuals.