Angular velocity of gravitational radiation from precessing binaries and the corotating frame

Michael Boyle

Phys. Rev. D **87**, 104006 — Published 3 May 2013

DOI: 10.1103/PhysRevD.87.104006

Angular velocity of gravitational radiation from precessing binaries and the corotating frame

Michael Boyle
Center for Radiophysics and Space Research, Cornell University, Ithaca, New York 14853, USA

This paper defines an angular velocity for time-dependent functions on the sphere, and applies it to gravitational waveforms from compact binaries. Because it is geometrically meaningful and has a clear physical motivation, the angular velocity is uniquely useful in helping to solve an important—and largely ignored—problem in models of compact binaries: the inverse problem of deducing the physical parameters of a system from the gravitational waves alone. It is also used to define the corotating frame of the waveform. When decomposed in this frame, the waveform has no *rotational* dynamics and is therefore as slowly evolving as possible. The resulting simplifications lead to straightforward methods for accurately comparing waveforms and constructing hybrids. As formulated in this paper, the methods can be applied robustly to both precessing and nonprecessing waveforms, providing a clear, comprehensive, and consistent framework for waveform analysis. Explicit implementations of all these methods are provided in accompanying computer code.

I. INTRODUCTION

Gravitational-wave astronomy stands on the brink of delivering numerous observations of merging compact binaries [1–6]. Though the uncertainties are large, black-hole binaries involving large spins are expected to constitute a significant fraction of observable events [7–10]. If these spins are misaligned with the orbital angular velocity, the system will precess, imprinting the gravitational radiation with strong variations [11–14]. While it is not clear how common such misalignment will actually be, it is entirely clear that we will need good models of the precessing waveforms if we hope to accurately measure them.

We can describe the motion of a precessing binary on very short timescales as a simple orbit in a plane; on longer timescales, that plane rotates. Now, we know that the gravitational-wave field of a nonprecessing system can be decomposed into relatively simple modes when the orbital plane is orthogonal to the z axis [15–17]. But precession moves the orbital plane out of alignment, causing the modes to mix and leading to complex behaviors which complicate analysis of the waveforms [18–29]. In particular, none of the methods developed to analyze nonprecessing systems will work correctly with precessing systems.

In the context of post-Newtonian models, Buonanno, Chen, and Vallisneri [18] proposed a convention whereby effects of precession can be isolated from orbital motion. Specifically, the system is analyzed at each instant in a frame with its z axis orthogonal to the orbital plane; from moment to moment, the frame is made to rotate to follow the precession. This method was later rediscovered in the context of numerical relativity, and techniques were developed for finding such a frame from the waveform itself in a geometrically meaningful way [30–32].

This paper extends previous work by developing a frame in which *all* rotational behavior is eliminated, simplifying the waveform as much as possible, and allowing direct generalizations of methods for analyzing nonprecessing systems. In the process, the angular velocity of a waveform is introduced, which also has important uses, such as supplying a partial solution to an important inverse problem.

A. The modeler's inverse problem

We might distinguish two significant inverse problems related to gravitational waves: the *modeler's* inverse problem and the equally important *astronomer's* inverse problem. The gravitational-wave astronomer's task is to deduce the parameters (masses, spins, etc.) of a system from observations at a single point over an extended time. In practice, it is greatly complicated by the presence of noise in the data. Usually referred to as parameter estimation, this problem has been extensively studied [33–44]. The modeler's task, on the other hand, is to deduce the parameters given observations of the entire sphere at infinity over a brief (possibly infinitesimal) interval of time. It is—in some sense—prior to the astronomer's problem, because it addresses the *meaning* of the parameters in models astronomers use.[1] This paper concerns itself exclusively with the modeler's inverse problem.

Various methods exist for producing gravitational waveforms—numerical-relativity, phenomenological, post-Newtonian, and effective-one-body models, for example. But no one of these is capable of producing an accurate and complete waveform on its own. Numerical-relativity (NR) simulations are too expensive to simulate more than a short portion of the waveform near merger. Phenomenological models use NR data as inputs. Post-Newtonian (PN) approximations break down before the merger. Even terms generating the effective-one-body (EOB) inspiral and the ad hoc method of attaching a ringdown must be "calibrated" by comparison to numerical results. Therefore, we need more than one model to generate a complete waveform, which means that we need to understand precisely how the different models relate to each other. This leads directly to the inverse problem.

The numbers we plug in to a computation of initial data for an NR simulation bear no clear relation to the numbers we insert into a PN or an EOB computation—or even to other NR simulations using different formulations. For example, the direction of a black-hole spin measured in the arbitrary coordinates of NR initial data need not correspond in any meaningful way to the direction measured in PN coordinates.

[1] Intriguingly, understanding the modeler's inverse problem may help to inform the astronomer's inverse problem more directly [29].

Even if the gauge *condition* used for the numerical simulation were the same as the one used to derive the PN formulas, the initial data would not be the same, so the gauge itself would be different. For nonprecessing systems, symmetries reduce the ambiguity to one of simple time and phase offsets, and numerous simple methods have been suggested to resolve those ambiguities [45]. But in the precessing case, we need to be much more careful. Simply using the same numbers in two different models leads to comparing systems with inherently different physics.

Fundamentally, we need to establish a mapping between the input parameters of different models such that they produce the same physics (as nearly as possible) during some span of time for which both models are valid. Because we have no access to any invariant physical meaning behind our parameters, we need to take a different approach. For example, given some particular set of parameters, we can run a numerical simulation. Then, we can work backwards from the resulting waveform and try to find the parameters needed to generate the same waveform with a PN system—which is the inverse problem.

The issue of ascribing meaningful physical interpretations to geometric quantities measured on $\mathscr{I}^+$ has been investigated to some extent [46], but there is a distinct lack of practical methods suited to analysis of gravitational waves. The angular-velocity vector introduced by this paper and a related vector introduced by O'Shaughnessy *et al*. [31] provide geometrically meaningful physical quantities which can be measured directly from the waveforms alone, and are thus prime candidates for use in solving the inverse problem. Indeed, we will see in Sec. III that these two vectors are very closely related to input parameters for the precessing PN system. This provides a partial solution to the inverse problem, leaving three remaining degrees of freedom. Several possibilities will also be suggested for completing the solution of the inverse problem, though they are beyond the scope of this paper.

B. Overview of this paper

Section II introduces the angular velocity of a waveform, finding a straightforward formula and a more intuitive interpretation of the mathematics. This and the related vector suggested in Ref. [31] are then used in Sec. III to find a partial solution to the inverse problem. A PN waveform is used as a test case, showing excellent agreement between the original parameters and the parameters deduced from the waveform alone. In Sec. IV, the angular velocity is used to determine a frame with that velocity. The same PN waveform used in the previous section is decomposed in this frame, showing that the amplitudes of the waveform modes become very simple, and their phases become nearly constant. Because this frame reduces the complexity of the waveform, it is ideally suited to practical manipulation of waveforms, as discussed in Sec. IV C. It is also worth noting that the partial solution to the inverse problem completely establishes all extrinsic parameters, giving us a solid foundation for comparisons between waveforms. Finally, the results are summarized and suggestions for future work are collected in Sec. V.

The appendices provide deeper background information which may be useful for implementing these methods or comparing to other methods. Appendix A presents a fairly comprehensive discussion of quaternions and various related details, including several new results. In Appendix B, formulas are derived for the rotation of arbitrary spin-weighted functions. While equivalent formulas have been derived previously [32, 47, 48], this derivation uses a somewhat different technique, and carefully develops conventions for consistency throughout this paper. In any case, the upshot is that modes of spin-weighted fields transform exactly as do modes of spin-weight-zero fields. Lastly, Appendix C discusses related previous work in the same formalism used in this paper allowing for more direct comparisons.

Ancillary files included with this paper (available on the paper's arXiv page) contain computer code implementing all of the concepts introduced here, among others. The core functions are written in C++ [49] for speed, using several functions from the GNU Scientific Library [50]. While this code could be incorporated directly into other C/C++ codes, an additional user interface is provided in Python [51, 52] code as the `GWFrames` module, which simply exposes all the C++ functions through Python. Documentation and examples can be found among the ancillary files. Relevant functions or classes are mentioned where appropriate throughout this paper.

C. Quaternion notation

The techniques of this paper necessarily involve rotations, which are best implemented in terms of the group of unit quaternions because of the numerous advantages over direct manipulation of rotation matrices or Euler-angle coordinates. By using quaternions, we obtain robust methods which can be blindly applied to general systems, including nonprecessing ones—which simplifies the processing of large numbers of waveforms. Moreover, once the basics are understood, quaternion rotations are more intuitive than either of those inferior descriptions. In fact, quaternions are essentially the axis-angle description of rotations, in a more practical guise. Therefore, quaternion notation will be used throughout. As mentioned above, Appendix A provides a thorough introduction to quaternions, while computer code included in ancillary files with this paper gives practical implementations of the necessary functionality through `GWFrames.Quaternion`. However, such details are not necessary for a good understanding of this paper; the following paragraph should provide sufficient background.

Quaternions can be thought of as generalizing the familiar complex numbers, where the imaginary part is generalized to a three-dimensional vector part.[2] We can write a quaternion as the sum of a scalar and a vector: $\mathbf{Q} = q_0 + \boldsymbol{q}$. The conjugate of the quaternion is $\bar{\mathbf{Q}} = q_0 - \boldsymbol{q}$. We can multiply quaternions together [see Eq. (A2)], the product being associative but not commutative in general. The norm of a quaternion is defined according to $|\mathbf{Q}|^2 = \mathbf{Q}\bar{\mathbf{Q}} = q_0^2 + \boldsymbol{q}\cdot\boldsymbol{q}$. Unit quaternions,

[2] In fact, both complex numbers and quaternions are special cases of geometric algebra [53], done in 2 and 3 dimensions, respectively. Much of our intuition from complex arithmetic transfers easily to quaternion arithmetic when $\mathrm{i} = \sqrt{-1}$ is replaced by a unit vector. The notable exception to this correspondence is noncommutativity of the quaternion product.

having norm $|\mathbf{R}| = 1$, are especially important, as they describe rotations. To see this, we can consider a vector to be a quaternion with scalar component equal to zero, in which case it makes sense to multiply a vector by a quaternion. Then we can define the transformation

$$\boldsymbol{v} \mapsto \boldsymbol{v}' \coloneqq \mathbf{R}\,\boldsymbol{v}\,\bar{\mathbf{R}}\,. \tag{1}$$

A simple exercise shows that this transformation is linear, and preserves lengths and orientations, so it is just a rotation. Ultimately, the best reason to use quaternions is the existence of simple formulas [Eqs. (A8) and (A9)] for the exponential and logarithm, which prove to be endlessly useful. In particular, we can express an arbitrary unit quaternion as $\mathbf{R} = \mathrm{e}^{\theta\,\hat{\boldsymbol{u}}/2} = \cos\frac{\theta}{2} + \hat{\boldsymbol{u}}\,\sin\frac{\theta}{2}$, where exponentiation of a quaternion is defined by the usual power series.[3] It turns out that this $\mathbf{R}$ produces a rotation through the angle θ about the axis $\hat{\boldsymbol{u}}$. Because *any* rotation may be expressed in this form, we will use unit quaternions as our only representation of rotations, and refer to them as *rotors*. A frame will be described by the rotor that generates it by rotating some standard basis frame.

II. ANGULAR VELOCITY OF A WAVEFORM

We can define the angular velocity of a gravitational waveform—or any field on a sphere—as the opposite of the velocity of the counter-rotation needed to keep the field as constant as possible. In the first part of this section, this definition will be formulated more precisely, resulting in a surprisingly simple formula for the angular velocity. The formula can be interpreted as a projection of the familiar operator equation $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L} = \partial_t$ onto the "rotational parts" of the waveform—a notion which can be made surprisingly rigorous using the language of Hilbert spaces, as discussed in the second part of this section.

A. Finding the angular velocity

The essential idea here is to remove the rotational behavior of the waveform by imposing a rotation that eliminates as much of the time dependence as possible. Suppose that $\mathscr{R}_j(t)$ is a time-dependent rotation operator acting on the wave field f (usually representing Ψ_4 or h) such that $\mathscr{R}_j(t_j) = 1$. We wish to find the rotation operator that—in some sense—minimizes the quantity

$$\frac{\partial}{\partial t}\Big[\mathscr{R}_j(t)\,f(t;\vartheta,\varphi)\Big]\Big|_{t=t_j}\,. \tag{2}$$

Clearly, this is a complex function of position on the sphere. To reduce it to a single real number, we take its squared magnitude and integrate over the sphere:

$$\Xi(\mathscr{R}_j) \coloneqq \int_{S^2}\left|\frac{\partial}{\partial t}\Big[\mathscr{R}_j(t)\,f(t;\vartheta,\varphi)\Big]\Big|_{t=t_j}\right|^2 \mathrm{d}\Omega\,. \tag{3}$$

[3] Note the striking—and not coincidental—similarity to Euler's formula with $\hat{\boldsymbol{u}}$ in place of the unit imaginary i. This results from the fact that, under quaternion multiplication, $\hat{\boldsymbol{u}}\,\hat{\boldsymbol{u}} = -1$.

If we expand the field f in spin-weighted spherical harmonics (SWSHs, discussed in Appendix B), the natural way to express the rotation operator is in its usual form $\mathscr{R}_j = \exp[-\mathrm{i}\,\boldsymbol{\theta}_j\cdot\boldsymbol{L}]$, where $\boldsymbol{L}$ is the standard angular-momentum operator and $\boldsymbol{\theta}_j$ is the time-dependent axis-angle description of the rotation. Note that we must have $\boldsymbol{\theta}_j(t_j) = 0$ because we have assumed that $\mathscr{R}_j(t_j) = 1$. This is absolutely crucial because it makes the differentiation in Eq. (3) tractable. We also define the angular velocity[4]

$$\boldsymbol{\omega} \coloneqq -\partial_t\boldsymbol{\theta}_j\,, \tag{4}$$

where the negative sign arises because $\boldsymbol{\theta}_j$ corresponds to the rotation needed to keep the field fixed in a moving frame, whereas $\boldsymbol{\omega}$ is intended to describe the motion of the field relative to the initial static frame. We have

$$\Xi(\boldsymbol{\omega}) = \int_{S^2}|\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}\,f + \partial_t f|^2\,\mathrm{d}\Omega\,. \tag{5}$$

Now, we can write the integral in terms of a sum over standard matrix elements of the angular-momentum operator and the problem simplifies nicely. We obtain

$$\Xi = \boldsymbol{\omega}\cdot\langle\boldsymbol{L}\boldsymbol{L}\rangle\cdot\boldsymbol{\omega} + 2\,\boldsymbol{\omega}\cdot\langle\boldsymbol{L}\,\partial_t\rangle + \sum_{\ell,m}\left|\partial_t f^{\ell,m}\right|^2\,. \tag{6}$$

where we have defined the matrix[5]

$$\langle\boldsymbol{L}\boldsymbol{L}\rangle^{ab} \coloneqq \sum_{\ell,m,m'}\bar{f}^{\ell,m'}\,\langle\ell,m'|L^{(a}\,L^{b)}|\ell,m\rangle\;f^{\ell,m}\,, \tag{7a}$$

and the vector

$$\langle\boldsymbol{L}\,\partial_t\rangle^a \coloneqq \sum_{\ell,m,m'}\Im\left[\bar{f}^{\ell,m'}\,\langle\ell,m'|L^a|\ell,m\rangle\;\partial_t f^{\ell,m}\right]\,. \tag{7b}$$

Noting that the last term in Eq. (6) is independent of $\boldsymbol{\omega}$, we can find the minimum[6] of Ξ analytically:

$$\boldsymbol{\omega} = -\langle\boldsymbol{L}\boldsymbol{L}\rangle^{-1}\cdot\langle\boldsymbol{L}\,\partial_t\rangle\,. \tag{7c}$$

[4] Subscripts are necessary on the rotation operator $\mathscr{R}_j$ and the associated vector $\boldsymbol{\theta}_j$ because these have certain properties depending on which instant of time t_j we are looking at. We have implicitly assumed that the $\boldsymbol{\theta}_j(t)$ are all related by simple constant offsets, as necessary to satisfy the conditions $\boldsymbol{\theta}_j(t_j) = 0$. Because the offsets are *constant*, the angular-velocity vector $\boldsymbol{\omega}$ does not have such a dependence, and so does not need the subscript.

[5] Here, the $|\ell,m\rangle$ represent the *spin-weighted* eigenfunctions, but the angular-momentum operator acts on these just as in the non-spin-weighted case [47], making this notation particularly familiar. The matrix denoted here as $\langle\boldsymbol{L}\boldsymbol{L}\rangle^{ab}$ is precisely the quantity $\langle L_{(a}\,L_{b)}\rangle_t$ defined by O'Shaughnessy *et al*. [31], except that the latter is normalized by $\sum_{\ell,m}|f^{\ell,m}|^2$.

[6] We can show that it is a true minimum rather than a more general stationary point by looking at the Hessian matrix of Ξ, which is just $2\,\langle\boldsymbol{L}\boldsymbol{L}\rangle$. We are free to rotate this matrix into a frame in which its dominant principal axis is along $\hat{\mathsf{z}}$. Then, we can calculate its eigenvalues as $\langle L_{\mathsf{z}}^2\rangle$ and $\langle L^2 - L_{\mathsf{z}}^2\rangle \pm \left|\langle L_+^2\rangle\right|$. As long as some mode with $m \neq 0$ is nonzero, these are always (strictly) positive. Hence, $\langle\boldsymbol{L}\boldsymbol{L}\rangle$ is positive definite, and we have a true minimum. Furthermore, we can calculate that the determinant is actually the product of these eigenvalues, and thus is also nonzero whenever the field is nonzero, allowing us to invert the matrix in Eq. (7c). Since $\langle\boldsymbol{L}\boldsymbol{L}\rangle$ is a geometric object, and eigenvalues and determinants are invariant under rotations, these conclusions hold in all frames.

The effects of L^a are familiar, so this may be directly computed from knowledge of $f^{\ell,m}(t)$, with no optimization or solution of the eigensystem necessary. There is no ambiguity in the direction of the angular velocity, and we obtain a meaningful magnitude.

In the computer code included among this paper's ancillary files, a waveform object may be constructed with `GWFrames.Waveform`. The angular velocity may then be found using the `AngularVelocityVector` method on such an object.

B. Interpreting the mathematics

Equation (7) gives a formula for the angular-velocity vector of the field f. Though it takes a relatively simple form, the reason it takes this particular form may seem somewhat opaque. In fact, it really has quite a simple interpretation, which may be instructive. In fact, we can start off with a simple observation and re-derive Eq. (7) in a very different way.

The angular-momentum operator $\boldsymbol{L}$ generates rotations, as is well known. So, for example, $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}\,f$ gives the time rate of change of the field under a simple rotation given by $\boldsymbol{\omega}$. More generally, the three components of $-\mathrm{i}\,\boldsymbol{L}\,f$ form a basis generating the Hilbert subspace Λ, consisting of functions describing possible rates of change for f under (complex) rotations. On the other hand, we also have a second operator ∂_t, which gives the actual time rate of change of the field, whether that change is a simple rotation or a change in amplitude—or a more complicated behavior. But we can extract the part of $\partial_t f$ caused by (real) rotation alone by projecting onto the basis vectors of Λ and taking the real part:

$$\Re\left[\int_{S^2} \overline{-\mathrm{i}\,\boldsymbol{L}\,f}\;\partial_t f\,\mathrm{d}\Omega\right] . \tag{8}$$

The three components of this expression completely describe the rotational part of $\partial_t f$. We take the real part because we ordinarily take the dot product of $-\mathrm{i}\,\boldsymbol{L}$ with a real-valued vector, so if we expect to find such terms in $\partial_t f$, they must have real components [54].

Now, the crucial point: if $\boldsymbol{\omega}$ correctly describes the rotation, the same projection of $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}\,f$ must give the same result:

$$\Re\left[\int_{S^2} \overline{-\mathrm{i}\,\boldsymbol{L}\,f}\;(-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}\,f)\,\mathrm{d}\Omega\right] = \Re\left[\int_{S^2} \overline{-\mathrm{i}\,\boldsymbol{L}\,f}\;\partial_t f\,\mathrm{d}\Omega\right] . \tag{9}$$

If we expand f in spin-weighted spherical harmonics, it turns out[7] that this equation reduces to precisely

$$\langle \boldsymbol{L}\boldsymbol{L}\rangle\cdot\boldsymbol{\omega} = -\langle \boldsymbol{L}\,\partial_t\rangle \ , \tag{10}$$

which is, of course, equivalent to Eq. (7c).

[7] As usual, we get integrals of the form $\int \ldots |\vartheta,\varphi\rangle\langle\vartheta,\varphi|\ldots\,\mathrm{d}\Omega$, which are just resolutions of the identity. Then, taking the real part on the left-hand side is equivalent to symmetrizing over the indices of the two $\boldsymbol{L}$ vectors before contracting with $\boldsymbol{\omega}$. On the right-hand side, taking the real part of i times a quantity is the same as taking the negative imaginary part, so this is precisely the definition of $-\langle \boldsymbol{L}\,\partial_t\rangle$.

Thus, we see the interpretation clearly. In the case of a pure rotation, we have $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}\,f = \partial_t f$. In general, however, we have to project onto the rotational parts of the waveform for equality to hold, which is just what Eqs. (7) and (10) do. It is also worth noting that in the purely rotational case, we can use $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L} = \partial_t$ directly and calculate $\Xi \equiv 0$. Recalling the definition of Ξ in Eq. (3), this says that the time variation is completely eliminated.

Interestingly, we can see this projection working directly by showing that $\langle \boldsymbol{L}\boldsymbol{L}\rangle$ and $\langle \boldsymbol{L}\,\partial_t\rangle$ are insensitive to changes in the amplitudes of the modes. Clearly, $\langle \boldsymbol{L}\boldsymbol{L}\rangle$ does not depend on any derivatives with respect to time. To see that $\langle \boldsymbol{L}\,\partial_t\rangle$ is insensitive to changing amplitude, we first note that it is a geometric object so we can evaluate it in any frame we choose—we choose a frame in which it is aligned with the z axis. Next, we decompose the field into (logarithmic) amplitude and phase parts:

$$f^{\ell,m}(t) = \exp\left[\chi^{\ell,m}(t) + \mathrm{i}\,\phi^{\ell,m}(t)\right] . \tag{11}$$

A pure rotation about the z axis leads to $\dot\chi^{\ell,m} = 0$ and $\dot\phi^{\ell,m} = -m\,|\boldsymbol{\omega}|$, so we expect that a projection onto the rotational part will eliminate $\dot\chi^{\ell,m}$ but must not eliminate $\dot\phi^{\ell,m}$. In fact, we can explicitly calculate

$$\langle \boldsymbol{L}\,\partial_t\rangle = \hat{\boldsymbol{z}}\sum_{\ell,m}\Im\left[\bar f^{\ell,m}\,\langle\ell,m|L^z|\ell,m\rangle\,\partial_t f^{\ell,m}\right] \tag{12a}$$

$$= \hat{\boldsymbol{z}}\sum_{\ell,m}\Im\left[\left(\dot\chi^{\ell,m} + \mathrm{i}\,\dot\phi^{\ell,m}\right) m\,\left|f^{\ell,m}\right|^2\right] \tag{12b}$$

$$= \hat{\boldsymbol{z}}\sum_{\ell,m} m\,\dot\phi^{\ell,m}\,\left|f^{\ell,m}\right|^2 , \tag{12c}$$

Here, taking the imaginary part has caused $\dot\chi^{\ell,m}$ to drop out entirely, leaving only $\dot\phi^{\ell,m}$, supporting the claim that we have removed non-rotational parts of the waveform. Note that this formula is entirely general; we have not assumed any particular behavior of χ or ϕ, for example.

III. SOLVING THE INVERSE PROBLEM

Section I A established the need to solve the inverse problem. Essentially, in order to create a complete gravitational waveform, we need to be able to take a finite or even infinitesimal portion of a waveform and infer the parameters of the PN (or similar) system that result in that waveform. Because it is the most extensively developed system, we will discuss the quasicircular PN model as a concrete example. In this section, we will first describe the parameters that need to be established. This will involve reviewing the basic elements of the PN model. We will then see how to solve part of the inverse problem using the angular-velocity vector $\boldsymbol{\omega}$ and the dominant eigenvector of $\langle \boldsymbol{L}\boldsymbol{L}\rangle$ (denoted $\hat{\boldsymbol{V}}_f$) [31], showing the effectiveness of this method with an example.

A. The required parameters

To the extent that different formulations of the PN model are correct, they predict the same physics, and so we are free to choose between them as we wish. Certain formulations may be

better than others with regard to solving the inverse problem. Here, we follow Refs. [24, 55]. First, we assume a pair of particles with masses M_1 and M_2, and spins $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. The unit vector pointing from the second to the first is $\hat{\boldsymbol{n}}$. The orbital angular velocity is defined as

$$\boldsymbol{\Omega}_{\rm orb} \coloneqq \hat{\boldsymbol{n}} \times \dot{\hat{\boldsymbol{n}}} \,. \tag{13a}$$

The direction of this vector is frequently expressed in the literature as $\hat{\boldsymbol{L}}_{\rm N} (\equiv \hat{\boldsymbol{\Omega}}_{\rm orb})$. There is (in general) an additional rotation of the system due to precession, denoted $\boldsymbol{\Omega}_{\rm prec}$. Now, if this vector were to have any component orthogonal to $\hat{\boldsymbol{n}}$, that would contradict the definition of $\boldsymbol{\Omega}_{\rm orb}$, so it must simply be proportional to $\hat{\boldsymbol{n}}$:[8]

$$\boldsymbol{\Omega}_{\rm prec} \coloneqq \Omega_{\rm prec}\, \hat{\boldsymbol{n}} \,. \tag{13b}$$

We also define the sum of these:

$$\boldsymbol{\Omega}_{\rm tot} \coloneqq \boldsymbol{\Omega}_{\rm orb} + \boldsymbol{\Omega}_{\rm prec} \,. \tag{13c}$$

During the evolution we must record the minimal-rotation frame[9] aligned with $\boldsymbol{\Omega}_{\rm orb}$ and the accumulated orbital phase $\Phi_{\rm orb}$ measured relative to $\hat{\boldsymbol{n}}$. Then, the frame of the binary will given by rotating the minimal-rotation frame by $\Phi_{\rm orb}$ about its z axis. These are the orbital elements of the system. Their evolution is not of particular concern here, as the details have no effect on our conclusions. The waveform can be calculated in this frame using standard formulas, and transformed to an inertial frame if needed to complete the construction of the waveform.

The initial data we need to begin a PN calculation, then, are the values of $(M_1, M_2, \boldsymbol{S}_1, \boldsymbol{S}_2, \boldsymbol{\Omega}_{\rm orb}, \hat{\boldsymbol{n}})$ at some initial time. These might be termed the *intrinsic* parameters of the system [18, 56, 57]. They are geometrically meaningful, and covariant under certain symmetries assumed for our system—namely time translation and rotation of coordinates. But this brings up a subtlety. We can think of two more classes of parameters: the *extrinsic* and the *fiducial*. Extrinsic parameters depend on the observer, and can be thought of generally as degrees of gauge freedom like the time offset or an overall rotation. Fiducial parameters are selected values of intrinsic quantities that depend on extrinsic parameters. By solving for the intrinsic parameters relative to a particular time function and a particular basis for the vectors, we will be tacitly setting the extrinsic parameters. Then, when comparing two waveforms, we must choose fiducial parameters and ensure that the extrinsic parameters are the same. We will find that it is a simple matter to ascertain the intrinsic parameters except for three degrees of freedom in the directions of the spin vectors. It will also be straightforward to completely establish the extrinsic parameters.

B. Deducing the parameters

We might only expect quantities to be meaningful if they are covariant objects measured at infinity—e.g., waveforms or ADM-type quantities. Coordinate locations of black holes in a simulation, for example, depend too much on details of gauge conditions and vagaries of initial data and junk radiation to be of any real use. On the other hand, some quantities are also reasonably well defined when the black holes are very widely separated. Therefore, if we find that certain quantities change slowly during the early part of the NR simulation, and are not expected to have changed much previously, then we might also be able to use those quantities in our analysis. This is typically true of the masses and spin magnitudes when measured appropriately [58–60], except to the extent that they are expected to change [61, 62]. Therefore, we assume that M_1, M_2, $|\boldsymbol{S}_1|$, and $|\boldsymbol{S}_2|$ can be measured in the simulation and used directly. The rest of our intrinsic parameters will come from the waveforms (or possibly other measurements on $\mathscr{I}^+$).

To see how we can derive orbital elements from quantities observable from the waveform, we need to see how orbital elements give rise to the waveform. Familiar calculations [15, 16] tell us that the PN waveform is created by motion of the binary. The complete motion is described by $\boldsymbol{\Omega}_{\rm tot}$, so we expect that $\boldsymbol{\omega}$ should be the same. On the other hand, the component along $\hat{\boldsymbol{n}}$ does not lead to changing multipole moments (to our level of approximation). So only the component of $\boldsymbol{\Omega}_{\rm tot}$ orthogonal to $\hat{\boldsymbol{n}}$ is involved—but that is precisely $\boldsymbol{\Omega}_{\rm orb}$. We can therefore expect that the waveform is oriented along this vector, in some sense. Now, the vector $\boldsymbol{z}$ happens to be the dominant eigenvector of $\langle \boldsymbol{L}\boldsymbol{L} \rangle$ for an individual spin-weighted spherical harmonic (though not necessarily for a combination of them). It also happens to be the dominant eigenvector when the field f is symmetric under reflection through the x–y plane [63]—as the PN waveform is in the frame aligned with $\boldsymbol{\Omega}_{\rm orb}$. Therefore, we should expect $\boldsymbol{\Omega}_{\rm orb}$ to be parallel to $\hat{\boldsymbol{V}}_f$.

Putting these considerations together, we can expect the following approximate equalities:

$$\boldsymbol{\Omega}_{\rm tot} \simeq \boldsymbol{\omega} \,, \tag{14a}$$

$$\boldsymbol{\Omega}_{\rm orb} \simeq \left(\hat{\boldsymbol{V}}_f \cdot \boldsymbol{\omega}\right) \hat{\boldsymbol{V}}_f \,, \tag{14b}$$

$$\boldsymbol{\Omega}_{\rm prec} \simeq \boldsymbol{\omega} - \left(\hat{\boldsymbol{V}}_f \cdot \boldsymbol{\omega}\right) \hat{\boldsymbol{V}}_f \,. \tag{14c}$$

Inspection of the PN model suggests that these expressions should become more exactly true in the asymptotic limit of low orbital velocities. Note that Eq. (13b) shows that $\hat{\boldsymbol{n}}$ is along $\boldsymbol{\Omega}_{\rm prec}$, so we effectively obtain that quantity as well, whenever the precession is nonzero.

Figure 1 compares the orbital elements to the related waveform expressions, for a PN system with significant precession. The direction of $\boldsymbol{\Omega}_{\rm orb}$ coincides extremely well with $\hat{\boldsymbol{V}}_f$—they agree to within the numerical precision throughout the inspiral. $\boldsymbol{\Omega}_{\rm tot}$ and $\boldsymbol{\omega}$ agree to within a few parts in 10^5 early in the inspiral, though the disagreement grows near merger. However, it may be possible to remove even this disagreement through more careful treatment of the distinction between the orbital phase and the phase of a waveform in PN theory.

Now, in each case of Eqs. (14), the quantities on the right-hand side are measured directly from the waveform. Thus, if we have a numerical waveform, we can simply measure the right-hand sides and *define* the "PN-equivalent" orbital elements according to these equations. A PN system given

[8] Note that for other formulations of the PN model, this equation may not be true. See Refs. [11, 12, 18–25] for more details.

[9] See Sec. C 4 and Ref. [32].

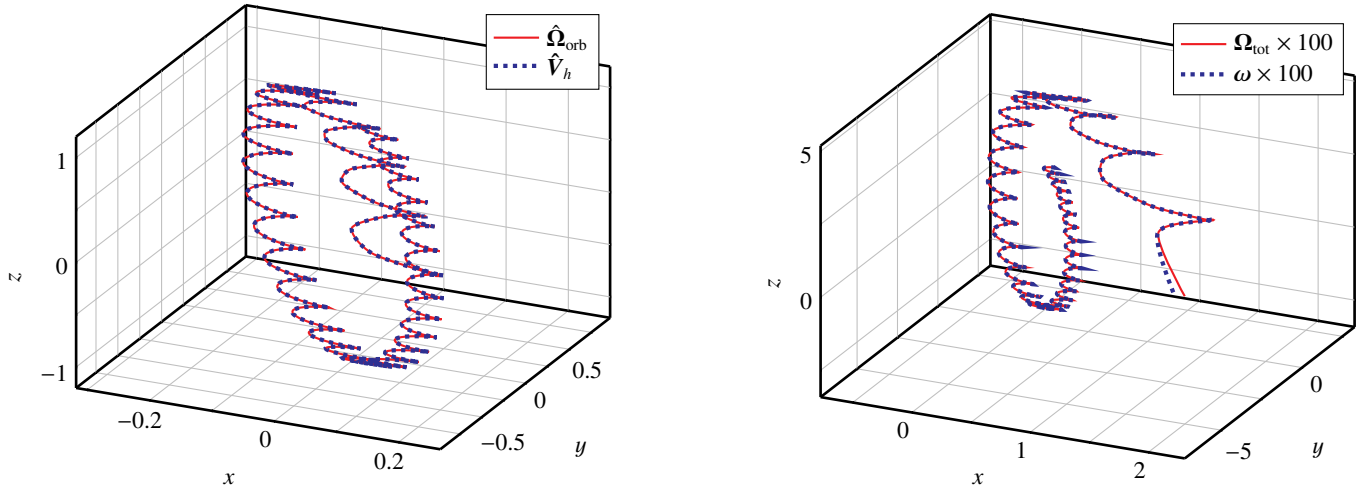

FIG. 1. **Orbital elements compared to waveform quantities**. These plots show the PN orbital elements $\hat{\mathbf{\Omega}}_{\rm orb}$ (left) and $\mathbf{\Omega}_{\rm tot}$ (right), compared to the "PN-equivalent" quantities derived from the waveforms alone given in Eqs. (14). The binary has a 6 : 1 mass ratio. Initially, the larger black hole has a spin of $S_1/M_1^2 = 0.9$ in the $(\vartheta, \varphi) = (2.00, 0.25)$ direction; the smaller black hole has a spin of $S_2/M_2^2 = 0.3$ in the $(\vartheta, \varphi) = (2.4, 2.9)$ direction. These parameters were chosen because the resulting orbital velocity happens to execute a complete flip, passing very close to $-\hat{\mathbf{z}}$, which is a rigorous test of these methods. In both plots, tighter oscillations correspond to earlier times; the last 3600 M before merger are shown. The directions $\hat{\mathbf{\Omega}}_{\rm orb}$ and $\hat{V}_h$ are identical to within the numerical accuracy throughout the inspiral. The vectors $\mathbf{\Omega}_{\rm tot}$ and $\boldsymbol{\omega}$ are the same to within a few parts in 10^5 early in the inspiral, though differences grow somewhat as the system approaches merger (roughly the end of the data shown here).

those parameters as initial conditions will necessarily be as similar to the numerical system as possible—at least by the measures of $\boldsymbol{\omega}$ and $\hat{\boldsymbol{V}}_f$.

This gives us a partial solution to the inverse problem. We are lacking four degrees of freedom corresponding to the directions of the two spin vectors. One additional piece of information is also available from the foregoing considerations. The magnitude $\Omega_{\rm prec}$ is given in PN theory as a bilinear function of $\boldsymbol{S}_1 \cdot \hat{\boldsymbol{n}}$ and $\boldsymbol{S}_2 \cdot \hat{\boldsymbol{n}}$, where the coefficients depend on the PN-expansion parameter $v := (M\,\Omega_{\rm orb})^{1/3}$, and are therefore already known. Thus, we can solve for $\boldsymbol{S}_1 \cdot \hat{\boldsymbol{n}}$, for example. This suggests other possible methods to find the remaining three components of spin. For example, PN expressions for the orbital angular *momentum* $\boldsymbol{L}$ are available in terms of the orbital elements and the various projections of the spin vectors. If it is practical to measure the total angular momentum of the spacetime $\boldsymbol{J}$ in the numerical solution [64, 65], we could then use the PN expression for $\boldsymbol{L} + \boldsymbol{S}$ to solve for the PN-equivalent components of spin. This would complete the solution of the inverse problem.

Alternatively, we might measure various modes of the waveform and equate them to the PN expressions for those modes. Again, these expressions contain various known quantities, as well as bilinear combinations of $\boldsymbol{S}_1 \cdot \hat{\mathbf{\Omega}}_{\rm orb}$ and $\boldsymbol{S}_2 \cdot \hat{\mathbf{\Omega}}_{\rm orb}$ [13, 66]. Therefore, we could solve for these combinations of the spin components. Seemingly, this would rely on the accuracy of the PN expressions, which is not very high for spin terms. On the other hand, the influence of any errors that result would be similarly diminished. One final degree of freedom would remain in this example, and would have to be fixed by other means. In any case, we leave these considerations to future work.

In the computer code included among this paper's ancillary files, a waveform object may be constructed with `GWFrames.Waveform`, or a PN waveform may be constructed with `GWFrames.PNWaveform`. The PN-equivalent orbital and precessional angular velocities may then be calculated using the `PNEquivalentOrbitalAV` and `PNEquivalentPrecessionalAV` methods.

IV. THE COROTATING FRAME

So far, we have calculated only the instantaneous angular velocity of the waveform, $\boldsymbol{\omega}$ [Eq. (7c)]. While this has already proven useful in the previous section, it can also be advantageous in determining a frame in which to decompose the waveform. Specifically, we seek a frame whose angular velocity is just $\boldsymbol{\omega}$. When decomposed in this frame, the waveform will have no rotation, and will be as constant as possible. The frame compares favorably to other frames introduced previously [30–32] (see Appendix C). It has practical benefits and suggests simple techniques for measuring, comparing, and processing waveforms from numerical simulations.

A. Finding the corotating frame

Our task here is to find the rotor $\mathbf{R}(t)$ describing a frame whose angular velocity is $\boldsymbol{\omega}$. We can relate the two by a simple equation:

$$\boldsymbol{\omega} = 2\,\dot{\mathbf{R}}\,\bar{\mathbf{R}}\,. \tag{15}$$

(See Ref. [32] and Sec. A 3.) Unfortunately, the solution we might naively write down is wrong:

$$\mathbf{R}(t) \neq \exp\left[\frac{1}{2}\int^{t} \boldsymbol{\omega}(t')\,\mathrm{d}t'\right], \tag{16}$$

except when the system is nonprecessing. Ultimately, the reason for the failure of this formula in general is that $\boldsymbol{\omega}$ is not parallel to its derivative (or integral). In the language of quaternions,[10] $\boldsymbol{\omega}$ fails to commute with its derivative (or integral); to find the correct version of Eq. (16), we need to account for that noncommutativity.

Given $\boldsymbol{\omega}$, we could solve Eq. (15) for $\dot{\mathbf{R}}$ and integrate as we would a vector equation. But in practice this would quickly violate the constraint that $\mathbf{R}$ should be a unit quaternion. Instead, we will need an expression in terms of the logarithm of this rotor:

$$\dot{\mathbf{R}}\,\bar{\mathbf{R}} = \dot{\mathfrak{r}} + \frac{\sin^2|\mathfrak{r}|}{2\,|\mathfrak{r}|^2}\,[\mathfrak{r},\dot{\mathfrak{r}}] + \frac{|\mathfrak{r}| - \sin|\mathfrak{r}|\,\cos|\mathfrak{r}|}{4\,|\mathfrak{r}|^3}\,[\mathfrak{r},[\mathfrak{r},\dot{\mathfrak{r}}]]\,, \tag{17}$$

with $\mathfrak{r}(t) \coloneqq \log \mathbf{R}(t)$. (See Sec. A 3 for the derivation.) The second and third terms on the right-hand side of this expression account for the noncommutativity as needed. Setting the right-hand side equal to $\boldsymbol{\omega}/2$, we can solve for $\dot{\mathfrak{r}}$ to find

$$\dot{\mathfrak{r}} = \left(\boldsymbol{\omega} - \frac{\mathfrak{r}\,(\mathfrak{r}\cdot\boldsymbol{\omega})}{|\mathfrak{r}|^2}\right)\frac{|\mathfrak{r}|\,\cot|\mathfrak{r}|}{2} + \frac{\mathfrak{r}\,(\mathfrak{r}\cdot\boldsymbol{\omega})}{2\,|\mathfrak{r}|^2} + \frac{1}{2}\boldsymbol{\omega}\times\mathfrak{r}\,. \tag{18a}$$

This is just an explicit first-order ordinary differential equation, so we can integrate numerically using standard techniques to arrive at the appropriate $\mathfrak{r}(t)$ and find the corotating frame

$$\mathbf{R}_{\mathrm{corot}}(t) = \exp\left[\mathfrak{r}(t)\right]\,. \tag{18b}$$

The advantage of this method over direct integration of Eq. (15) is that it ensures that the resulting quaternion truly does have norm 1. When integrated directly, the quaternion in Eq. (15) has four degrees of freedom, whereas a unit quaternion has only three. By integrating Eq. (18a) instead, we eliminate the extra degree of freedom, reducing this to a truly three-dimensional problem while automatically satisfying the constraint on the norm. In general, transforming equations in such a way improves the accuracy of numerical results significantly—as is certainly the case with this system when tested.

Naturally, imposing a condition on the angular velocity of a frame leaves its overall orientation free. Assuming $\mathbf{R}_{\mathrm{corot}}(t)$ describes a frame whose angular velocity is $\boldsymbol{\omega}$, then the frame $\mathbf{R}_{\mathrm{corot}}(t)\,\mathbf{R}_{\mathrm{c}}$ will have the same angular velocity for any constant $\mathbf{R}_{\mathrm{c}}$. Alternatively, the frame $\mathbf{R}_{\mathrm{c}}\,\mathbf{R}_{\mathrm{corot}}(t)$ would have angular velocity $\boldsymbol{\omega}$ rotated by $\mathbf{R}_{\mathrm{c}}$. In the interests of simplifying the waveform, it is best to choose some particular time during the inspiral at which to align the z axis of the frame with $\hat{\boldsymbol{V}}_f$, as suggested by O'Shaughnessy *et al*. [31]. Once this is done at one instant of time, z and $\hat{\boldsymbol{V}}_f$ should be aligned at all other times to very high accuracy. We are still free to rotate about the z axis, so we can set the phase of the $(\ell, m) = (2, 2)$ mode to 0 at this instant, for example. We will see below that the phase is very slowly varying in the corotating frame, so this will not be a delicate operation. Alternatively, if the waveform is precessing, we can align the x axis with the PN-equivalent $\hat{\boldsymbol{n}}$, which should be roughly equivalent to setting the $(2, 2)$ phase to 0. When comparing two waveforms, the only requirement is that these instants of time be comparable, which is assured by choosing a common fiducial quantity, as discussed in Sec. III A.

In the computer code included among this paper's ancillary files, the function `GWFrames.FrameFromAngularVelocity` returns the corotating frame, given an array of quaternions representing the angular-velocity vector as a function of time. Alternatively, a waveform object may be constructed with `GWFrames.Waveform` and transformed to the corotating frame with the `TransformToCorotatingFrame` method.

B. Gravitational waveforms in the corotating frame

Figure 2 demonstrates the effects of decomposing the PN waveform described in Fig. 1 in various frames. First is the usual inertial frame, where a stationary observer at infinity has constant coordinate position. In this frame, the moduli of the modes oscillate wildly, as power shifts between them (upper left panel of Fig. 2). Similarly, the phase (upper right panel) shows strange features. The $\ell = \pm 2$ and $\ell = \pm 1$ modes have roughly the same frequency, as power from the dominant modes leaks into and overwhelms the $\ell = \pm 1$ modes. Those phases also change direction each time the rotation axis passes through the x–y plane (at times of roughly -2400 and -600). Both the modulus and phase are very complicated functions in this frame. They would be hard to model directly, and their rapid variations are not conducive to accurate numerics.

The next pair of panels shows the waveform in the waveform-aligned frame suggested by O'Shaughnessy *et al*. [31] supplemented with the minimal-rotation condition (see Sec. C 4 and Ref. [32]). Here, $\langle \boldsymbol{LL} \rangle$ is evaluated using all modes up to $\ell = 8$, the dominant eigenvector is found, and the fame is rotated so that its z axis coincides with that eigenvector while obeying the minimal-rotation condition. This drastically simplifies both the modulus and phase, as seen in the middle panels of Fig. 2. The modulus is very smoothly sweeping up as the binary spirals in toward merger. The phases are now separated as usual, with slopes more nearly equal to $-m\,\Omega_{\mathrm{orb}}$.

Decomposing the waveform in the waveform-aligned frame also requires recording the orientation of the frame. In that case, there is no additional overhead in going to the corotating frame, shown in the lower panels of Fig. 2. The modulus plot is identical to the one in the previous case. However, there is further improvement in the phase, with each mode having nearly constant phase throughout. Similar behavior is seen in other modes with $\ell > 2$. Naturally, such a waveform is particularly well suited to interpolation and hybridization [26, 32, 70].

A minor feature to note in the phase is the non-constancy of the $(2, \pm 1)$ modes. Considered on their own, these variations could be removed by a rotation because the curves change in

[10] Note that the failure to commute is by no means specific to the quaternion description of rotations; it is a feature of rotations themselves. Quaternions do, however, provide a very effective means of solving the problem.

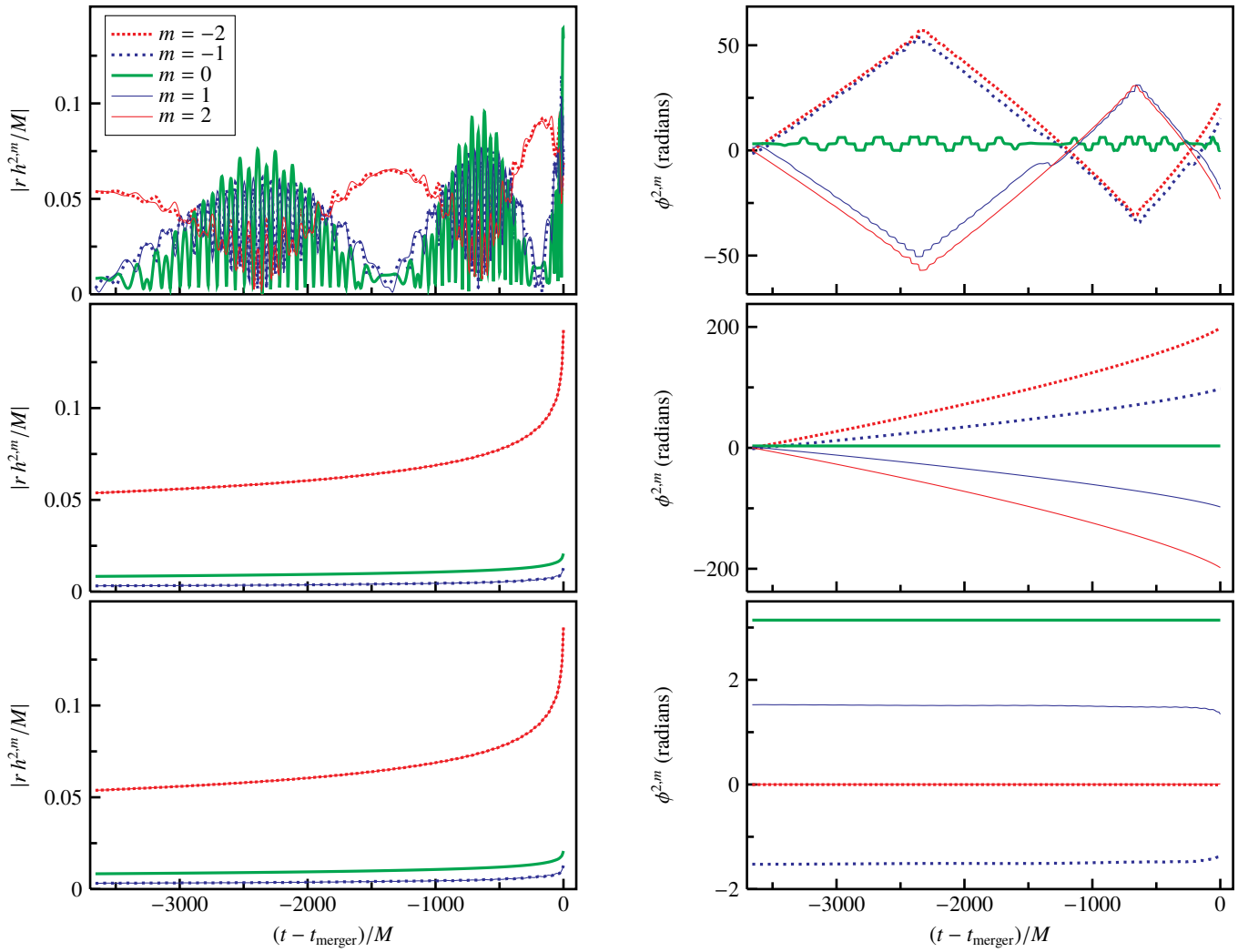

FIG. 2. **Precessing waveform in various frames**. These plots show the modulus (left) and phase (right) of the $\ell = 2$ modes of a post-Newtonian waveform in the inertial (top), waveform-aligned minimal-rotation (middle), and corotating (bottom) frames. The phase is defined as usual [67–69] so that $h^{\ell,m} = |h^{\ell,m}| \exp[\mathrm{i}\,\phi^{\ell,m}]$, with branch-cut discontinuities removed. The system is the same as the one shown in Fig. 1. Going from the inertial frame to the aligned frame drastically simplifies the waveform amplitudes and significantly simplifies the phase. In the aligned frame, the waveform looks very much like a nonprecessing waveform [26]. Expressing the waveform in the corotating frame retains the smoothness in amplitude seen in the aligned frame, but makes the phases of the modes nearly constant, with values of roughly 0, $\pm\pi/2$, and π. Similar results can also be seen for modes with $\ell > 2$.

opposite directions. On the other hand, this would cause the phases of other modes to vary. Equation (7c) automatically balances these concerns; the amplitudes of the $(2, \pm 1)$ modes are so small that they do not carry much weight. By transforming to the corotating frame, we isolate the waveform's intrinsic dynamics—seen here in $\phi^{2,\pm1}$—from the rotational dynamics of the system, allowing for separate analyses. This is important because they are separately modeled, so it is useful to be able to inspect each effect on its own.

C. Extrapolation, comparison, alignment, and hybridization

As a practical matter, we need to manipulate numerical waveforms in various ways. We must eliminate physical and gauge effects associated with extraction of data at finite radius, usually by extrapolation [71–73].[11] To compare numerical waveforms to each other or to analytical waveforms, we need to determine the extrinsic parameters corresponding to freedom in choosing the zero of time and the overall orientation of our axes. To construct a complete waveform, we may occasionally need to hybridize waveforms from different systems [26, 32, 70, 77–80]. Various approaches to these problems have been introduced,

[11] Cauchy-characteristic extraction is another method of finding the correct waveform at $\mathscr{I}^+$ [74–76]. This can proceed as usual, and the final waveform can be transformed to the corotating frame. A recent implementation [76] has improved the efficiency to make this an especially attractive alternative to extrapolation.

but most are designed exclusively for nonprecessing systems, or are otherwise incomplete or fragile. The angular velocity and corotating frame provide excellent tools for addressing these issues more generally, and are especially robust when implemented by means of quaternions.

Practical extrapolation relies on smoothness of the extrapolated functions [73]. In the corotating frame, the modes of the waveform are essentially constant during inspiral—except for the overall growth in modulus—suggesting that this is the ideal frame for extrapolation. We can implement such a procedure in the usual way, with one minor addition. We first impose a time-retardation offset to all the data, as usual. Then, we add the step of finding the corotating frame of the outermost extracted data, and transforming the data at all radii to that frame. (We cannot rotate data at each radius into its own corotating frame, as that would require extrapolation of rotors, which is not well understood.) The waveforms at the other radii will not be precisely in their own corotating frames, but should be close enough that the data are quite smooth. The extrapolation may then proceed as usual, resulting in an extrapolated waveform in the corotating frame of the outermost data. Again, the extrapolated result will not be precisely its own corotating frame, but the transformation is routine.

In the nonprecessing case, a multitude of methods have been suggested to compare and align waveforms (fixing the extrinsic parameters) and to construct hybrids of NR and PN waveforms [45]. These all require generalization to use in the case of significant precession. By using the corotating frame, we simplify the modes of the waveform decomposition sufficiently that ordinary methods can still be used. (Comparing phase differences or relative differences in modulus, for example.) But each waveform now comes with its own frame, $\mathbf{R}_A(t)$ and $\mathbf{R}_B(t)$, encoding most of the phase dynamics, so we will also need to compare the difference between the frames themselves. Fortunately, quaternions provide us with *geometrically meaningful* measures of the difference.

The difference itself is given in quaternion form as[12]

$$\mathbf{R}_\Delta(t) := \mathbf{R}_A(t)\,\bar{\mathbf{R}}_B(t)\,. \tag{19a}$$

This is the rotation taking frame B into frame A, and is independent of the basis with respect to which A and B are defined. Now, we might want to know how "big" this difference rotation is. As noted in Sec. I C, we can write any rotation, including $\mathbf{R}_\Delta$, in axis-angle form:

$$\mathbf{R}_\Delta(t) = \exp\left[\mathbf{\Phi}_\Delta/2\right]\,. \tag{19b}$$

We can easily solve this equation for $\mathbf{\Phi}_\Delta$ by taking the logarithm. In particular, its magnitude is the angle through which the system must be rotated:

$$\Phi_\Delta(t) = 2\left|\log \mathbf{R}_\Delta(t)\right|\,. \tag{19c}$$

This can be used as a simple but complete description of the phase difference between two systems.[13]

We can understand this better and make contact with previous work by recognizing $\mathbf{\Phi}_\Delta$ as a more general version of a common measure of the difference between waveforms common in analysis of nonprecessing systems. That measure is $\Delta\phi^{\ell,m}$, the difference between the phases of the modes as measured in the static frame. Because the angular velocity is conventionally chosen to be along the z axis, we can usually relate the orbital phase $\Phi_{\rm orb}$ to the waveform phase as $\phi^{\ell,m} \approx -m\,\Phi_{\rm orb}$. Here, we have the similar expression

$$\Delta\phi^{\ell,m} \approx -m\,\Phi_\Delta\,. \tag{20}$$

However, as we saw in the previous section, $\Delta\phi^{\ell,m}$ is a less useful measure for precessing systems. In contrast, $\mathbf{\Phi}_\Delta$ encapsulates the differences in both the orbital and the precessional dynamics[14] in one convenient function while leaving the waveform dynamics separate, and is equally relevant in both precessing and nonprecessing systems.

The quantity Φ_Δ gives us a compact description of the difference between two waveforms as measured in their corotating frames. But it depends on the extrinsic parameters discussed in Sec. III A: the overall time offset and orientation of the static basis frame. Alignment of waveforms consists of minimizing differences between the waveforms at some instant or over some span of time by adjusting the extrinsic parameters as needed. This can be seen as a restricted version of the inverse problem, where we simply assume that the intrinsic parameters are identical—as when we wish to compare waveforms evolved the same initial data with different numerical resolution. Section III B used the implicit assumption that the time coordinate and basis frame would be defined to be the same in both waveforms. Here, we are simply given two waveforms, along with their arbitrary time offsets and overall orientations. They must be aligned more actively.

We can separate this into two steps: first align the time, then align the frames. To align the time, we will need some measure of the waveform that is independent of orientation. For example, we can use the magnitude of the angular velocity $|\boldsymbol{\omega}|$. We then choose some fiducial time $t_{\rm fid}$ and find the value of δt such that

$$|\boldsymbol{\omega}_A(t_{\rm fid})| = |\boldsymbol{\omega}_B(t_{\rm fid} + \delta t)|\,. \tag{21}$$

The time coordinates of waveform B may then be shifted as $t \mapsto t + \delta t$. The main limitation with this method is that the magnitude $|\boldsymbol{\omega}|$ is not always strictly monotonic for highly precessing systems. Usually it is possible to find a time for which it is monotonic. Alternatively, we can find δt by minimizing the squared difference between the two sides of Eq. (21) integrated over some significant span of time. Quantities other than $|\boldsymbol{\omega}|$ may also be more robust against this

[12] The inverse of an arbitrary nonzero quaternion $\mathbf{Q}$ is just $\bar{\mathbf{Q}}/|\mathbf{Q}|^2$. Since rotors have norm 1, the inverse of a rotor $\mathbf{R}$ is just $\bar{\mathbf{R}}$. Therefore, this formula is analogous to subtraction, but applied to rotation operators.

[13] It is crucial to note that $\log(\mathbf{R}_A\,\bar{\mathbf{R}}_B) \neq \log\mathbf{R}_A - \log\mathbf{R}_B$ because rotations do not commute. In particular, the latter depends on the basis frame, and is therefore not a useful measure of the difference between frames.

[14] If needed, the orbital evolution can be further isolated from the precessional dynamics using the minimal-rotation frame.

non-monotonicity—quantities such as flux or the total power in the waveform.

Now, assuming that the time coordinates have been properly aligned, it is a simple matter to align the frames. We simply apply the transformation $\mathbf{R}_B(t) \mapsto \mathbf{R}_\Delta(t_{\rm fid})\,\mathbf{R}_B(t)$. Then, at $t_{\rm fid}$, there will be precisely no difference between the frames. In particular, $\Phi_\Delta(t_{\rm fid}) = 0$. Reference [32] suggested essentially this same transformation, but included an additional rotation about the z axis because there was still one degree of rotational freedom in that paper. Here, we have assumed that the orientation of each waveform has been completely fixed at $t_{\rm fid}$, as discussed in Sec. IV A, though not necessarily fixed to the same basis frame.

In previous work, alignment has also been done by minimizing the squared difference in some quantity ($\Delta\phi^{2,2}$, for example) integrated over some span of time. This has been used in an effort to nullify spurious effects such as junk radiation or residual eccentricity [79, 80]. A similar program could certainly be applied to Φ_Δ, and would require simultaneously optimizing over the time offset and all three degrees of rotational freedom. In particular, a simplification that occurs in the nonprecessing case and allows the problem to be reduced to one dimension will not work in the precessing case due to noncommutativity of rotations; the problem must remain truly four-dimensional.

Once the waveforms are aligned, it is a simple matter to hybridize them with a slight generalization of the standard method. Typically, we use only information from waveform A before some time t_1, and only information from waveform B after some t_2, with a transition in between. The waveforms are assumed to have been aligned somewhere in the range between t_1 and t_2. The transition may be accomplished with some (usually smooth) monotonic function $\tau(t)$ that equals 1 before t_1 and 0 after t_2. Then, the hybrid version of any mode $f^{\ell,m}$ may be defined as a simple linear interpolation between the two waveforms:[15]

$$f^{\ell,m}_{\rm hybrid} \coloneqq f^{\ell,m}_A(t)\,\tau(t) + f^{\ell,m}_B(t)\,[1-\tau(t)]\,. \tag{22a}$$

Again, however, each waveform comes with its own frame, and these frames have to be hybridized. As suggested in Ref. [32], this can be accomplished with a form of linear interpolation defined for rotors:

$$\mathbf{R}_{\rm hybrid} \coloneqq L\Big(\tau(t); \mathbf{R}_A(t), \mathbf{R}_B(t)\Big)\,, \tag{22b}$$

where the interpolant L is given by Eq. (A30). As discussed in Appendix A 4, it is critically important to use the correct interpolant. Finally, we must note we only have the frames $\mathbf{R}_A$ and $\mathbf{R}_B$ sampled at discrete (essentially arbitrary) points, so we will need to interpolate between those points to the desired t. For this, smoother interpolation is required. Appendix A 4 discusses a method using cubic splines reinterpreted for rotors.

[15] In previous work, this formula is usually applied separately to the phase and modulus of the mode. With this new description, there seems to be no advantage in decomposing the waveform in this way. The formula given here is written assuming complex mode data.

In the computer code included among this paper's ancillary files, waveform objects may be constructed with `GWFrames.Waveform`. They may be aligned, compared, and hybridized with methods such as `AlignTime`, `AlignFrame`, `AlignTimeAndFrame`, `Compare`, and `Hybridize`.

V. CONCLUSIONS

The angular velocity of a waveform was defined in Sec. II by the rotation which minimizes the time dependence of the waveform. This fairly nebulous criterion was reformulated precisely, and led to a simple formula, providing us with a geometrically meaningful description of the motion of a waveform. We also saw that $\boldsymbol{\omega}$ can be considered to be that vector which makes the action of the operator $-\mathrm{i}\,\boldsymbol{\omega}\cdot\boldsymbol{L}$ as equal as possible to the action of ∂_t, in a sense that can also be made surprisingly precise.

The angular-velocity vector and the dominant eigenvector $\hat{\boldsymbol{V}}_f$ of $\langle \boldsymbol{LL}\rangle$ proposed by O'Shaughnessy *et al*. [31] provide us with powerful tools to understand and manipulate waveforms, with no reference to meaningless gauge quantities. Section III showed that these two vectors can be used very effectively and accurately to find at least part of the solution to the important inverse problem. Determining the three remaining degrees of freedom is left for future work, though some suggestions were made for how to do this.

Beyond this fundamental benefit, $\boldsymbol{\omega}$ also provides key practical advantages. We can readily calculate the corotating frame, which also has angular velocity $\boldsymbol{\omega}$. Transformed to this frame, the waveform is literally as constant as possible. When functions are slowly varying, they are easily approximated by low-order functions; they can be numerically interpolated and differentiated quite accurately; and fewer data points are needed to record their values than for quickly varying functions. This type of technique has already seen great success in numerical simulations themselves [81, 82].

Putting these together, we can also perform all the standard manipulations needed for waveform analysis. Data collected from a simulation at different radii can be extrapolated nicely. Two waveforms (e.g., different resolutions of a numerical simulation, or an NR and a PN waveform) can be aligned, compared, and hybridized readily. The only additional steps necessary are comparison and hybridization of the frames, but these are easily achieved using formulas given by Eqs. (19a) and (22b). Notably, the use of quaternions vastly improves numerics and allows us to access the geometrically meaningful elements of rotations.

The code included with this paper implements all the techniques discussed above, showing that they are ready to use in waveform analysis. There are, however, issues that may benefit from further investigation. As mentioned, more work is needed to complete the solution of the inverse problem. Also, it is certainly possible that different techniques could further simplify the ringdown, for example. While preliminary results show that reasonable, smooth results for $\boldsymbol{\omega}$ and $\hat{\boldsymbol{V}}_f$ are obtained throughout the inspiral, merger, and ringdown, the waveform in any frame still has very complicated structure during ringdown, presumably stemming from the difference

between spin-weighted *spheroidal* harmonics and spherical ones [83, 84]. More specific methods [29, 84, 85] will likely be needed to adequately capture features of general ringdowns with simple models.

Nonetheless, we can conclude that $\boldsymbol{\omega}$ and $\hat{\boldsymbol{V}}_f$ already deliver a complete system for waveform analysis. When implemented with quaternion methods, the system is robust enough to be applied blindly to both precessing and nonprecessing systems. This consistency simplifies the production and analysis of both types of waveform.

ACKNOWLEDGMENTS

It is my great pleasure to thank Larry Kidder, Abdul Mroué, Evan Ochsner, Richard O'Shaughnessy, Sergei Ossokine, Robert Owen, Harald Pfeiffer, Christian Reisswig, and Saul Teukolsky for useful conversations and comments on earlier drafts of this paper. I also appreciate the hospitality of the Kavli Institute for Theoretical Physics at UC Santa Barbara during the early stages of this work. This project was supported in part by a grant from the Sherman Fairchild Foundation; by NSF Grants No. PHY-0969111, No. PHY-1005426, No. PHY11-25915, and No. PHY11-25915; and by NASA Grant No. NNX09AF96G.

Appendix A: Quaternions and rotations

Unit quaternions clearly constitute the representation of choice when computing with spatial rotations. Quaternions have become the dominant technique in fields as diverse as computer graphics, robotics, molecular dynamics, navigation, and orbital mechanics. They are closely related to the axis–angle formalism, which gives us clear geometric intuition and avoids the problem of gimbal lock associated with singularities of the Euler angles. But there is also a clear notion of them as operators, giving us all the advantages of the matrix representation of rotations. They are trivially inverted and easily composed, and the logarithm and exponential functions are easy to evaluate, presenting further advantages over all other representations.

For all these reasons, this paper and the accompanying code use the notation of quaternions. Here, the basic elements of quaternion math are summarized, Wigner's $\mathfrak{D}$ matrices and the spin-weighted spherical harmonics are expressed directly in terms of quaternions, and formulas for the linear interpolants and splines of rotors are given. The computer code included among this paper's ancillary files contains all of the quaternion functions discussed here. The fundamental object is the `GWFrames.Quaternion`, which has numerous methods. See the documentation for more details. Also, note that `Mathematica` [86] returns incorrect results for logarithms of general quaternions, and is thus not a reliable tool for most of the calculations in this paper.

1. Elements of quaternion mathematics

A quaternion is a set of four numbers, usually denoted as

$$\mathbf{Q} = (q_0, q_1, q_2, q_3) = q_0 + \boldsymbol{q} \,. \tag{A1}$$

TABLE I. Quaternion notation

$\mathbf{Q}$	Quaternion
q_α	Component α of the quaternion
$\bar{\mathbf{Q}}$	Conjugate: $(q_0, -q_1, -q_2, -q_3)$
$\lvert\mathbf{Q}\rvert$	Norm: $\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$
$\boldsymbol{q}$	Vector part: (q_1, q_2, q_3)
q	Magnitude of vector part: $\sqrt{q_1^2 + q_2^2 + q_3^2}$
$\hat{\boldsymbol{q}}$	Normalized vector part: $\boldsymbol{q}/q$
$\mathfrak{q}$	Logarithm: $\log \mathbf{Q}$
$\mathfrak{q}$	Magnitude of the logarithm: $\lvert\mathfrak{q}\rvert$
$\operatorname{ang}\mathbf{R}$	Angle of a rotation: $2\,\mathfrak{r} = 2\,\lvert\log\mathbf{R}\rvert$

We summarize the notation in Table I. The quaternions form an algebra, meaning that the quaternions form a vector space (over the real numbers), as well as a group where the product is defined by

$$\mathbf{P}\,\mathbf{Q} = (p_0\, q_0 - \boldsymbol{p}\cdot\boldsymbol{q}) + (p_0\,\boldsymbol{q} + q_0\,\boldsymbol{p} + \boldsymbol{p}\times\boldsymbol{q}) \,. \tag{A2}$$

Here, the dot product and cross product of vectors take their usual meanings. Note that this product is neither commutative nor anti-commutative in general. The conjugate of a quaternion is defined as

$$\bar{\mathbf{Q}} \coloneqq (q_0, -q_1, -q_2, -q_3) = q_0 - \boldsymbol{q} \,, \tag{A3}$$

and the norm of a quaternion according to

$$|\mathbf{Q}|^2 = \mathbf{Q}\,\bar{\mathbf{Q}} = q_0^2 + q_1^2 + q_2^2 + q_3^2 = q_0^2 + \boldsymbol{q}\cdot\boldsymbol{q} \,. \tag{A4}$$

A unit quaternion is simply a quaternion with unit norm. Because quaternion multiplication is associative, we can find useful inverses of a quaternion by taking the conjugate and dividing by the squared norm:

$$\mathbf{Q}^{-1} = \frac{\bar{\mathbf{Q}}}{|\mathbf{Q}|^2} \,. \tag{A5}$$

In particular, the inverse of a unit quaternion is just its conjugate. Note, however, that while a unit quaternion has *norm* $|\mathbf{R}| = 1$, its *square* is not 1 in general. For example, if $\mathbf{R} = \hat{\boldsymbol{u}}$ is some unit vector, we have $\mathbf{R}^2 = -1$.

Now, given any vector $\boldsymbol{v}$, we can define the transformation law

$$\boldsymbol{v}' = \mathbf{R}\,\boldsymbol{v}\,\bar{\mathbf{R}} \,, \tag{A6}$$

where the right-hand side involves quaternion multiplication with $\boldsymbol{v}$ interpreted as a quaternion with scalar part $v_0 = 0$. It is not hard to check that if $\mathbf{R}$ has unit magnitude, then this transformation law preserves orientation, angles, and lengths—and is therefore a rotation. These rotations compose in the natural way, and we will see below that we can construct a unit quaternion representing any desired rotation, which means that the unit quaternions form a representation of the rotation group.[16]

[16] In fact, because of the double-sided rotation law, Eq. (A6), $\mathbf{R}$ and $-\mathbf{R}$

Using the product law for quaternions, we can define the exponential of a quaternion according to the standard power series:

$$\exp \mathbf{Q} \coloneqq \sum_{n=0}^{\infty} \frac{\mathbf{Q}^n}{n!} \,. \tag{A7}$$

Note that, because of the non-commutativity of quaternion multiplication, the usual rules of exponents do not apply. In particular, $\exp[\mathbf{P} + \mathbf{Q}] \neq \exp \mathbf{P} \exp \mathbf{Q}$ unless $\mathbf{P}$ and $\mathbf{Q}$ commute—which happens precisely when their vector parts are parallel. Given some angle θ and some unit vector $\hat{\boldsymbol{u}}$, we can show that the unit quaternion[17]

$$\mathbf{R} = \exp\left[\frac{\theta}{2}\,\hat{\boldsymbol{u}}\right] = \cos\frac{\theta}{2} + \hat{\boldsymbol{u}}\,\sin\frac{\theta}{2} \tag{A8}$$

represents a rotation through the angle θ about the axis $\hat{\boldsymbol{v}}$ (in the positive sense, using the right-hand rule). This illustrates the connection between the axis–angle and the unit-quaternion representations of rotation. The factor of $1/2$ needed in the exponential is a result of the double-sided rotation law, Eq. (A6).

By inspection of Eq. (A8), we see that we can also define a reasonable logarithm of nonzero quaternions:[18]

$$\log \mathbf{Q} \coloneqq \log|\mathbf{Q}| + \frac{\boldsymbol{q}}{q} \arctan\frac{q}{q_0} \,. \tag{A9}$$

Note that the logarithm of a unit quaternion will be a pure vector—$\log|\mathbf{Q}| = 0$. For compactness, we define the notation $\mathfrak{q} \coloneqq \log \mathbf{Q}$ and $\mathfrak{q} \coloneqq |\mathfrak{q}|$. As with the usual complex logarithm and the real arctangent function, this function is multivalued; the magnitude of the vector part is ambiguous up to integer multiples of 2π. We typically choose the principal value so that the norm of the vector part is in $[0, \pi]$, as with the complex logarithm. Choosing the branch must be done carefully, in order to obtain correct geometric results and reasonably continuous functions of time. When differentiating the logarithm (as in Sec. A 3 for example), we will treat the function as being continuous. On the other hand, sometimes in the very same formula, we will assume the logarithm takes on its principal value.

The principal value of the quaternion logarithm can actually be restricted further if our purpose is only to cover $SO(3)$. Because of the double-sided rotation law of Eq. (A6), the final vector is invariant under $\mathbf{R} \mapsto -\mathbf{R}$. This is equivalent to

$$\mathfrak{r} \mapsto \frac{\mathfrak{r} - \pi}{\mathfrak{r}}\,\mathfrak{r} \,. \tag{A10}$$

represent the same rotation, so the unit quaternions provide a double cover of the rotation group $SO(3)$; the group of unit quaternions is actually isomorphic to $SU(2)$. Unsurprisingly, the logarithms [defined in Eq. (A9)] of unit quaternions form a group isomorphic to $\mathfrak{su}(2)$. Hence the notation $\mathfrak{r} = \log \mathbf{R}$.

[17] More generally, the exponential of any quaternion is $\exp \mathbf{Q} = \exp|\mathbf{Q}| \exp(\mathbf{Q}/|\mathbf{Q}|)$, where the second factor can be evaluated according to the given formula.

[18] Again, this expression generalizes the more familiar complex relation $\log z = \log|z| + \mathrm{i} \arctan \Im z/\Re z$, where i is replaced by a general three-vector.

In particular, we can apply this formula when $\mathfrak{r} > \pi/2$, ensuring that $\mathfrak{r} \in [0, \pi/2]$. The transformation gives rise to different *rotors*, but the same *rotation*.

As with exponents of real numbers, we can define

$$\mathbf{Q}^{\mathbf{P}} \coloneqq \exp[\mathbf{P} \log \mathbf{Q}] \,. \tag{A11}$$

This formula will be usually be applied in cases where $\mathbf{P}$ is a pure real number, though other formulas may be advantageous in such cases—as illustrated in the case of the square-root below.

The square root of a quaternion is particularly useful in constructing rotations taking one vector into another as directly as possible. We can find a formula for it with an elegant geometric interpretation and important numerical advantages over Eq. (A11) with $\mathbf{P} = 1/2$. The product of two unit vectors $-\hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}}$ is a rotation in the $\hat{\boldsymbol{u}}$–$\hat{\boldsymbol{w}}$ plane of twice the angle between those vectors, in the sense from $\hat{\boldsymbol{w}}$ to $\hat{\boldsymbol{u}}$. The square root of this product is the same rotation through only half that angle—in particular, $\sqrt{-\hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}}}$ is the most direct rotation taking $\hat{\boldsymbol{w}}$ into $\hat{\boldsymbol{u}}$. We need to bisect the angle between them, and a familiar geometric construction that achieves this is the diagonal of the rhombus having $\hat{\boldsymbol{w}}$ and $\hat{\boldsymbol{u}}$ as sides:

$$\hat{\boldsymbol{v}} = \frac{\hat{\boldsymbol{u}} + \hat{\boldsymbol{w}}}{|\hat{\boldsymbol{u}} + \hat{\boldsymbol{w}}|} \,. \tag{A12}$$

Then the rotation we want is

$$\sqrt{-\hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}}} = \pm\,\hat{\boldsymbol{v}}\,\hat{\boldsymbol{w}} = \pm\frac{\hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}} - 1}{|\hat{\boldsymbol{u}} + \hat{\boldsymbol{w}}|} = \pm\frac{1 - \hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}}}{\sqrt{2[1 - (\hat{\boldsymbol{u}}\,\hat{\boldsymbol{w}})_0]}} \,. \tag{A13}$$

Computing the square root using this expression is easier than using Eq. (A11), in the sense that no transcendental functions are required and fewer singularities are encountered. This expression is very robust and deals well with finite numerical precision. This expression is ill defined whenever $\hat{\boldsymbol{u}} + \hat{\boldsymbol{w}} = 0$—which is not surprising, as there are infinitely many "shortest" ways to rotate a vector into its opposite. These are two ways of expressing the fact that there are infinitely many square roots of -1 among the unit quaternions.

2. Formulas for rotations of SWSHs

We now express Wigner's $\mathfrak{D}$ matrices and the spin-weighted spherical harmonics (SWSHs) directly in terms of quaternions, so that no conversion to or from the more usual Euler-angle representation is necessary. In the following, we will treat the general case in which the spin weight s is arbitrary; the formulas given here do not assume $s = \pm 2$.

The SWSHs form a basis for spin-weighted functions on the sphere [87–89]. Goldberg *et al*. [47] showed that the SWSHs can be expressed as special cases of Wigner's $\mathfrak{D}$ matrices, so that by constructing $\mathfrak{D}^{(\ell)}_{m',m}$, we will obtain ${}_sY_{\ell,m}$. Defining the parts of the quaternion $\mathbf{Q}$ as[19]

$$Q_a \coloneqq q_0 + \mathrm{i}\,q_3 \quad \text{and} \quad Q_b \coloneqq q_2 + \mathrm{i}\,q_1 \,, \tag{A14}$$

[19] The choices of signs in these definitions are—to a great extent—arbitrary

$$\mathfrak{D}^{(\ell)}_{m',m}(\mathbf{R}) = \begin{cases} \delta_{m',-m}\, R_b^{-2m}\, (-1)^{\ell+m} & \text{when } R_a = 0, \\ \delta_{m',m}\, R_a^{+2m} & \text{when } R_b = 0, \\ \sqrt{\frac{(\ell+m')!\,(\ell-m')!}{(\ell+m)!\,(\ell-m)!}}\; |R_a|^{2\ell-2m'}\; R_a^{m'+m}\, R_b^{m'-m}\; \sum_\rho (-1)^\rho \binom{\ell+m}{\rho} \binom{\ell-m}{\ell-\rho-m'} \left(\frac{|R_b|}{|R_a|}\right)^{2\rho} & \text{otherwise.} \end{cases} \tag{A16}$$

This expression is valid for all integral and half-integral values of $\ell \geq 0$; naturally, we only need integral values $\ell \geq 2$ for the $s = -2$ fields discussed in this paper. To recover the usual expressions for $\mathfrak{D}$ in terms of Euler angles, we use $\mathbf{R} = \mathrm{e}^{\alpha\,\hat{\mathbf{z}}/2}\, \mathrm{e}^{\beta\,\hat{\mathbf{y}}/2}\, \mathrm{e}^{\gamma\,\hat{\mathbf{z}}/2}$, from which we can easily find

$$R_a = \cos\frac{\beta}{2}\, \mathrm{e}^{\mathrm{i}\,\frac{\gamma+\alpha}{2}} \qquad R_b = \sin\frac{\beta}{2}\, \mathrm{e}^{\mathrm{i}\,\frac{\gamma-\alpha}{2}}\,. \tag{A17a}$$

It must be emphasized, of course, that evaluating Eq. (A16) directly is faster and deals with numerical-precision issues better than using the form with sines and cosines.

Now, to express the SWSHs in terms of these $\mathfrak{D}$ matrices, we adopt conventions to agree with Ref. [91], which attempts to establish uniform conventions for use in numerical relativity. We have

$${}_sY_{\ell,m}(\vartheta,\varphi) = (-1)^s\, \sqrt{\frac{2\,\ell+1}{4\,\pi}}\, \mathfrak{D}^{(\ell)}_{-s,m}\left(\mathrm{e}^{\varphi\,\hat{\mathbf{z}}/2}\, \mathrm{e}^{\vartheta\,\hat{\mathbf{y}}/2}\right). \tag{A18}$$

These functions are implemented in the ancillary files as `GWFrames.WignerDMatrix` and `GWFrames.SWSH`.

3. Integrating the angular velocity

In many contexts, quaternion-valued functions of time turn up. These may be differentiated or integrated with respect to time, much as vector-valued functions may be. However, noncommutativity leads to certain problems. In the next section, we will see how interpolation can be handled sensibly. Here, we prove a vital formula used in the main text of this paper to integrate the angular velocity vector to find the rotor describing the frame with that angular velocity.

conventions. However, care must be taken to ensure that the resulting $\mathfrak{D}$ matrices form a representation of the rotation group rather than an anti-representation, and to ensure that the handedness of space is preserved. Our purpose in choosing these particular signs is to reproduce the standard SWSHs as special cases. In particular, note that the presence of q_3 in the definition of Q_a is what picks out the z axis as the point of reference on the sphere, so that the polar angle is measured with respect to it, rather than the x or y axes.

we can express quaternion multiplication as

$$(PQ)_a = P_a\, Q_a - \bar{P}_b\, Q_b\,, \tag{A15a}$$
$$(PQ)_b = P_b\, Q_a + \bar{P}_a\, Q_b\,. \tag{A15b}$$

Quaternions are isomorphic to spinors, and the two parts of the quaternion defined here are essentially the two components of the spinor. Then, following the standard derivation [90], we obtain

First, we need formula for the derivative of the inverse, which can be obtained by differentiating $\mathbf{Q}\,\mathbf{Q}^{-1} = 1$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{Q}^{-1} = -\mathbf{Q}^{-1}\, \frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t}\, \mathbf{Q}^{-1}\,. \tag{A19}$$

This is the crucial relation that allows us to calculate the angular velocity $\boldsymbol{\varpi}$ of a frame described by the rotor $\mathbf{R}(t)$ [32]. Suppose that a vector $\boldsymbol{v}_0$ is stationary in the rotating frame. Then, that vector is given in the inertial frame as $\boldsymbol{v}(t) = \mathbf{R}(t)\, \boldsymbol{v}_0\, \bar{\mathbf{R}}(t)$. We also know that $\mathrm{d}\boldsymbol{v}/\mathrm{d}t = \boldsymbol{\varpi}\times\boldsymbol{v}$. Using the definition of quaternion multiplication and the usual commutator (Lie product), we can calculate $\boldsymbol{\varpi}\times\boldsymbol{v} = \frac{1}{2}[\boldsymbol{\varpi},\boldsymbol{v}]$. Another way of writing this is

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{R}\,\boldsymbol{v}_0\,\bar{\mathbf{R}}) = \frac{1}{2}\,[\boldsymbol{\varpi}, \mathbf{R}\,\boldsymbol{v}_0\,\bar{\mathbf{R}}] \tag{A20a}$$
$$= [\dot{\mathbf{R}}\,\bar{\mathbf{R}}, \mathbf{R}\,\boldsymbol{v}_0\,\bar{\mathbf{R}}]\,, \tag{A20b}$$

where the second line comes from simply evaluating the left-hand side and using Eq. (A19). Now, if this is to be true for all vectors $\boldsymbol{v}_0$, then we must have

$$\boldsymbol{\varpi} = 2\,\dot{\mathbf{R}}\,\bar{\mathbf{R}}\,. \tag{A21}$$

The factor of 2 appears here because we are using quaternions; this factor does not appear in the equivalent result for rotation operators.

As explained in Sec. IV, we need an expression for the right-hand side in terms of logarithms. To borrow notation from the theory of Lie groups, we define the adjoint operator using the familiar commutator:

$$\mathrm{ad}_{\mathbf{P}}\,\mathbf{Q} \coloneqq [\mathbf{P},\mathbf{Q}] = \mathbf{P}\,\mathbf{Q} - \mathbf{Q}\,\mathbf{P}\,. \tag{A22}$$

This notation is convenient because we will need repeated applications of the commutators. For example, $\mathrm{ad}^2_{\mathbf{P}}\mathbf{Q} = [\mathbf{P},[\mathbf{P},\mathbf{Q}]]$. Now, if $\mathbf{P}$ and $\mathbf{Q}$ are unit quaternions, their logarithms will be pure vectors: $\log\mathbf{P} = \mathfrak{p}$ and $\log\mathbf{Q} = \mathfrak{q}$. We will also use the notation $\mathfrak{p} \coloneqq |\mathfrak{p}|$, etc. Again, we can use the definition of quaternion multiplication in Eq. (A2) to see

that $[\mathfrak{p}, \mathfrak{q}] = 2\,\mathfrak{p} \times \mathfrak{q}$, which allows us to use familiar properties of the cross product to calculate

$$\mathrm{ad}^n_{\mathfrak{p}}\,\mathfrak{q} = \begin{cases} \mathfrak{q} & n = 0; \\ (-1)^{(n-1)/2}\,[\mathfrak{p}, \mathfrak{q}]\,(2\mathfrak{p})^{n-1} & n \text{ odd}; \\ (-1)^{(n-2)/2}\,[\mathfrak{p}, [\mathfrak{p}, \mathfrak{q}]]\,(2\mathfrak{p})^{n-2} & n > 0 \text{ even}. \end{cases} \tag{A23}$$

The proof is a simple induction. A standard formula [92] says

$$\mathrm{e}^{\mathfrak{p}}\,\mathfrak{q}\,\mathrm{e}^{-\mathfrak{p}} = \sum_{n=0}^{\infty} \frac{1}{n!}\,\mathrm{ad}^n_{\mathfrak{p}}\mathfrak{q}\,, \tag{A24}$$

while a somewhat less-standard formula [93] gives us

$$\dot{\mathbf{P}} = \frac{\mathrm{d}\mathrm{e}^{\mathfrak{p}}}{\mathrm{d}t} = \int_0^1 \mathrm{e}^{s\,\mathfrak{p}}\,\frac{\mathrm{d}\mathfrak{p}}{\mathrm{d}t}\,\mathrm{e}^{(1-s)\,\mathfrak{p}}\,\mathrm{d}s \tag{A25}$$

for $\mathfrak{p} \le \pi$. We can multiply this formula on the right by $\mathrm{e}^{-\mathfrak{p}}$, and substitute using Eq. (A24). We then separate the resulting sum into three parts, corresponding to the three cases in Eq. (A23). These can be readily evaluated, yielding simple trigonometric functions, which can then be integrated:[20]

$$\dot{\mathbf{R}}\,\bar{\mathbf{R}} = \int_0^1 \mathrm{e}^{s\,\mathfrak{r}}\,\frac{\mathrm{d}\mathfrak{r}}{\mathrm{d}t}\,\mathrm{e}^{-s\,\mathfrak{r}}\,\mathrm{d}s \tag{A26a}$$

$$= \int_0^1 \left(\sum_{n=0}^{\infty} \frac{1}{n!}\,\mathrm{ad}^n_{s\mathfrak{r}}\dot{\mathfrak{r}}\right)\mathrm{d}s \tag{A26b}$$

$$= \int_0^1 \left(\dot{\mathfrak{r}} + \frac{\sin(2\,s\,\mathfrak{r})}{2\,\mathfrak{r}}\,[\mathfrak{r}, \dot{\mathfrak{r}}] + \frac{\sin^2(s\,\mathfrak{r})}{2\,\mathfrak{r}^2}\,[\mathfrak{r}, [\mathfrak{r}, \dot{\mathfrak{r}}]]\right)\mathrm{d}s \tag{A26c}$$

$$= \dot{\mathfrak{r}} + \frac{\sin^2\mathfrak{r}}{2\,\mathfrak{r}^2}\,[\mathfrak{r}, \dot{\mathfrak{r}}] + \frac{\mathfrak{r} - \sin\mathfrak{r}\,\cos\mathfrak{r}}{4\,\mathfrak{r}^3}\,[\mathfrak{r}, [\mathfrak{r}, \dot{\mathfrak{r}}]]\,. \tag{A26d}$$

As discussed in Sec. A 1, we evaluate the derivative $\dot{\mathfrak{r}}$ by treating $\mathfrak{r}$ as a continuous function, removing any branch cuts. On the other hand, when used without differentiating, we have assumed that $\mathfrak{r} \le \pi$.

We can re-express this relation as a matrix equation by defining

$$A := \left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \frac{\sin^2\mathfrak{r}}{\mathfrak{r}^2} \begin{pmatrix} 0 & -\mathfrak{r}_3 & \mathfrak{r}_2 \\ \mathfrak{r}_3 & 0 & -\mathfrak{r}_1 \\ -\mathfrak{r}_2 & \mathfrak{r}_1 & 0 \end{pmatrix} - \frac{\mathfrak{r} - \sin\mathfrak{r}\,\cos\mathfrak{r}}{\mathfrak{r}^3} \begin{pmatrix} \mathfrak{r}_2^2 + \mathfrak{r}_3^2 & -\mathfrak{r}_1\,\mathfrak{r}_2 & -\mathfrak{r}_1\,\mathfrak{r}_3 \\ -\mathfrak{r}_1\,\mathfrak{r}_2 & \mathfrak{r}_1^2 + \mathfrak{r}_3^2 & -\mathfrak{r}_2\,\mathfrak{r}_3 \\ -\mathfrak{r}_1\,\mathfrak{r}_3 & -\mathfrak{r}_2\,\mathfrak{r}_3 & \mathfrak{r}_1^2 + \mathfrak{r}_2^2 \end{pmatrix} \right\}. \tag{A27}$$

in which case we have the much more compact formula

$$\dot{\mathbf{R}}\,\bar{\mathbf{R}} = \frac{\boldsymbol{\varpi}}{2} = A\,\dot{\mathfrak{r}}\,. \tag{A28}$$

[20] Again, note the assumption that $\mathfrak{r} \le \pi$, which is essential to the correctness of Eq. (A26), where the actual magnitude $\mathfrak{r}$ is used. Nonetheless we also assume that the derivative $\dot{\mathfrak{r}}$ exists and is continuous everywhere, which must be enforced by removing branch-cut discontinuities before differentiating.

The determinant of the matrix simplifies to $\sin^2\mathfrak{r}/\mathfrak{r}^2$, and is thus invertible for $\mathfrak{r} < \pi$. For the rare edge case with exactly $\mathfrak{r} = \pi$, the rotation $\exp\mathfrak{r} = -1$, which corresponds to the identity rotation, so we should have $\dot{\mathfrak{r}} = \boldsymbol{\varpi}/2$. For all other cases, we can invert the matrix explicitly to find[21]

$$\dot{\mathfrak{r}} = \left(\boldsymbol{\varpi} - \frac{\mathfrak{r}\,(\mathfrak{r}\cdot\boldsymbol{\varpi})}{\mathfrak{r}^2}\right)\frac{\mathfrak{r}\,\cot\mathfrak{r}}{2} + \frac{\mathfrak{r}\,(\mathfrak{r}\cdot\boldsymbol{\varpi})}{2\,\mathfrak{r}^2} + \frac{1}{2}\boldsymbol{\varpi}\times\mathfrak{r}\,. \tag{A29}$$

Thus, we are left with an ordinary differential equation to solve for $\mathfrak{r}$, as discussed in Sec. IV. Finally, we obtain (up to the constant of integration) $\mathbf{R} = \exp\mathfrak{r}$. This is implemented in the ancillary files as `GWFrames.FrameFromAngularVelocity`.

4. Interpolation

When comparing waveforms, one of the most basic requirements is the ability to interpolate. The description of a gravitational waveform has now expanded to include both the SWSH modes of the waveform and the rotor describing the frame of that decomposition. So we need a way to interpolate rotors. But interpolation of rotors is complicated by the fact that the interpolant needs to remain normalized to unity at all times. While it is possible to simply interpolate the quaternions in $\mathbb{R}^4$ and normalize the result, the interpolant will generally exhibit unnatural accelerations between the interpolated points, even in the simplest case of uniform rotation. Interpolation of rotation matrices is just as bad. It goes without saying, of course, that interpolation of Euler angles leads to complete nonsense—the result is highly sensitive to the orientation of the coordinate basis, and depends very strongly on the conventions for which directions the successive Euler rotations take. A reasonable suggestion might be to interpolate the logarithms of the rotors and exponentiate the interpolant. However, this also leads to unnatural behaviors in fairly simple cases, whenever the logarithms of the rotors are not parallel. Fortunately, there are well-motivated solutions to the problem of quaternion interpolation that can give reasonable results in very general cases.

Recognizing that the unit quaternions can also be regarded as points on the unit sphere S^3, we might further expect an interpolant to follow the geodesic between two points on the sphere. In fact, achieving this property is actually quite simple, using the fact that the quaternions operate as a (Lie) group. A simple interpolation between unit quaternions $\mathbf{R}_0$ and $\mathbf{R}_1$ that preserves the normalization is given by [95]

$$L(\tau; \mathbf{R}_0, \mathbf{R}_1) = \left(\mathbf{R}_1\,\bar{\mathbf{R}}_0\right)^{\tau}\,\mathbf{R}_0 = \mathbf{R}_0\,\left(\bar{\mathbf{R}}_0\,\mathbf{R}_1\right)^{\tau}\,. \tag{A30}$$

Obviously, $L(0; \mathbf{R}_0, \mathbf{R}_1) = \mathbf{R}_0$ and $L(1; \mathbf{R}_0, \mathbf{R}_1) = \mathbf{R}_1$, and the norm of $L(\tau; \mathbf{R}_0, \mathbf{R}_1)$ is always 1. This formula is strongly analogous to the formula for standard linear interpolation, except that multiplication by τ becomes exponentiation and addition

[21] An equivalent formula was found using a very different derivation by Grassia [94].

becomes multiplication.[22] This interpolation is referred to as "slerp" for *s*pherical *l*inear int*erp*olation. It will be useful to note that

$$\frac{\mathrm{d}}{\mathrm{d}\tau} L(\tau; \mathbf{R}_0, \mathbf{R}_1) = \log\left(\mathbf{R}_1 \bar{\mathbf{R}}_0\right) L(\tau; \mathbf{R}_0, \mathbf{R}_1) \,, \tag{A31a}$$

$$= L(\tau; \mathbf{R}_0, \mathbf{R}_1) \log\left(\bar{\mathbf{R}}_0 \mathbf{R}_1\right) \,. \tag{A31b}$$

This formula shows us that the speed along the slerp path is constant, as it must be for a geodesic.

Many problems only call for a linear interpolation like slerp. In particular, when blending PN and NR waveforms, each waveform possesses its own frame. To transition between the two waveforms, we must transition between the frames, which is just a simple linear interpolation at each instant of time where the extent of the interpolation (the τ argument to the function above) depends on the time. However, we also need to be able to interpolate each individual waveform as a function of time.[23] For this, we cannot use linear interpolation for the motion of the frame of either waveform. If we did, we would see the frame abruptly change rotation speed as it goes through each original data point, just as a linearly interpolated graph changes slope abruptly as it passes through each original data point. Instead, we would prefer some higher-order technique.

We can approach this problem in analogy with the construction of curves in space, which suggests various approaches such as the de Casteljau algorithm for constructing Bézier curves. Unfortunately, the various methods—while being equivalent for real numbers—are not equivalent when using quaternions because of noncommutativity [96]. It is not clear that any particular formulation will give better results than any other, so we may take the pragmatic approach of simply choosing one which is easily implemented. The result will be a *s*pherical interpolation based on the *quad*rilateral of a standard spline, referred to as *squad*.

In that spirit, we will define the cubic-spline interpolant in terms of the linear interpolant:

$$C(t; \mathbf{R}_i, \mathbf{A}_i, \mathbf{B}_{i+1}, \mathbf{R}_{i+1}) = L\Big(2\tau_i(1-\tau_i); L(\tau_i; \mathbf{R}_i, \mathbf{R}_{i+1}), L(\tau_i; \mathbf{A}_i, \mathbf{B}_{i+1})\Big) \,, \tag{A32}$$

where $\mathbf{A}_i$ and $\mathbf{B}_{i+1}$ are "control points" to be solved for. We have also defined $\tau_i(t) = (t - t_i)/(t_{i+1} - t_i)$, where i is assumed to be the index of the nearest time sample such that $t_i \leq t$. We can evaluate the derivative

$$\frac{\mathrm{d}}{\mathrm{d}\tau_i} C = \frac{\mathrm{d}}{\mathrm{d}\tau_i}\Big\{ \exp\Big[2\tau_i(1-\tau_i) \log\Big(L(\tau_i; \mathbf{A}_i, \mathbf{B}_{i+1})\, L(\tau_i; \mathbf{R}_i, \mathbf{R}_{i+1})^{-1}\Big)\Big] L(\tau_i; \mathbf{R}_i, \mathbf{R}_{i+1})\Big\} \tag{A33a}$$

$$= (2 - 4\tau_i) \log\Big(L(\tau_i; \mathbf{A}_i, \mathbf{B}_{i+1})\, L(\tau_i; \mathbf{R}_i, \mathbf{R}_{i+1})^{-1}\Big) C + 2\tau_i(1-\tau_i)\, G + C \log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1}) \,, \tag{A33b}$$

where G is a complicated expression, which is easy to compute, but messy to write; fortunately do not need to evaluate it because that term drops out when we evaluate at $\tau_i = 0$ or $\tau_i = 1$, as the factor in front of G goes to zero. We wish to ensure that the time derivatives are equal at the end of one segment and the beginning of the next:

$$\frac{\mathrm{d}}{\mathrm{d}t} C(\tau_{i-1}, \mathbf{R}_{i-1}, \mathbf{A}_{i-1}, \mathbf{B}_i, \mathbf{R}_i)\Big|_{\tau_{i-1}=1} = \frac{\mathrm{d}}{\mathrm{d}t} C(\tau_i, \mathbf{R}_i, \mathbf{A}_i, \mathbf{B}_{i+1}, \mathbf{R}_{i+1})\Big|_{\tau_i=0} \,. \tag{A34}$$

Note that we differentiate with respect to t, rather than τ_i, to account for differences in the time steps of the given data.Plugging in the result of Eq. (A33) and simplifying, we get

$$\frac{1}{\Delta t_{i-1}} \left\{-2 \log\left(\mathbf{B}_i \bar{\mathbf{R}}_i\right) \mathbf{R}_i + \mathbf{R}_i \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)\right\} = \frac{1}{\Delta t_i} \left\{2 \log\left(\mathbf{A}_i \bar{\mathbf{R}}_i\right) \mathbf{R}_i + \mathbf{R}_i \log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})\right\} \,, \tag{A35}$$

or equivalently

$$\frac{1}{\Delta t_{i-1}} \mathbf{R}_i \left\{-2 \log\left(\bar{\mathbf{R}}_i \mathbf{B}_i\right) + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)\right\} = \frac{1}{\Delta t_i} \mathbf{R}_i \left\{2 \log\left(\bar{\mathbf{R}}_i \mathbf{A}_i\right) + \log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})\right\} \,. \tag{A36}$$

We need one more condition to solve for both variables $\mathbf{A}_i$ and $\mathbf{B}_i$. We may choose[24] to set the velocity at either side equal to the

[22] This analogy should not be carried too far because quaternion multiplication is noncommutative. In particular, it is crucial to note that the right-hand side of Eq. (A30) is not equal to $\mathbf{R}_1 \mathbf{R}_0^{1-\tau}$, for example, whenever $\mathbf{R}_0$ and $\mathbf{R}_1$ do not commute; such a formula actually gives very poor interpolation in many cases. Equation (A30) is preferable because the path it describes is a geodesic in the space of unit quaternions.

[23] For example, the PN waveform and the NR waveform will generally be calculated at different instants of time. To compare them, we need to be able to interpolate the values of one waveform onto the time steps of the other.

[24] This choice has the nice property of agreeing with our intuition in the case of "straight-line" motion. To be precise: if the transformation from $\mathbf{R}_{i-1}$ to $\mathbf{R}_i$ is written as multiplication by $\mathbf{R}_i \bar{\mathbf{R}}_{i-1}$, then "straight-line" motion occurs when $\mathbf{R}_{i+1} = \mathbf{R}_i \bar{\mathbf{R}}_{i-1} \mathbf{R}_i$ (and for simplicity, we assume $\Delta t_{i-1} = \Delta t_i$). Then this average velocity is $\mathbf{R}_i \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)/\Delta t$, which is precisely the velocity of a linear interpolation at that point.

average velocity of linear interpolations on those two sides, giving us the two equations:

$$\mathbf{R}_i \frac{\log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})/\Delta t_i + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)/\Delta t_{i-1}}{2} = \frac{1}{\Delta t_{i-1}} \mathbf{R}_i \left\{-2 \log\left(\bar{\mathbf{R}}_i \mathbf{B}_i\right) + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)\right\} \tag{A37a}$$

$$\mathbf{R}_i \frac{\log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})/\Delta t_i + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)/\Delta t_{i-1}}{2} = \frac{1}{\Delta t_i} \mathbf{R}_i \left\{2 \log\left(\bar{\mathbf{R}}_i \mathbf{A}_i\right) + \log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})\right\} . \tag{A37b}$$

We can now solve for the control points:

$$\mathbf{A}_i = \mathbf{R}_i \exp\left[\frac{\log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1}) + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)\, \Delta t_i/\Delta t_{i-1} - 2 \log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})}{4}\right] \tag{A38a}$$

$$\mathbf{B}_i = \mathbf{R}_i \exp\left[-\frac{\log(\bar{\mathbf{R}}_i \mathbf{R}_{i+1})\, \Delta t_{i-1}/\Delta t_i + \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i) - 2 \log(\bar{\mathbf{R}}_{i-1} \mathbf{R}_i)}{4}\right] . \tag{A38b}$$

To apply these formulas to the edge cases of $i = 0$ and $i = N - 1$, we also define the quantities $\mathbf{R}_{-1} = \mathbf{R}_0 \bar{\mathbf{R}}_1 \mathbf{R}_0$ and $\mathbf{R}_N = \mathbf{R}_{N-1} \bar{\mathbf{R}}_{N-2} \mathbf{R}_{N-1}$, which roughly represent straight-line motion.

In the computer code included among this paper's ancillary files, the functions `GWFrames.Slerp` and `GWFrames.Squad` implement linear and cubic interpolations of rotors.

Appendix B: Rotating spin-weighted functions

Gravitational radiation is a complex field of nonzero spin weight, meaning that it picks up a position-dependent phase under rotation [89]. The reason for this is its definition with respect to a dyadic which is itself defined in terms of a *coordinate* basis; when the coordinates rotate, the dyadic rotates. Depending on details of the definition of the gravitational-wave field, the spin weight may be $s = 2$ or $s = -2$—the most common choice being the latter. Throughout the rest of this paper, we have assumed $s = -2$; in order to discuss the properties of general spin-weighted fields, this appendix will apply to general values of s.

Suppose we have a field f of spin weight s on the sphere. To measure this field, we first need some standard basis for our space, $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}})$. We can define the usual spherical coordinates relative to this basis, and write the field as a function of the coordinates, so that in some particular direction $\hat{\boldsymbol{n}}$, we have $f(\hat{\boldsymbol{n}}) = f(\vartheta, \varphi)$. We define the rotor

$$\mathbf{R}_{(\vartheta,\varphi)} \coloneqq \mathrm{e}^{\varphi \hat{\mathbf{z}}/2}\, \mathrm{e}^{\vartheta \hat{\boldsymbol{y}}/2} \tag{B1}$$

and note that

$$\hat{\boldsymbol{n}} = \mathbf{R}_{(\vartheta,\varphi)}\, \hat{\mathbf{z}}\, \bar{\mathbf{R}}_{(\vartheta,\varphi)} . \tag{B2}$$

Now, given a second basis $(\hat{\boldsymbol{x}}', \hat{\boldsymbol{y}}', \hat{\mathbf{z}}')$ related to the first by a rotor $\mathbf{R}$, we can define another set of spherical coordinates, and another rotor $\mathbf{R}_{(\vartheta',\varphi')}$, for which

$$\hat{\boldsymbol{n}} = \mathbf{R}_{(\vartheta',\varphi')}\, \hat{\mathbf{z}}'\, \bar{\mathbf{R}}_{(\vartheta',\varphi')} = \mathbf{R}_{(\vartheta',\varphi')}\, \mathbf{R}\, \hat{\mathbf{z}}\, \bar{\mathbf{R}}\, \bar{\mathbf{R}}_{(\vartheta',\varphi')} . \tag{B3}$$

Because $\hat{\boldsymbol{n}}$ has a geometric meaning independent of any basis, we can equate these two expressions. In general, this implies the relationship

$$\mathbf{R}_{(\vartheta,\varphi)}\, \mathrm{e}^{\gamma \hat{\mathbf{z}}/2} = \mathbf{R}_{(\vartheta',\varphi')}\, \mathbf{R} , \tag{B4}$$

for some angle γ which depends on (ϑ, φ).[25] The term involving γ represents an *initial* rotation through that angle in the positive sense about the direction $\hat{\mathbf{z}}$, which is equivalent to a *final* rotation about the direction $\hat{\boldsymbol{n}}$. For spin-weighted functions, this corresponds [89] to multiplication of the function value by $\mathrm{e}^{-\mathrm{i}\, s\, \gamma}$. Thus, in this basis, we measure a different field $\hat{f}$, related to the field f measured in the first basis by

$$\hat{f}(\vartheta', \varphi') = f(\vartheta, \varphi)\, \mathrm{e}^{-\mathrm{i}\, s\, \gamma} . \tag{B5}$$

For $s = 0$, we recover the familiar result that a scalar field does not depend on the frame in which it is measured.

