

Notes on modeling for Kerr black holes: Basis learning, QNM frequencies, and spherical-spheroidal mixing coefficients

L. London¹

¹*School of Physics and Astronomy, Cardiff University, The Parade, Cardiff, CF24 3AA, United Kingdom*
(Dated: June 18, 2018)

The ongoing direct detection of gravitational wave signals is aided by representative models of theoretical predictions. In particular, the process of model based detection, and subsequent comparison of signals the general relativity's predictions are aided by the modeling of information related to perturbed Kerr black holes. Here, we summarize recent methods and models for the analytically understood gravitational wave spectra (quasi-normal mode frequencies), and harmonic structure of Kerr black holes (mixing coefficients between spherical and spheroidal harmonics). Towards the construction of these models, two algorithms, GMVP and GMVR, for the automated polynomial and rational modeling of general dimensional complex scalars are presented.

I. INTRODUCTION

In the coming years, expectations for frequent Gravitational wave (GW) detections of increasing signal-to-noise ratio (SNR) are high. Concurrent with Virgo, the Advanced LIGO (aLIGO) detectors will enter their third observing run in **late 2018**. At this time, binary black hole (BH) detections are expected at a rate of **X** per month. In this context, signal detection and subsequent inference of physical parameters hinges upon efficient models for source properties and dynamics. Most prominently, there is ongoing interest in signal models for binary BH inspiral, merger and ringdown (IMR). As the merger of isolated BHs is expected to result in a perturbed Kerr BH, there is related interest in having computationally efficient models for perturbative parameters, namely those that enable evaluation of the related ringdown radiation.

In particular, a perturbed Kerr BH (e.g. resulting from binary BH merger) will have GW radiation that rings down with characteristic frequencies, $\tilde{\omega}_{\ell mn} = \omega_{\ell mn} + i/\tau_{\ell mn}$. These discrete frequencies have associated radial and spatial functions which are *spheroidal* harmonic in nature. These frequencies and harmonic functions are the so-called Quasi-Normal Modes (QNMs). They are the eigen-solutions of the source free linearized Einstein's equations (i.e. Teukolsky's equations) for a perturbed BH with final mass, M_f , and dimensionless final spin, j_f . The well known and effective completeness of these solutions allows gravitational radiation from generic perturbations to be well approximated by a spectral (multipolar) sum which combines the complex QNM amplitude, $A_{\ell mn}$, with the spin weight -2 spheroidal harmonics, $_{-2}S_{\ell mn}$.

$$\begin{aligned} h &= h_+ - i h_\times \\ &= \frac{1}{r} \sum_{\ell mn} A_{\ell mn} e^{i\tilde{\omega}_{\ell mn} t} {}_{-2}S_{\ell mn}(j_f \tilde{\omega}_{\ell mn}, \theta, \phi) \\ &= \frac{1}{r} \sum_{\ell \bar{m}} h_{\ell \bar{m}}(t) {}_{-2}Y_{\ell \bar{m}}(\theta, \phi). \end{aligned} \quad (1)$$

In the first and second lines of Eqn. (1), we relate the observable GW polarizations, h_+ and h_\times , with the analytically understood morphology of the time domain waveform. Here, the labels ℓ and m are eigenvalues of Teukolsky's angular equations, where units of $M = 1$ (e.g. the initial mass of the binary BH system), and $c = 1$. In the third line of Eqn. (1), we represent h in terms of *spherical* harmonic multipoles. This latter form is ubiquitous for the development and implementation of IMR signal models for binary BHs.

Towards the development of these models, Eqn. (1) enters in many incarnations. In the Effective One Body (EOB) formalism, $h_{\ell \bar{m}}$ is modeled such that, after its peak (near merger), the effective functional form reduces (asymptotically) to Eqn. (1)'s second line. This view currently comes with the added assumption that $_{-2}S_{\ell mn} = {}_{-2}Y_{\ell m}$, where only $n = 0$ is explicitly considered. The consequences of this choice are discussed in reference [X]. For the so-called *Phenom* models, the frequency domain multipoles, $\tilde{h}_{\ell \bar{m}}(f)$, are constructed such that their high frequency behavior is consistent with Eqn. (1) in the time domain.

Both Phenom and EOB approaches directly use phenomenological models (i.e. fits) for the QNM frequencies, as these fits are more computationally efficient than the underlying analytic calculations, which involve the solving of continued fraction equations. In the case of PhenomHM and derivative models, fits for the QNM frequencies are used in the process of mapping $\tilde{h}_{22}(f)$ into other $\tilde{h}_{\ell \bar{m}}(f)$ [X]. In that setting, the QNM frequencies impact the morphology of each $h_{\ell \bar{m}}$ in not only ringdown, but also merger and late inspiral.

For models that assist tests of the No-Hair Theorem, and thereby only include precise ringdowns, the perspective of Eqn. (1)'s second and third lines are used to write each spherical harmonic multipole moment as

$$h_{\ell \bar{m}} = \frac{1}{r} \sum_{\ell mn} A_{\ell mn} e^{i\tilde{\omega}_{\ell mn} t} \sigma_{\ell \bar{m} \ell mn} \quad (2)$$

where, the spherical-spheroidal mixing coefficient, $\sigma_{\ell \bar{m} \ell mn}$, is

$$\sigma_{\ell \bar{m} \ell mn} = \int_{\Omega} {}_{-2}S_{\ell mn} {}_{-2}Y_{\ell \bar{m}}^* d\Omega. \quad (3)$$

In Eqn. (3), $*$ denotes complex conjugation, and Ω is the standard solid angle in spherical polar coordinates.

In practice, Eqn. (2) is computationally efficient: The calculation of each $_{-2}S_{\ell mn}$ involves a series solution which slowly converges for j_f near unity. Therefore, it is more effective to have accurate models for $\sigma_{\ell \bar{m} \ell mn}$, which can then be used directly to calculate $h_{\ell \bar{m}}$ via Eqn. (2), and thereby the GW polarizations via Eqn. (1).

In this combined context, it is clear that the modeling of QNM frequencies, $\tilde{\omega}_{\ell mn}$, and spherical-spheroidal mixing coefficients, $\sigma_{\ell \bar{m} \ell mn}$, underpin a wide range of GW signal models. While models for $\tilde{\omega}_{\ell mn}$ and $\sigma_{\ell \bar{m} \ell mn}$ are present in many publications, there exist minor shortcomings which we wish to address here.

For the QNM frequencies, it is well known that for nearly

extremal BHs (i.e. $j_f \rightarrow 1$) some of the frequencies have zero-damping (i.e. $\tau_{\ell mn} \rightarrow \infty$). In the context of aLIGO data analysis, where source parameters are estimated using routines which sample over the space of possible BH masses and spins, it is useful to have accurate physical behavior in the extremal limit. Here, we present the first models for $\tilde{\omega}_{\ell mn}$ that explicitly account for zero-damping in the extremal Kerr limit.

For the modeling of $\sigma_{\tilde{\ell} m \ell mn}$, we note that the results presented in [X] are limited to cases where the azimuthal indices, ℓ and $\tilde{\ell}$, are less than or equal to 3. As the most advanced signal models include at least ℓ or $\tilde{\ell}$ of 3, there is use in extending prior results. In particular, it is well known that σ_{43330} can have a significant impact on the (4, 3) spherical multipole. For consistency with the multipolar content of current ring-down models, here, we extend previous results to include the most significant multipoles with $\ell \leq 5$.

In parallel, the methods for modeling $\tilde{\omega}_{\ell mn}$ and $\sigma_{\tilde{\ell} m \ell mn}$ have been dispersed: different phenomenological techniques have been used under no coherent framework. Here we will present linear modeling techniques, namely the greedy-multivariate-polynomial (GMVP) and greedy-multivariate-rational (GMVR) algorithms, in which model terms are iteratively learned with no initial guess. The description of GMVP given here is complementary to similar algorithms used to model QNM excitation amplitudes, $A_{\ell mn}$, as present in reference [cite]. As we will discuss, the GMVR algorithm is an iterative approach to the (pseudo) non-linear modeling of multivariate rational functions. Both GMVP and GMVR are intended for use with low noise data (e.g. the results of analytic calculations), and each employs a reverse (or negative) greedy algorithm to counter over modeling. As the underlying process for GMVP and GMVR is stepwise regression, highly correlated basis vectors (i.e. polynomial terms) are handled via an approach we will call *degree tempering*. It will be demonstrated that these approaches are readily capable of modeling the complex valued $\tilde{\omega}_{\ell mn}$ and $\sigma_{\tilde{\ell} m \ell mn}$. Results suggest that the versions of GMVP and GMVR presented here may have broad application in instances where training data are approximately noiseless, and an initial guess is difficult to obtain.

The plan of the paper is as follows. In section Section (II), we outline the GMVP and GMVR algorithms. In Section (III), we demonstrate the application of each algorithm. We first consider the application of GMVP to the modeling of QNM frequencies. We then consider the application of GMVR to the modeling of spherical-spheroidal mixing coefficients. In Section (IV), we review the performance of GMVP and GMVR, and we discuss potential applications for these methods.

II. METHODS

Within the topic of regression, linear regression has particular advantages: its matrix based formulation can be computationally efficient, and it does not require initial guesses for model parameters. Perhaps most intriguingly, the formal series expansions of smooth functions support linear and rational models (e.g. Pade approximants) having application to many datasets. Here we will develop algorithms for the linear modeling of scalar functions (real or complex valued) of many variables.

If we consider a scalar function, f , of N variables sampled in j , $\vec{x}_j = \{x_{\alpha j}\}_{\alpha=0}^N$, then $f(\vec{x})_j$ can be represented (possibly in-

accurately) as a sum over K linearly independent basis functions, $\phi_k(\vec{x}_j)$:

$$f(\vec{x})_j = \sum_{k=0}^K \mu_k \phi_{kj}. \quad (4)$$

The central player in Eqn. (4) is the set of basis coefficients μ_k . Typically, one chooses or derives $\phi_k(\vec{x}_j)$ to capture inherent features of $f(\vec{x}_j)$. With $\phi_k(\vec{x}_j)$ assumed to be known, the linear representation (Eqn. 4) is lastly defined by the set of μ_k .

From here it is useful to note that Eqn. (4) has a linear homogeneous matrix form. In particular, defining $\hat{U} = \{\phi_{kj}\}$, and $\vec{f} = \{f_j\}$, then $\vec{f} = \hat{U} \vec{\mu}$. This implies that

$$\vec{\mu} = \hat{P} \vec{f}, \quad (5)$$

where

$$\hat{P} = \left((\hat{U}^{-1})^T \hat{U}^{-1} \right)^{-1} \quad (6)$$

is the *pseudo-inverse* of \hat{U} . Equations (4), and related discussion through Eqn. (6) illustrate the most rudimentary solution to the linear modeling problem. However, there are many ways to expand upon and refine the solution presented thus far. In the following subsections we will consider two such approaches. First we will consider the general polynomial modeling of multivariate scalar functions. This will encompass the GMVP algorithm. Second, we will build upon the GMVP approach by considering models of rational functions: polynomials divided by polynomials. To consider these two approaches in a largely automated way (i.e. where the set of possible basis functions is known, but the select basis functions ultimately used are *learned*), we will make use of the *greedy* algorithm approach.

A. A Generic Greedy Algorithm

While we most often want a single model for a given dataset (e.g. some approximation of \vec{f} from numerical calculation or experiment), there are often many more modeling choices than desired. In particular, if we refer to our set of all possible basis functions as our “*symbol space*”, then the problem of determining how many, and which basis vectors (i.e. symbols) to use is a problem of combinatoric complexity.

A well known method for finding an approximate solution to this problem is the so-called “greedy” algorithm. The premise is this: We will iteratively construct models with increasing number of symbols. The process begins by finding the single symbol (basis vector) that yields the most accurate model (in the sense of minimizing least-squares error). That encompasses the first iteration of a process in which we will greedily add symbols to our model. In each following iteration, all remaining symbols are added to the model one at a time, and the single additional symbol which yields the greatest increase in model accuracy is kept for the following iteration. In this way, a list of optimal model symbols is learned. The forward process is to end when the model accuracy (or changes thereof) pass a specified threshold. This rough algorithmic picture encapsulated in Alg. 1.

Note that we will use the terms “basis vector” and “model symbol” interchangeably. However, in practice, it can be use-

ful to represent basis vectors symbolically (e.g. with strings), and have a way of bijectively mapping between symbols and numerical basis vectors.

Algorithm 1 A positive (forward) greedy algorithm, PGREEDY. Note that a required input, \mathcal{A} , is a function that takes in a list of basis symbols, and outputs an estimator of fit error. In this setting, \mathcal{A} is assumed to have access to peripheral information, such as the training data.

```

1: Input:  $\{\lambda_{bulk} = \text{basis symbols}, \mathcal{A} = \text{action}, tol = \text{greedy tolerance}\}$ 

2: Define empty list of kept symbols:  $\lambda_{kept} = \{\}$ 
3: Initialize estimator value and loop boolean:  $\epsilon_{last} = \text{inf}, done = \text{False}$ 
4: while not done do
5:    $\epsilon_{min} = \epsilon_{last}$ 
6:   for  $\lambda$  in  $\lambda_{bulk}$  do
7:      $\lambda_{trial} = \lambda_{kept} \cup \lambda$            (add  $\lambda$  to  $\lambda_{kept}$ )
8:      $\epsilon = \mathcal{A}(\lambda_{trial})$            (action returns fit error)
9:     if  $\epsilon < \epsilon_{min}$  then
10:       $\epsilon_{min} = \epsilon$            (store trial min)
11:       $\lambda_{min} = \lambda_{trial}$ 
12:     end if
13:   end for
14:    $done = |\epsilon_{min} - \epsilon_{last}| < tol$ 
15:   if not done then
16:      $\epsilon_{last} = \epsilon_{min}$ 
17:      $\lambda_{kept} = \lambda_{kept} \cup \lambda_{min}$    (update kept symbols)
18:   end if
19: end while

20: Output: Greedy Basis,  $\{\lambda_{kept}\}$ 

```

B. Greedy Multivariate Polynomial Fitting

The study of smooth scalar functions (e.g. \vec{f}) often centers about the Taylor series expansion. In that instance, it is clear that any infinitely differentiable scalar function of many variables can be represented in terms of its derivatives by

$$f(\vec{x} + \vec{h}) = e^{\vec{x} \cdot \vec{\nabla}'} f(\vec{x}')|_{\vec{x}'=\vec{h}} \quad (7)$$

$$\approx \sum_{k=0}^K \frac{1}{k!} (\vec{x} \cdot \vec{\nabla}')^k f(\vec{x}')|_{\vec{x}'=\vec{h}}$$

From the first to second line of Eqn. (7), we have used the definition of the exponential function (i.e. its series expansion form). In the second line, the equality has been replaced by an approximation as we have limited the linear representation to K terms. This latter point is key to the perspective of GMVP: given training data thought to be related to a smooth multivariate function, it may be well approximated by a truncated series expansion in the appropriate coordinate basis.

This truncated expansion happens to be a polynomial in at most N variables. In this setting, the uncertainty of which and how many basis terms to include makes this a problem ripe for the application of Eqn. (5) and Alg. (1).

In the current setting, the basis symbols required by Alg. (1) are the multinomial terms in Eqn. (7). Each term is an element of the cartesian inner-product of the

power-set of each model dimension. That is $\lambda_{bulk} = \{x_0, x_1 \dots x_N, \dots x_0^1, x_0^2, x_0^3, \dots, (x_0^K x_1^K \dots x_N^K)\}$. Note that in practice it may be useful to encode elements of λ_{bulk} with strings representing their constituents (e.g. $x_0 x_0 x_1 x_2 x_2 x_4$ could be represented by the string “001224”).

The action, $\mathcal{A}(\lambda_{trial})$, required by Alg. (1) encompasses the evaluation of Eqn. (5) to solve for the basis coefficients, μ_k , and the calculation of the modeling error. An explicit sketch of this is given by Alg. (2).

Algorithm 2 \mathcal{A}_{GMVP} , the action for GMVP. Model calculation given basis symbols, and output of model error estimate.

```

1: Input:  $\lambda_{trial}$ 

2: Calculate  $\mu_k$  via Eqn. (5).
3: Calculate the model representation error, e.g.:  $\epsilon = |var(\hat{U}\vec{\mu} - \vec{f})|/var(\vec{f})$ , where var is the variance of the respective list.

4: Output:  $\epsilon$ 

```

The combination of these two ideas alone results in an algorithm prone to a deficit of stepwise methods: there is the very likely possibility that the algorithm would confuse correlated basis vectors (e.g. x^2 may be confused for x^4). To counter this, we may incrementally increase, or *temper*, the maximum allowed multinomial order. Recall that a term such as $x_0 x_1 x_3^2$ has an effective order of 4. The combination of degree tempering with the greedy approach results in the GMVP algorithm as presented in Alg. (3)

Algorithm 3 GMVP, a degree tempered stepwise algorithm for multivariate polynomial modeling of scalar data.

```

1: Input:  $\{\mathbf{x}, \mathbf{y}, \text{max\_order} = 6\}$ 

2: Define bulk symbol space using cartesian inner-product of the power set of each basis symbol
3: Define list of allowed polynomial degrees
4: for all degrees do
5:   Select all symbols with degree less than or equal to current
6:   Apply PGREEDY to get symbol subset and estimator val
7:   if estimator has stalled then
8:     Break
9:   end if
10: end for
11: Apply negative greedy to get final model

12: Output: Final model

```

Algorithm 4 GMVR, a degree tempered stepwise algorithm for multivariate rational modeling of scalar data.

```

1: Input: {x, y}

2: Define action using matrix least squares
3: Define bulk symbol space using cartesian inner-product
4: Define list of allowed polynomial degrees
5: for all degrees do
6:   Select all numerator symbols with degree less than or
   equal to current
7:   Select all denominator symbols with degree less than
   or equal to current, enforcing no constant term
8:   Construct new symbol space by tagging each polynomial
   symbol with a boolean denoting whether it is in the
   numerator or denominator
9:   Apply PGREEDY to get symbol subset and estimator
   val
10:  if estimator has stalled then
11:    Break
12:  end if
13: end for
14: Apply negative greedy to get final model

15: Output: Final model

```

III. RESULTS

$$\tilde{\Omega}_{220}(\kappa) = 1.0 + \kappa(1.5578e^{2.9031i} + 1.9510e^{5.9210i}\kappa + 2.0997e^{2.7606i}\kappa^2 + 1.4109e^{5.9143i}\kappa^3 + 0.4106e^{2.7952i}\kappa^4) \quad (8)$$

$$\tilde{\Omega}_{221}(\kappa) = 1.0 + \kappa(1.8709e^{2.5112i} + 2.7192e^{5.4250i}\kappa + 3.0565e^{2.2857i}\kappa^2 + 2.0531e^{5.4862i}\kappa^3 + 0.5955e^{2.4225i}\kappa^4) \quad (9)$$

$$\tilde{\Omega}_{330}(\kappa) = 1.5 + \kappa(2.0957e^{2.9650i} + 2.4696e^{5.9967i}\kappa + 2.6655e^{2.8176i}\kappa^2 + 1.7584e^{5.9327i}\kappa^3 + 0.4991e^{2.7817i}\kappa^4) \quad (10)$$

$$\tilde{\Omega}_{331}(\kappa) = 1.5 + \kappa(2.3391e^{2.6497i} + 3.1399e^{5.5525i}\kappa + 3.5916e^{2.3472i}\kappa^2 + 2.4490e^{5.4435i}\kappa^3 + 0.7004e^{2.2830i}\kappa^4) \quad (11)$$

$$\tilde{\Omega}_{440}(\kappa) = 2.0 + \kappa(2.6589e^{3.0028i} + 2.9783e^{6.0510i}\kappa + 3.2184e^{2.8775i}\kappa^2 + 2.1276e^{5.9897i}\kappa^3 + 0.6034e^{2.8300i}\kappa^4) \quad (12)$$

$$\tilde{\Omega}_{430}(\kappa) = 1.5 + \kappa(0.2050e^{0.5953i} + 3.1033e^{3.0162i}\kappa + 4.2361e^{6.0388i}\kappa^2 + 3.0289e^{2.8262i}\kappa^3 + 0.9084e^{5.9152i}\kappa^4) \quad (13)$$

$$\tilde{\Omega}_{550}(\kappa) = 2.5 + \kappa(3.2405e^{3.0279i} + 3.4906e^{6.0888i}\kappa + 3.7470e^{2.9212i}\kappa^2 + 2.4725e^{6.0365i}\kappa^3 + 0.6994e^{2.8766i}\kappa^4) \quad (14)$$

$$\begin{aligned} \tilde{\Omega}_{320}(\kappa) = & 1.0225e^{0.0049i} + 0.2473e^{0.6653i}\kappa + 1.7047e^{3.1383i}\kappa^2 + 0.9460e^{0.1632i}\kappa^3 + 1.5319e^{5.7036i}\kappa^4 \\ & + 2.2805e^{2.6852i}\kappa^5 + 0.9215e^{5.8417i}\kappa^6 \end{aligned} \quad (15)$$

$$\begin{aligned} \tilde{\Omega}_{210}(\kappa) = & 0.5891e^{0.0435i} + 0.1890e^{2.2899i}\kappa + 1.1501e^{5.8101i}\kappa^2 + 6.0459e^{2.7420i}\kappa^3 + 11.1263e^{5.8441i}\kappa^4 \\ & + 9.3471e^{2.6694i}\kappa^5 + 3.0384e^{5.7915i}\kappa^6 \end{aligned} \quad (16)$$

$$\sigma_{22220} = 0.99733 e^{6.2813i} + 0.0075336 \frac{14.592 e^{5.0601i} \kappa + (28.761 e^{1.629i}) \kappa^2 + (14.511 e^{4.6362i}) \kappa^3 + (1.9624 e^{3.0113i})}{1 + 0.88674 e^{3.0787i} \kappa + (1.002 e^{0.13211i}) \kappa^2 + (0.082148 e^{5.6369i}) \kappa^3} \quad (17)$$

$$\sigma_{21210} = 0.99716 e^{6.2815i} + 0.0063542 \frac{1.4345 \times 10^5 e^{4.5061i} \kappa + (3.5469 \times 10^5 e^{1.7327i}) \kappa^2 + (2.4038 \times 10^5 e^{5.1629i}) \kappa^3 + (6026.9 e^{1.8881i})}{1 + 73780 e^{4.5545i} \kappa + (97494 e^{1.398i}) \kappa^2 + (34815 e^{4.5623i}) \kappa^3} \quad (18)$$

$$\sigma_{22221} = 0.99683 e^{6.2782i} + 0.020758 \frac{15.077 e^{4.8323i} \kappa + (31.139 e^{1.585i}) \kappa^2 + (15.449 e^{4.6727i}) \kappa^3 + (0.71897 e^{2.8084i})}{1 + 0.80592 e^{3.3995i} \kappa + (0.69502 e^{0.54275i}) \kappa^2 + (0.35613 e^{5.9545i}) \kappa^3} \quad (19)$$

$$\sigma_{32320} = 0.99009 e^{6.2804i} + 0.02369 \frac{71893 e^{1.2395i} \kappa + (1.7055 \times 10^5 e^{5.0371i}) \kappa^2 + (1.2947 \times 10^5 e^{2.359i}) \kappa^3 + (1935.5 e^{4.668i})}{1 + 38206 e^{1.2254i} \kappa + (35811 e^{3.9618i}) \kappa^2 + (8378.3 e^{0.11726i}) \kappa^3} \quad (20)$$

$$\sigma_{33331} = 0.99478 e^{6.2688i} + 0.040478 \frac{4.4113 e^{1.2501i} \kappa + (11.588 e^{0.27959i}) \kappa^2 + (17.322 e^{3.7904i}) \kappa^3 + (0.67724 e^{2.5797i})}{1 + 3.8782 e^{2.2864i} \kappa + (3.4913 e^{5.6655i}) \kappa^2 + (1.0368 e^{2.9082i}) \kappa^3} \quad (21)$$

$$\sigma_{32221} = 0.02203 e^{0.16452i} + 0.073233 \frac{24.932 e^{1.0181i} \kappa + (30.197 e^{4.4047i}) \kappa^2 + (11.274 e^{2.981i}) \kappa^3 + (2.4374 e^{6.1959i})}{1 + 11.397 e^{3.9953i} \kappa + (10.915 e^{5.8025i}) \kappa^2 + (7.2196 e^{1.8176i}) \kappa^3} \quad (22)$$

$$\sigma_{33330} = 0.99569 e^{6.2785i} + 0.014546 \frac{7.2112 e^{0.62811i} \kappa + (6.5381 e^{4.6216i}) \kappa^2 + (4.451 e^{2.9228i}) \kappa^3 + (1.7113 e^{2.9527i})}{1 + 1.4974 e^{1.6687i} \kappa + (1.5288 e^{5.3885i}) \kappa^2 + (0.52114 e^{2.5471i}) \kappa^3} \quad (23)$$

$$\sigma_{32220} = 0.020598 e^{0.04743i} + 0.06919 \frac{2.7657 e^{2.133i} \kappa + (3.9562 e^{4.653i}) \kappa^2 + (2.3364 e^{2.6444i}) \kappa^3 + (2.399 e^{6.2767i})}{1 + 1.0595 e^{4.7865i} \kappa + (0.91308 e^{2.887i}) \kappa^2 + (0.69468 e^{0.1912i}) \kappa^3} \quad (24)$$

$$\sigma_{43330} = 0.028112 e^{0.048488i} + 0.086383 \frac{12.087 e^{0.47221i} \kappa + (30.626 e^{3.3281i}) \kappa^2 + (16.328 e^{6.1785i}) \kappa^3 + (2.3603 e^{6.2662i})}{1 + 4.9638 e^{3.5931i} \kappa + (6.2552 e^{6.2001i}) \kappa^2 + (1.4538 e^{2.5539i}) \kappa^3} \quad (25)$$

$$\sigma_{43430} = 0.98735 e^{6.2795i} + 0.033028 \frac{7.0084 \times 10^5 e^{1.1067i} \kappa + (1.843 \times 10^6 e^{4.8808i}) \kappa^2 + (1.4367 \times 10^6 e^{2.1412i}) \kappa^3 + (13844 e^{4.5601i})}{1 + 3.5667 \times 10^5 e^{1.0149i} \kappa + (3.274 \times 10^5 e^{3.7746i}) \kappa^2 + (88621 e^{0.10095i}) \kappa^3} \quad (26)$$

$$\sigma_{44440} = 0.99478 e^{6.2776i} + 0.024791 \frac{6.5172 e^{0.79835i} \kappa + (7.7748 e^{4.2485i}) \kappa^2 + (1.1577 e^{1.5905i}) \kappa^3 + (1.2434 e^{2.9616i})}{1 + 0.44548 e^{1.2496i} \kappa + (0.59437 e^{5.6732i}) \kappa^2 + (0.24743 e^{2.8292i}) \kappa^3} \quad (27)$$

$$\sigma_{55550} = 0.99434 e^{6.2773i} + 0.03126 \frac{6.5508 e^{0.93398i} \kappa + (8.0558 e^{4.2881i}) \kappa^2 + (0.92971 e^{1.0436i}) \kappa^3 + (1.0904 e^{2.9712i})}{1 + 0.23128 e^{1.7666i} \kappa + (0.54958 e^{5.9178i}) \kappa^2 + (0.213 e^{3.0092i}) \kappa^3} \quad (28)$$

IV. DISCUSSION

- [1] E. Leaver, "An Analytic representation for the quasi normal modes of Kerr black holes," *Proc.Roy.Soc.Lond.*, vol. A402, pp. 285–298, 1985.

