TFpy-PREPRINT
REV-1 (not for distribution)

TFpy: A Python package to analyse and fit time-domain experimental data to transfer function model

M. Sotgia*
Dipartimento di Fisica, Università degli Studi di Genova, I-16146 Genova, Italy

(Dated: August 20, 2022)

Keywords: transfer function; complex space model; python package

I. INTRODUCTION

At the basis of any time-domain physical experiment –such as Gravitational-wave detection – is often needed to analyse frequency-domain data, usually presented in the form of a transfer function $H(\nu)$ for which the phase and amplitude diagrams –the Bode diagrams– are useful. Numerous Python packages already exist, most of them of broader spectrum of use [1, 2], some built to meet very specific use-cases [3, 4]. All those package refer to Fourier Transform, or Fast Fourier Transform (FFT) to perform frequency-series calculation, and most of them is non used to fit actual experimental data to TFM, in a sense familiar to us.

A non-tipical application for the Bode diagrams is to fit the transfer function models (TFM) to the experimental data. The main problem to adress is the complex-valued model shape for the transfer function $H(\nu)$. Using the phasor notation the frequency ν is expressed as $s = 2\pi\nu{\cdot}i$, where i is the complex imaginary unit. This way the gain and phase are implicitly obtained from the transfer function, which in itself can be represented in the complex value notation

$$H(s) = |H(s)| \exp[i\phi(s)].$$

[1] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods **17**, 261 (2020).

[2] Control systems library for python, `http://github.com/python-control/python-control` (2017).

[3] D. M. Macleod, J. S. Areeda, S. B. Coughlin, T. J. Massinger, and A. L. Urban, GWpy: A Python package for gravitational-wave astrophysics, SoftwareX **13**, 100657 (2021).

[4] T. T. L. Tsan, Kontrol, `https://kontrol.readthedocs.io/` (2020).

* s4942225@studenti.unige.it