The spin-weighted spherical harmonics (SWSHs) form a basis for spin-weighted functions on the sphere [87–89], just as standard spherical harmonics form a basis for spin-zero functions. We can write

$$f(\vartheta, \varphi) = \sum_{\ell,m} f^{\ell,m}\, {}_sY_{\ell,m}(\vartheta, \varphi) , \tag{B6a}$$

$$\hat{f}(\vartheta', \varphi') = \sum_{\ell,m} \hat{f}^{\ell,m}\, {}_sY_{\ell,m}(\vartheta', \varphi') . \tag{B6b}$$

The SWSHs themselves are just special cases of the Wigner $\mathfrak{D}$ matrices (see Eq. (A18) and Ref. [47]). We can then use the fact that the $\mathfrak{D}$ matrices form a representation of the rotation group to find the transformation law for SWSHs:

$$\mathfrak{D}^{(\ell)}_{m',m}(\mathbf{R}_1\, \mathbf{R}_2) = \sum_{m''} \mathfrak{D}^{(\ell)}_{m',m''}(\mathbf{R}_1)\, \mathfrak{D}^{(\ell)}_{m'',m}(\mathbf{R}_2) \tag{B7}$$

implies, using Eq. (B4), that

$${}_sY_{\ell,m}(\vartheta, \varphi)\, \mathrm{e}^{-\mathrm{i}\, s\, \gamma} = \sum_{m'} {}_sY_{\ell,m'}(\vartheta', \varphi')\, \mathfrak{D}^{(\ell)}_{m',m}(\mathbf{R}) . \tag{B8}$$

[25] This angle is required to account for the full three-dimensional freedom in choosing $\mathbf{R}$. It is always possible to find such an angle. However, this angle need not be unique for certain orientations; γ may be degenerate with φ. Similarly, because of the familiar singularities of the spherical coordinates, there may not be a unique choice of (ϑ, φ) or (ϑ', φ') for certain positions. Nonetheless, the rotations $\mathbf{R}_{(\vartheta,\varphi)}\, \mathrm{e}^{\gamma \hat{\mathbf{z}}/2}$ and $\mathbf{R}_{(\vartheta',\varphi')}$ generated by these angles *will* be uniquely determined, much as the North and South Poles are uniquely determined despite the ill-defined longitude at those points.

The dependence of γ on (ϑ, φ) means that, strictly speaking, the SWSHs with $s \neq 0$ do *not* transform among themselves under rotations. Naturally, when coupled to the appropriate spin-weighted tensors, the complete object transforms as expected [16]. Similarly, the modes $f^{\ell,m}$ transform nicely thanks to a convenient cancellation. Inserting Eqs. (B6) and (B8) into Eq. (B5), we can show that

$$\hat{f}^{\ell,m'} = \sum_m f^{\ell,m}\, \mathfrak{D}^{(\ell)}_{m',m}(\mathbf{R})\,, \tag{B9a}$$

or equivalently

$$f^{\ell,m} = \sum_{m'} \hat{f}^{\ell,m'}\, \mathfrak{D}^{(\ell)}_{m,m'}(\bar{\mathbf{R}})\,. \tag{B9b}$$

These are precisely the same as the transformation laws for modes of standard ($s = 0$) spherical harmonics, and do not depend on γ.

Expressions for the Wigner $\mathfrak{D}$ matrices are given directly in terms of the rotor in Eq. (A16), which avoids the need for conversion to Euler angles. The SWSHs are expressed as particular components of these matrices in Eq. (A18). In the computer code included among this paper's ancillary files, the Wigner $\mathfrak{D}$ matrices and SWSHs are implemented as `GWFrames.WignerDMatrix` and `GWFrames.SWSH`. Waveform objects may be constructed with `GWFrames.Waveform`, and transformed to different frames using the method `RotateDecompositionBasis`.

Appendix C: Other methods of choosing a frame

In the interests of completeness, and to facilitate direct comparisons using common language, we now review three other methods of choosing a frame to eliminate mode-mixing in waveforms from precessing systems. Each of these methods constructs a new frame by ensuring that the $\hat{\mathbf{z}}'$ direction lies along some chosen axis which is roughly the axis of rotation of the waveform. These differ from the corotating frame introduced in the main text of this paper, in that the waveform is still rotating in these new frames. The considerations of Sec. III (and in particular the left panel of Fig. 1) suggest that among these three, the preferred method is that of O'Shaughnessy *et al*. supplemented with the minimal-rotation condition [32]. In particular, when setting the integration constant discussed near the end of Sec. IV A, that is the method of choice. However, over all, the corotating frame is generally still a preferable choice.

We first describe two methods suggested by Schmidt *et al*. [30] and O'Shaughnessy *et al*. [31], using a common notation which allows a common implementation by means of explicit maximization of a quality function. We then describe the method of O'Shaughnessy *et al*. in the way in which it was introduced, which allows a second implementation by solution of an eigensystem. A third possible axis suggests itself given the results of this paper: the angular-velocity vector $\boldsymbol{\omega}$ given by Eq. (7). All three need an additional step to remove sharp features in the waveforms, given by the minimal-rotation condition. In this section, we review each alternative in the language of quaternions, suggesting improvements for numerical accuracy and robustness.

1. Maximization

In general, we can describe the process of finding the radiation axis as a maximization over $\mathbf{R}$ of the quantity

$$Q(\mathbf{R}) = \sum_{\ell,m} w_{\ell,m} \left|\hat{f}^{\ell,m}\right|^2 \tag{C1a}$$

$$= \sum_{\ell,m} w_{\ell,m} \left|\sum_{m'} f^{\ell,m'}\, \mathfrak{D}^{(\ell)}_{m,m'}(\mathbf{R})\right|^2 . \tag{C1b}$$

Here, the $w_{\ell,m}$ are simply weighting factors. Schmidt *et al*. took these factors to be $w_{2,\pm2} = 1$, and zero otherwise; O'Shaughnessy *et al*. effectively chose $w_{\ell,m} = m^2$, with some cutoff ℓ above which $w_{\ell,m} = 0$.

This function is actually degenerate with respect to initial rotations about the $\hat{\mathbf{z}}$ axis, because such rotations simply affect the overall phase of the term inside the absolute value. For numerical efficiency, we need to restrict Q to some nondegenerate domain for efficient numerical maximization. Whereas previous references [30–32] used rotations of the form $\mathbf{R}_{(\vartheta,\varphi)}$, we choose instead to use rotations of the form $\mathbf{R}_{(\vartheta,\varphi)}\, \mathrm{e}^{-\varphi\,\hat{\mathbf{z}}/2}$. All such rotations can be written as $\mathbf{R}_{\boldsymbol{v}} = \mathrm{e}^{\boldsymbol{v}}$ for some vector $\boldsymbol{v}$ in the x–y plane, of magnitude less than or equal to $\pi/2$. Using rotations of this form significantly simplifies calculation of the Wigner $\mathfrak{D}$ matrices and eliminates the degeneracy near the identity, which substantially improves numerical accuracy and stability for mildly precessing systems.

Of course, the sphere cannot be covered homeomorphically by a single coordinate chart, so an additional degeneracy remains: all vectors on the boundary of our domain result in rotations with equal values of Q. However, this set has measure zero in the domain itself, meaning that it is almost never encountered. Moreover, the effect of this degeneracy will be completely eliminated in Sec. C 4. In fact, we find it convenient to extend the domain further. We maximize $Q(\mathrm{e}^{\boldsymbol{v}})$ for all vectors $\boldsymbol{v}$ in the entire x–y plane, parameterizing the function arguments by the usual coordinates $(x, y) \in \mathbb{R}^2$. There are now degeneracies on circles of radius $n\,\pi/2$ centered on the origin for all integers $n > 0$. Again, however, these degeneracies cause no practical difficulties.

Given values for the modes $f^{\ell,m}$ and the weights $w_{\ell,m}$, the right-hand side of Eq. (C1b) is known analytically, using Eq. (A16), as are its derivatives with respect to x and y. These functions are ungainly, but can be written down explicitly, plugged into a computer, and used in efficient numerical optimization routines. Direct maximization of Eq. (C1) is simple to implement, and can be made reasonably efficient and robust. It does have disadvantages, however. At each step in the minimization routine, all relevant $\mathfrak{D}$ matrices need to be recomputed. When the $w_{\ell,m}$ are nonzero for many values, this can become very expensive. In such cases, it can be significantly more efficient to find a radiation axis using the following method.

In the computer code included among this paper's ancillary files, a waveform object may be constructed with `GWFrames.Waveform`. The axis suggested by Schmidt *et al*. may then be found by applying the `SchmidtEtAlVector` method.

2. Dominant principal axis

In general, if $w_{\ell,m} = w_\ell\, m^2$ where w_ℓ only depends on ℓ, then this can be presented in a different form and solved as an eigenvector problem—which is the approach O'Shaughnessy *et al*. actually used when introducing their method. Define[26]

$$\langle L_{(a}\, L_{b)}\rangle \coloneqq \sum_{\ell,m,m'} w_\ell\, \bar{f}^{\ell,m'}\, \langle \ell, m'|L_{(a}\, L_{b)}|\ell, m\rangle\; f^{\ell,m}\,, \qquad \text{(C2)}$$

where L_a is the usual angular-momentum operator. The radiation axis is chosen to be the dominant principal axis $\hat{\boldsymbol{V}}_f$ of this tensor—the eigenvector with the eigenvalue of largest magnitude, which can be found with standard algebraic techniques. We can find some rotation $\mathbf{R}_{\rm ax}$ taking the z axis into the dominant principal axis. Reference [32] showed that such a rotation maximizes the function Q of Eq. (C1).

Again, however, this rotation is not unique. Moreover, the dominant principal axis is only defined up to a sign, and numerical implementations may choose between the two options effectively randomly. A naive choice of $\mathbf{R}_{\rm ax}(t)$, then, may flip back and forth discontinuously. Fortunately, we can overcome this problem easily by taking $\hat{\boldsymbol{a}}_i \mapsto -\hat{\boldsymbol{a}}_i$ whenever $\hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_{i-1} < 0$. Then, we can ensure that the appropriate $\mathbf{R}_{\rm ax}(t_i)$ is as close[27] as possible to $\mathbf{R}_{\rm ax}(t_{i-1})$ by choosing

$$\mathbf{R}_\Delta \coloneqq \sqrt{-\hat{\boldsymbol{a}}_i\, \hat{\boldsymbol{a}}_{i-1}}\,, \qquad \text{(C3a)}$$

$$\mathbf{R}_{\rm ax}(t_i) = \mathbf{R}_\Delta\, \mathbf{R}_{\rm ax}(t_{i-1})\,. \qquad \text{(C3b)}$$

In Eq. (C3a), the vectors are multiplied as quaternions, and the square root may be found with the help of Eq. (A13). We start out with $\hat{\boldsymbol{a}}_{-1} = \hat{\boldsymbol{z}}$, and build up the frame by stepping forward in time according to Eq. (C3), where each $\hat{\boldsymbol{a}}_i$ is the dominant principal axis at that instant of time. However, for reasons of numerical stability, $\hat{\boldsymbol{a}}_{i-1}$ in Eq. (C3a) should be expressed as $\mathbf{R}_{\rm ax}(t_{i-1})\, \hat{\boldsymbol{z}}\, \bar{\mathbf{R}}_{\rm ax}(t_{i-1})$, rather than as the principal axis at the previous instant.

The frame found by this method has certain advantages over maximization of Eq. (C1). In particular, the matrix in Eq. (C2) need only be computed once for each time step. The dominant principal axis is then obtained from this. No calculations of Wigner's $\mathfrak{D}$ matrices are necessary, which tends to make the computation fast. Also, some minor care is needed to make the present method robust, mostly involving choosing the direction of the axis to be consistent from moment to moment.

In the computer code included among this paper's ancillary files, a waveform object may be constructed with `GWFrames.Waveform`. The dominant principal axis of $\langle L_{(a}\, L_{b)}\rangle$ may then be found by applying the `OShaughnessyEtAlVector` method.

[26] O'Shaughnessy *et al*. used $w_2 = 1$, and 0 for all other weights, as well as an overall normalization which is ignored here for simplicity.

[27] The distance between two rotations $\mathbf{R}_1$ and $\mathbf{R}_2$ can be defined as $2\,\left|\log(\bar{\mathbf{R}}_1\, \mathbf{R}_2)\right|$, which is the minimum angle needed to rotate one into the other. See Appendix A 1 for more details.

3. Aligned with the angular velocity

A very similar frame can be defined, using the angular-velocity vector $\boldsymbol{\omega}$ in place of the dominant principal axis of $\langle L_{(a}\, L_{b)}\rangle$. The vector $\boldsymbol{\omega}$ was found in Sec. II A and is given explicitly by Eq. (7). This can be used for the $\hat{\boldsymbol{a}}_i$ in Eq. (C3). Note that using the only the direction of the vector to align the axis of the new frame throws away some information. Specifically, the magnitude of $\boldsymbol{\omega}$ is meaningful and is used in Sec. IV to derive a fully corotating frame. Nonetheless, the waveform rotation in this frame is about the z' axis at each instant, making the time dependence of the waveform in this frame quite similar to that of a nonprecessing system in a stationary frame.

In the computer code included among this paper's ancillary files, a waveform object may be constructed with `GWFrames.Waveform`. The angular velocity may then be found using the `AngularVelocityVector` method on such an object.

4. The minimal-rotation condition

Each of the three methods discussed above is critically flawed when applied to a time-series of data, unless followed by the procedure described here. The end result of any of the three previous methods is some rotation $\mathbf{R}_{\rm ax}(t)$ that takes the z axis into the chosen radiation axis: $\hat{\boldsymbol{a}}(t) = \mathbf{R}_{\rm ax}(t)\, \hat{\boldsymbol{z}}\, \bar{\mathbf{R}}_{\rm ax}(t)$. As mentioned, however, this is by no means the only such rotation. Indeed, because of the invariance of $\hat{\boldsymbol{z}}$ under rotations about the z axis, any rotation of the form

$$\mathbf{R}(t) = \mathbf{R}_{\rm ax}(t)\, \exp\left[\frac{\gamma(t)}{2}\, \hat{\boldsymbol{z}}\right] \qquad \text{(C4)}$$

will do the same. Arbitrarily setting $\gamma(t) = 0$ leaves us with large extraneous features in the phase of each mode of the waveform. Reference [32] showed that it is easy to impose a condition on $\gamma(t)$ such that the total rotation $\mathbf{R}$ satisfies a geometrically and physically meaningful criterion referred to as the *minimal-rotation condition*. This section simply reiterates the previous description in quaternion form and suggests a more accurate way of finding $\dot{\mathbf{R}}_{\rm ax}$ in some cases.

To motivate this condition, we first define the radiation frame's instantaneous angular-velocity vector $\boldsymbol{\varpi}$. Then, for any vector $\boldsymbol{v}$ that is stationary in the radiation frame, its derivative in an inertial frame is given by

$$\dot{\boldsymbol{v}} = \boldsymbol{\varpi} \times \boldsymbol{v}\,, \qquad \text{(C5)}$$

where a dot denotes differentiation with respect to time. A radiation frame is—by definition—a frame in which the radiation axis $\hat{\boldsymbol{a}}$ is stationary. So, its derivative in an inertial frame is $\dot{\hat{\boldsymbol{a}}} = \boldsymbol{\varpi} \times \hat{\boldsymbol{a}}$. Taking the cross product of both sides of this equation by $\hat{\boldsymbol{a}}$, using the standard vector triple product formula with the fact that $\hat{\boldsymbol{a}}$ has unit magnitude, then rearranging, we find

$$\boldsymbol{\varpi} = \hat{\boldsymbol{a}} \times \dot{\hat{\boldsymbol{a}}} + (\hat{\boldsymbol{a}} \cdot \boldsymbol{\varpi})\, \hat{\boldsymbol{a}}\,. \qquad \text{(C6)}$$

Now, we might hope that since $\hat{\boldsymbol{a}}(t)$ is actually measured from the waveform, this might be enough to specify the frame.

Unfortunately, Eq. (C6) defines the component of $\boldsymbol{\varpi}$ along $\hat{\boldsymbol{a}}$ circularly; it is undetermined, so we need another condition. Of course, an obvious solution presents itself. When the radiation axis is stationary, we can expect that the frame should be stationary. To achieve this, Eq. (C6) shows that we must have $\boldsymbol{\varpi} \cdot \hat{\boldsymbol{a}} = 0$. Because $\hat{\boldsymbol{a}}$ is a geometric object, independent of the frame in which it is measured, this relation is geometrically meaningful. We therefore require this condition even in the nonprecessing case. This minimizes the magnitude of $\boldsymbol{\varpi}$, so we refer to it as the *minimal-rotation condition* [18, 32]. We adopt this condition as the criterion for selecting the radiation frame.