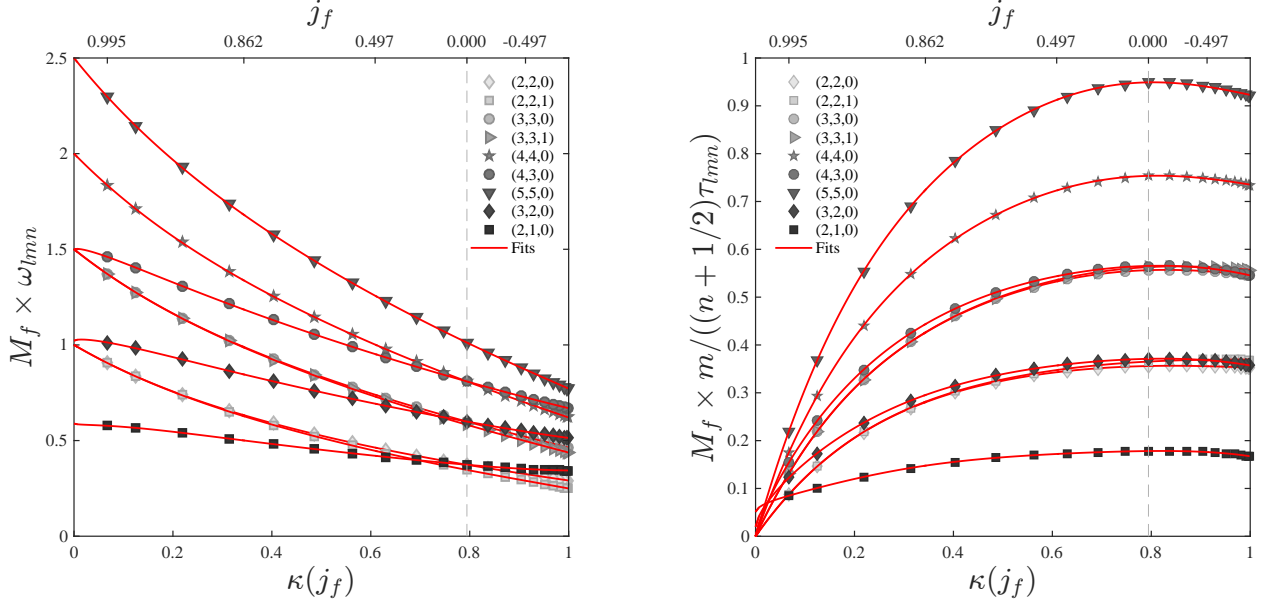


FIG. 1. Fits of dimensionless QNM central frequencies (solid lines) along with select numerical values (grey markers) computed using Leaver's method [1]. Before the application of $\kappa(j)$, points are spaced between -0.995 and 0.995 according to 0.995 times the sin of a fiducial angle which is uniformly spaced between $-\pi/2$ and $\pi/2$. Values of j are shown in the upper axis for κ at $l = m$. The grey dashed line marks the value of κ where $j = 0$. Fits of dimensionless QNM decay rates (solid lines) along with select numerical values (grey markers) computed using Leaver's method [1].

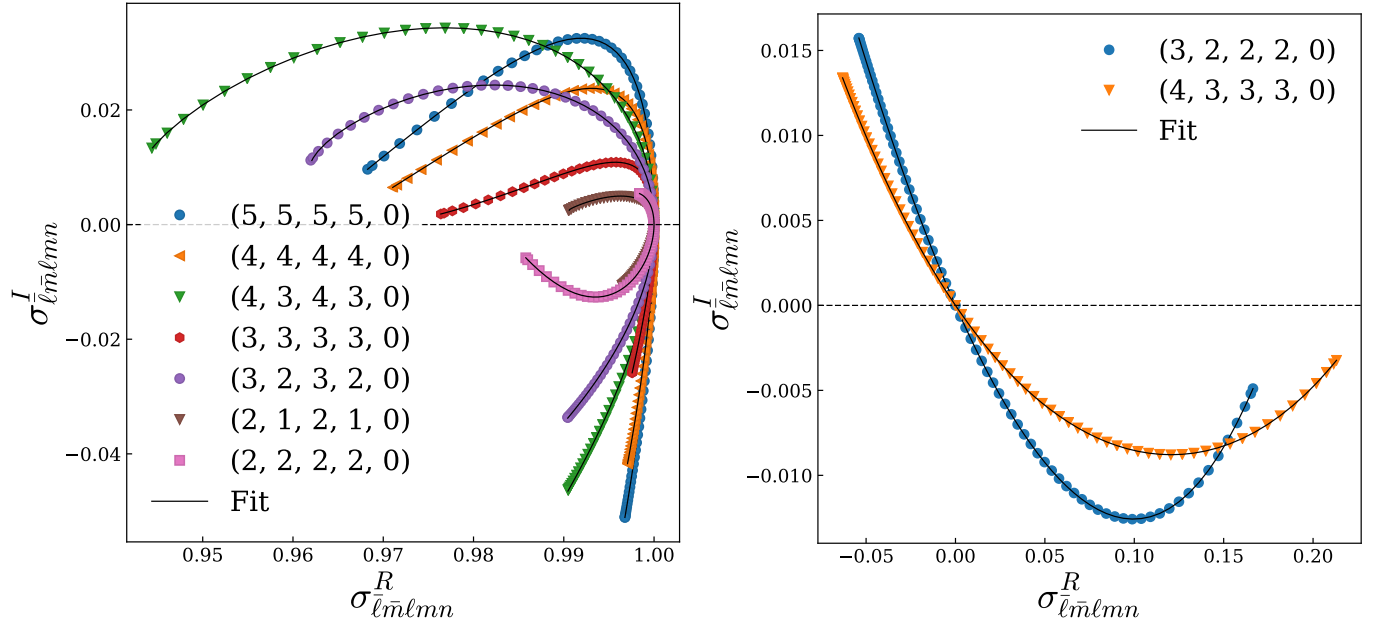


FIG. 2. Spherical-spheroidal harmonic mixing coefficients.