Of course, the frame is not given by its instantaneous rotation vector, but by its orientation at each instant of time. So we need to express $\boldsymbol{\varpi}$ in terms of $\mathbf{R}(t)$ to impose our condition. This is conveniently calculated in Sec. A 3, which shows that $\boldsymbol{\varpi} = 2\,\dot{\mathbf{R}}\,\bar{\mathbf{R}}$. Because the radiation axis is given by $\hat{\boldsymbol{a}} = \mathbf{R}\,\hat{\boldsymbol{z}}\,\bar{\mathbf{R}}$, the minimal-rotation condition becomes $\boldsymbol{\varpi} \cdot \hat{\boldsymbol{a}} = 2\,\dot{\mathbf{R}}\,\bar{\mathbf{R}} \cdot \mathbf{R}\,\hat{\boldsymbol{z}}\,\bar{\mathbf{R}} = 0$. Invariance of the dot product under rotation shows that we can also write this as $\bar{\mathbf{R}}\,\dot{\mathbf{R}} \cdot \hat{\boldsymbol{z}} = 0$. Expanding $\mathbf{R}$ as given in Eq. (C4), we see that the minimal-rotation condition is satisfied if $\gamma(t)$ satisfies

$$\dot{\gamma}(t) = -2\,\bar{\mathbf{R}}_{\rm ax}(t)\,\dot{\mathbf{R}}_{\rm ax}(t) \cdot \hat{\boldsymbol{z}} = 2\left(\bar{\mathbf{R}}_{\rm ax}(t)\,\dot{\mathbf{R}}_{\rm ax}(t)\,\hat{\boldsymbol{z}}\right)_0 \,, \tag{C7}$$

where the subscript 0 here denotes the scalar part. Now, since $\mathbf{R}_{\rm ax}$ is assumed to be known—perhaps by one of the three foregoing methods—we can evaluate the right-hand side, then integrate in time, and insert the result into Eq. (C4). Note that the integration constant $\gamma(0)$ is undetermined. This corresponds to the usual freedom in choosing a phase, familiar from nonprecessing systems, and will have to be fixed in a similar way.

As a practical matter, the rotor $\mathbf{R}_{\rm ax}$ is typically computed using Eq. (A13) with $\hat{\boldsymbol{w}} = \hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{u}} = \hat{\boldsymbol{a}}$. We can easily differentiate this, assuming $\hat{\boldsymbol{w}}$ is constant, and arrive at an analytical formula for $\dot{\mathbf{R}}_{\rm ax}$ in terms of $\dot{\hat{\boldsymbol{a}}}$. In the construction of PN waveforms, the latter is known analytically, and may be inserted into this formula for higher accuracy:

$$\dot{\mathbf{R}}_{\rm ax} = \pm\partial_t \frac{1 - \hat{\boldsymbol{a}}\,\hat{\boldsymbol{z}}}{\sqrt{2[1 - (\hat{\boldsymbol{a}}\,\hat{\boldsymbol{z}})_0]}} \tag{C8a}$$

$$= \pm\left(\frac{-\partial_t \hat{\boldsymbol{a}}\,\hat{\boldsymbol{z}}}{\sqrt{2[1 + \hat{\boldsymbol{a}}_3]}} - \frac{\partial_t \hat{\boldsymbol{a}}_3}{2[1 + \hat{\boldsymbol{a}}_3]}\mathbf{R}_{\rm ax}\right) . \tag{C8b}$$

Here, we have used $-(\hat{\boldsymbol{a}}\,\hat{\boldsymbol{z}})_0 = \hat{\boldsymbol{a}}_3$ for simplicity.

In the computer code included among this paper's ancillary files, an array of quaternions can be put into minimal-rotation form using the `GWFrames.MinimalRotation` function. A waveform object constructed with `GWFrames.Waveform` can be transformed into the frames discussed in this section using methods beginning with `TransformTo`.

[1] The LIGO Scientific Collaboration, Repts. Prog. Phys. **72**, 076901 (2009).
[2] D. Shoemaker, *LIGO-T0900288-v3: Advanced LIGO anticipated sensitivity curves*, Tech. Rep. (LIGO, 2010).
[3] S. J. Waldman for the LIGO Scientific Collaboration and the Virgo Collaboration, "The advanced LIGO gravitational wave detector," (2011), arXiv:1103.2728 [gr-qc].
[4] The Virgo Collaboration, J. Instr. **7**, P03012 (2012).
[5] Kentaro Somiya for the KAGRA collaboration, Class. Quant. Grav. **29**, 124007 (2012).
[6] Alan J. Weinstein for the LIGO Scientific Collaboration and the Virgo Collaboration, Class. Quant. Grav. **29**, 124012 (2012).
[7] V. Kalogera, Astrophys. J. **541**, 319 (2000).
[8] R. O'Shaughnessy, J. Kaplan, V. Kalogera, and K. Belczynski, Astrophys. J. **632**, 1035 (2005).
[9] P. Grandclément, M. Ihm, V. Kalogera, and K. Belczynski, Phys. Rev. D **69**, 102002 (2004).
[10] LIGO Scientific Collaboration and Virgo Collaboration, Class. Quant. Grav. **27**, 173001 (2010).
[11] L. E. Kidder, Phys. Rev. D **52**, 821 (1995).
[12] T. A. Apostolatos, C. Cutler, G. J. Sussman, and K. S. Thorne, Phys. Rev. D **49**, 6274 (1994).
[13] C. M. Will and A. G. Wiseman, Phys. Rev. D **54**, 4813 (1996).
[14] T. A. Apostolatos, Phys. Rev. D **54**, 2438 (1996).
[15] A. Einstein, in *Albert Einstein: Akademie-Vorträge*, edited by D. Simon (Wiley-VCH Verlag GmbH & Co. KGaA, 2006) pp. 154–166.
[16] K. S. Thorne, Rev. Mod. Phys. **52**, 299 (1980).
[17] L. Blanchet, G. Faye, B. R. Iyer, and S. Sinha, Class. Quant. Grav. **25**, 165003 (2008), note erratum.
[18] A. Buonanno, Y. Chen, and M. Vallisneri, Phys. Rev. D **67**, 104025 (2003).
[19] L. Blanchet, A. Buonanno, and G. Faye, Phys. Rev. D **74**, 104034 (2006), note the two associated errata; the arXiv version is fully up to date, arXiv:gr-qc/0605140v4.
[20] L. Blanchet, A. Buonanno, and G. Faye, Phys. Rev. D **75**, 049903(E) (2007).
[21] K. G. Arun, A. Buonanno, G. Faye, and E. Ochsner, Phys. Rev. D **79**, 104023 (2009), note that the 1.5 and 2.5pN spin terms in the flux and energy expressions do not account for an erratum from 2010.
[22] L. Blanchet, A. Buonanno, and G. Faye, Phys. Rev. D **81**, 089901(E) (2010).
[23] S. Marsat, A. Bohé, G. Faye, and L. Blanchet, "Next-to-next-to-leading order spin-orbit effects in the equations of motion of compact binary systems," (2012), arXiv:1210.4143 [gr-qc].
[24] A. Bohé, S. Marsat, G. Faye, and L. Blanchet, "Next-to-next-to-leading order spin-orbit effects in the near-zone metric and precession equations of compact binaries," (2012), arXiv:1212.5520 [gr-qc].
[25] P. Ajith, Phys. Rev. D **84**, 084037 (2011).
[26] P. Schmidt, M. Hannam, and S. Husa, "Towards models of gravitational waveforms from generic binaries: A simple approximate mapping between precessing and non-precessing inspiral signals," (2012), arXiv:1207.3088 [gr-qc].
[27] D. A. Brown, A. Lundgren, and R. O'Shaughnessy, Phys. Rev. D **86**, 064020 (2012).
[28] D. A. Brown, I. Harry, A. Lundgren, and A. H. Nitz, Phys. Rev. D **86**, 084017 (2012).
[29] R. O'Shaughnessy, L. London, J. Healy, and D. Shoemaker,

"Precession during merger 1: Strong polarization changes are observationally accessible features of strong-field gravity during binary black hole merger," (2012), arXiv:1209.3712 [gr-qc].
[30] P. Schmidt, M. Hannam, S. Husa, and P. Ajith, Phys. Rev. D **84**, 024046 (2011).
[31] R. O'Shaughnessy, B. Vaishnav, J. Healy, Z. Meeks, and D. Shoemaker, Phys. Rev. D **84**, 124002 (2011).
[32] M. Boyle, R. Owen, and H. P. Pfeiffer, Phys. Rev. D **84**, 124011 (2011).
[33] P. Jaranowski, K. D. Kokkotas, A. Królak, and G. Tsegas, Class. Quant. Grav. **13**, 1279 (1996).
[34] M. V. van der Sluys, C. Röver, A. Stroeer, V. Raymond, I. Mandel, N. Christensen, V. Kalogera, R. Meyer, and A. Vecchio, Astrophys. J. **688**, L61 (2008).
[35] M. van der Sluys, V. Raymond, I. Mandel, C. Röver, N. Christensen, V. Kalogera, R. Meyer, and A. Vecchio, Class. Quant. Grav. **25**, 184011 (2008).
[36] V. Raymond, M. V. van der Sluys, I. Mandel, V. Kalogera, C. Röver, and N. Christensen, Class. Quant. Grav. **26**, 114007 (2009).
[37] S. Nissanke, D. E. Holz, S. A. Hughes, N. Dalal, and J. L. Sievers, Astrophys. J. **725**, 496 (2010).
[38] L. Wen and Y. Chen, Phys. Rev. D **81**, 082001 (2010).
[39] S. Fairhurst, Class. Quant. Grav. **28**, 105021 (2011).
[40] S. Klimenko, G. Vedovato, M. Drago, G. Mazzolo, G. Mitselmakher, C. Pankow, G. Prodi, V. Re, F. Salemi, and I. Yakushin, Phys. Rev. D **83**, 102001 (2011).
[41] S. Nissanke, J. Sievers, N. Dalal, and D. Holz, Astrophys. J. **739**, 99 (2011).
[42] M. Vallisneri, Phys. Rev. Lett. **107**, 191104 (2011).
[43] S. Vitale and M. Zanolin, Phys. Rev. D **84**, 104020 (2011).
[44] J. Veitch, I. Mandel, B. Aylott, B. Farr, V. Raymond, C. Rodriguez, M. van der Sluys, V. Kalogera, and A. Vecchio, Phys. Rev. D **85**, 104045 (2012).
[45] P. Ajith *et al.*, Class. Quant. Grav. **29**, 124001 (2012).
[46] C. Kozameh, E. T. Newman, and G. Silva-Ortigoza, Class. Quant. Grav. **25**, 145001 (2008) and references therein.
[47] J. N. Goldberg, A. J. Macfarlane, E. T. Newman, F. Rohrlich, and E. C. G. Sudarshan, J. Math. Phys. **8**, 2155 (1967).
[48] L. Gualtieri, E. Berti, V. Cardoso, and U. Sperhake, Phys. Rev. D **78**, 044024 (2008).
[49] "The C++ programming language," (2012).
[50] M. Galassi, *GNU Scientific Library Reference Manual*, 3rd ed. (2009).
[51] "Python programming language," (2012).
[52] F. Pérez and B. E. Granger, Comput. Sci. Eng. **9**, 21 (2007).
[53] C. Doran and A. Lasenby, *Geometric algebra for physicists*, 3rd ed. (Cambridge University Press, 2003).
[54] P. Szekeres, *A course in modern mathematical physics: Groups, Hilbert space, and differential geometry* (Cambridge University Press, 2006).
[55] G. Faye, L. Blanchet, and A. Buonanno, Phys. Rev. D **74**, 104033 (2006).
[56] B. J. Owen, Phys. Rev. D **53**, 6749 (1996).
[57] Y. Pan, A. Buonanno, Y. Chen, and M. Vallisneri, Phys. Rev. D **69**, 104017 (2004).
[58] R. Owen, *Topics in numerical relativity: The periodic standing-wave approximation, the stability of constraints in free evolution, and the spin of dynamical black holes*, Ph.D. thesis, California Institute of Technology (2007).
[59] G. B. Cook and B. F. Whiting, Phys. Rev. D **76**, 041501 (2007).
[60] Appendix A of G. Lovelace, R. Owen, H. P. Pfeiffer, and T. Chu, Phys. Rev. D **78**, 084017 (2008).
[61] K. Alvi, Phys. Rev. D **64**, 104020 (2001).
[62] G. Lovelace, M. Boyle, M. A. Scheel, and B. Szilágyi, Class. Quant. Grav. **29**, 045003 (2012).
[63] E. Ochsner and R. O'Shaughnessy, Phys. Rev. D **86**, 104037 (2012).
[64] O. M. Moreschi, Class. Quant. Grav. **3**, 503 (1986).
[65] T. M. Adamo, C. Kozameh, and E. T. Newman, Living Rev. Relativity **12** (2009).
[66] A. Buonanno, G. Faye, and T. Hinderer, "Spin effects on gravitational waves from inspiraling compact binaries at second post-Newtonian order," (2012), arXiv:1209.6349 [gr-qc].
[67] E. R. Cohen and P. Giacomo, *Symbols, Units, Nomenclature and Fundamental Constants in Physics*, Tech. Rep. (International Union of Pure and Applied Physics, 1987).
[68] ISO Technical Committee 12, *Quantities and Units: Mathematical signs and symbols to be used in the natural sciences and technology*, Tech. Rep. 80000-2 (International Organization for Standardization, 2009).
[69] P. Ajith, M. Boyle, D. A. Brown, S. Fairhurst, M. Hannam, I. Hinder, S. Husa, B. Krishnan, R. A. Mercer, F. Ohme, C. D. Ott, J. S. Read, L. Santamaría, and J. T. Whelan, "Data formats for numerical relativity," (2011), arXiv:0709.0093v3 [gr-qc].
[70] M. Campanelli, C. O. Lousto, H. Nakano, and Y. Zlochower, Phys. Rev. D **79**, 084010 (2009).
[71] M. Boyle, D. A. Brown, L. E. Kidder, A. H. Mroue, H. P. Pfeiffer, M. A. Scheel, G. B. Cook, and S. A. Teukolsky, Phys. Rev. D **76**, 124038 (2007).
[72] M. Boyle, *Accurate gravitational waveforms from binary black-hole systems*, Ph.D. thesis, California Institute of Technology (2008).
[73] M. Boyle and A. H. Mroué, Phys. Rev. D **80**, 124045 (2009).
[74] N. T. Bishop, R. Gomez, P. R. Holvorcem, R. A. Matzner, P. Papadopoulos, and J. Winicour, Phys. Rev. Lett. **76**, 4303 (1996).
[75] J. Winicour, Living Rev. Relativity **15** (2012).
[76] C. Reisswig, N. T. Bishop, and D. Pollney, "General relativistic null-cone evolutions with a high-order scheme," (2012), arXiv:1208.3891 [gr-qc].
[77] M. Boyle, D. A. Brown, and L. Pekowsky, Class. Quant. Grav. **26**, 114006 (2009).
[78] M. Boyle, Phys. Rev. D **84**, 064013 (2011).
[79] I. MacDonald, S. Nissanke, and H. P. Pfeiffer, Class. Quant. Grav. **28**, 134002 (2011).
[80] I. MacDonald, A. H. Mroué, H. P. Pfeiffer, M. Boyle, L. E. Kidder, M. A. Scheel, B. Szilágyi, and N. W. Taylor, Phys. Rev. D **87**, 024009 (2013).
[81] M. A. Scheel, H. P. Pfeiffer, L. Lindblom, L. E. Kidder, O. Rinne, and S. A. Teukolsky, Phys. Rev. D **74**, 104006 (2006).
[82] D. A. Hemberger, M. A. Scheel, L. E. Kidder, B. Szilágyi, and S. A. Teukolsky, "Dynamical excision boundaries in spectral evolutions of binary black hole spacetimes," (2012), arXiv:1211.6079 [gr-qc].
[83] Y. Pan, A. Buonanno, M. Boyle, L. T. Buchman, L. E. Kidder, H. P. Pfeiffer, and M. A. Scheel, Phys. Rev. D **84**, 124052 (2011).
[84] B. J. Kelly and J. G. Baker, "Decoding mode-mixing in black-hole merger ringdown," (2012), arXiv:1212.5553 [gr-qc].
[85] H. Yang, D. A. Nichols, F. Zhang, A. Zimmerman, Z. Zhang, and Y. Chen, Phys. Rev. D **86**, 104006 (2012).
[86] Wolfram Software, Inc., "Mathematica, Version 8.0," (2010).
[87] E. T. Newman and R. Penrose, Phys. Rev. Lett. **15**, 231 (1965).
[88] A. I. Janis and E. T. Newman, J. Math. Phys. **6**, 902 (1965).
[89] E. T. Newman and R. Penrose, J. Math. Phys. **7**, 863 (1966). Note that the sense of the rotation used to define spin weights is somewhat counterintuitive.
[90] Chapter 15 of E. P. Wigner, *Group Theory and Its Applications*

to the Quantum Mechanics of Atomic Spectra (Academic Press, New York, NY, 1959). Note that Wigner defines the Euler angles differently, interchanging the first and last: $\alpha \leftrightarrow \gamma$.

[91] P. Ajith, S. Babak, Y. Chen, M. Hewitson, B. Krishnan, J. T. Whelan, B. Brügmann, P. Diener, J. Gonzalez, M. Hannam, S. Husa, M. Koppitz, D. Pollney, L. Rezzolla, L. Santamaría, A. M. Sintes, U. Sperhake, and J. Thornburg, Class. Quant. Grav. **24**, S689 (2007).

[92] Lemma 5.3 of W. Miller, *Symmetry Groups and Their Applications*, Pure and Applied Mathematics (Academic Press, New York, 1972).

[93] Derived from Eq. (B.10) of J. J. Duistermaat and J. A. C. Kolk, *Lie Groups* (Springer-Verlag, New York, NY, 1999).

[94] F. S. Grassia, Journal of Graphics Tools **3**, 29 (1998).

[95] K. Shoemake, SIGGRAPH Comput. Graph. **19**, 245 (1985).

[96] M. Kim, M. Kim, and S. Y. Shin, in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '95 (ACM, New York, NY, USA, 1995) pp. 369–376.